Reputation-based trust management systems and their applicability to grids

Gheorghe Cosmin Silaghi[1], *Alvaro E. Arenas*[1], *Luis Moura Silva*[2]

[1]{g.c.silaghi, a.e.arenas}@rl.ac.uk
CCLRC, Rutherford Appleton Laboratory
Chilton, Didcot, OX11 0QX, UK

[2]luis@dei.uc.pt
University of Coimbra
3030-290 Coimbra, Portugal

CoreGRID Technical Report
Number TR-0064
February 23, 2007

Institute on Knowledge and Data Management
Institute on System Architecture

CoreGRID - Network of Excellence
URL: http://www.coregrid.net

CoreGRID is a Network of Excellence funded by the European Commission under the Sixth Framework Programme

Project no. FP6-004265

Reputation-based trust management systems and their applicability to grids

Gheorghe Cosmin Silaghi[1], Alvaro E. Arenas[1], Luis Moura Silva[2]

[1]{g.c.silaghi, a.e.arenas}@rl.ac.uk
CCLRC, Rutherford Appleton Laboratory
Chilton, Didcot, OX11 0QX, UK

[2]luis@dei.uc.pt
University of Coimbra
3030-290 Coimbra, Portugal

CoreGRID TR-0064
February 23, 2007

Abstract

This paper reviews the main reputation-based trust systems. We directed our analysis trying to identify how these systems fulfill the requirements of grid computing, anticipating a further inclusion of reputation-based technologies for bringing trust in grid systems. We analyzed a wide area of developments, from reputation systems used in e-commerce to systems coming from agent research, P2P and Grids.

1 Introduction

Trust and reputation systems have been recognized as playing an important role in decision making in the Internet world [20, 24]. Customers and sellers must trust themselves and the services they are offered. Regarding the grid systems, the fundamental idea is that of resource sharing [13]. The grid research was initiated as a way of supporting scientific collaboration, and grid systems were mainly used in e-science projects. Entities from trusted institutions are put together to collaborate and form the grid. However, when grid systems are intended to be used for business purposes, it is necessary to share resources between unknown, un-trusted parties. If one intends to generalize the wide usage of grid systems, the problem of un-trusted parties should be considered. The grid definition of CoreGrid emphasizes the dynamic property of almost every issue: a fully distributed dynamically reconfigurable, scalable and autonomous infrastructure to provide location independent, pervasive, reliable, secure and efficient access to a coordinated set of services encapsulating and virtualizing resources in order to generate knowledge. As the CoreGrid survey material on trust and security acknowledges, modeling trust is of great importance for the future developments of the grid [6].

Reputation-based trust systems were mainly used in electronic markets, as a way of assessing the participants. In a lot of such environments, they proved effective as the number of participants was large and the system was running a sufficient amount of time [39]. But, there are still a lot of issues under study as not everywhere reputation systems were fully effective.

This research work is carried out under the FP6 Network of Excellence CoreGRID funded by the European Commission (Contract IST-2002-004265).

In grid systems, usually trust is constructed and maintained through security mechanisms [6]. Technical advances go toward enabling one point sign-on for an entity in the system, considering that the entities belong to some generally trusted organizations. Authentication mechanisms based on certificates guarantee the nodes belong to the trusted organizations that participate in the computation. One organization can join the computation only if the they agree to fulfill the requirements imposed by the certification authority and if the certification authority agrees to deliver certificates to the new organization. A human-coordinated process is deployed when a new organization wants to join the grid.

But, as the scope of grid enlarges to ubiquitous and pervasive computing, there will be a need to assess and maintain the reputation of any entities, once they are allowed to participate in the grid. We are referring to entities belonging to any organizations or to volunteers, regardless the organization they belong to. We consider reputation as one of the tools that research comunity needs to supply, if we want to let the grid to expand beyond the organizational boundaries.

Our paper intends to evaluate the suitability of existing reputation management systems with regard to the grid security requirements, considering movement of the grid toward computing in untrusted environments.

Several reviews addressed the problem of trust and reputation models for various domains. Grandison and Sloman [20] survey several existing trust models, focusing on Internet applications. The main contribution of this paper is a good conceptual definition for trust and the establishing of some trust properties. They do not address computational trust management models, while they focus more on trust gained by certification. Reputation is not addressed in this review.

Regarding trust in E-Commerce applications, Manchala [34] evaluates some trust metrics but they do not address the reputation problem. Before presenting their developments for trust management through reputation, Zacharia and Maes [54] review some systems live in 2000 that address reputation management in e-commerce sites. Regarding on-line trading environments, Dellarocas [7] analyzes reputation mechanisms from a game-theoretical point of view. He allows opportunistic players to take part of the game and his analysis is fully based on mathematics developments.

Suryanarayana and Taylor [49] address the topic of peer-to-peer applications, analyzing properties of reputation systems related with peer-to-peer requirements. They list the main requirements imposed to the usage of reputation system in P2P environments.

Josang et al. [24] refer to the problem of on-line service provision, but they address the topic of trust and reputation systems from a general point of view, covering applications from both e-commerce and p2p. They analyze the computational engines classified according with their category: simple summation, Bayesian systems, discrete trust models, belief models, fuzzy models and flow models. Also, they describe some reputation systems live at the time moment of the paper. They do not make clear which system to which category belongs to.

Sabater and Sierra [43] review some works regarding reputation as a method for creating trust from the agent-related perspective. They do not categorize the described models, but they try to find how those models are related with some theoretical requirement properties for reputation. The review of Ramchurn et al [37] considers also a multi-agent perspective while debating on the notion of trust. Research in agent systems focus more on properly defining the concept of trust and on supplying with complete and reliable reputation systems, without regard of the usage of the systems in some specific environments like the grid or P2P.

Individual works concerned with building trust through reputation were developed as part of multi-agent research, P2P, mobile ad-hoc networks, grids, virtual organizations etc. This paper we will review such approaches.

In section 2 we will develop the concepts of trust and reputation, establishing some desirable properties a reputation system should fulfill. These will be the properties we will look for when analyzing a reputation system. These properties were extracted in close relationship with the requirements that grid imposes, on the movement toward a widely used grid infrastructure. Section 3 described the main advances in reputation research. We collect studies from a wide area of computer science: multi-agent research, knowledge engineering, grid systems, learning, information retrieval, e-commerce, etc. Section 4 will shortly point on the usage of reputation systems for enhancing grids with fault-tolerance in desktop grids and to improve resource management in classical grids. Section 5 will conclude the paper.

2 Trust and reputation

This section will define the concepts of trust and reputation and will identify the main properties that trust and reputation management should fulfill, considering also the requirements imposed by the grid systems.

2.1 Defining trust and reputation

2.1.1 Trust

According to Gambetta [19], trust refers to the subjective probability by which an individual A expects that another individual B performs a given action on which its welfare depends. This definition taken from sociology is very popular in computer science today.

From the business point of view, the European Commission Joint Research Centre defines trust as the property of a business relationship, such that reliance can be placed on the business partners and the business transactions developed with them [23].

Marsh [35] was one of the first to define the trust concept from a computational point of view. He takes the definition of Deutch [10] which states that trusting behaviour occurs when an individual perceives an ambiguous path, the result of which could be good or bad, and the occurrence of the result is dependent on the actions of another person, the bad result being more harming than the good result beneficial. If the individual chooses to go down that path, he can be said to have made a trustful choice. Marsh agrees that trust implies some degree of uncertainty and hopefulness or optimism regarding an outcome, being subjective and dependent on the views of the individual.

A recent definition of trust is the one of Grandison and Sloman [20]: the firm belief in the competence of an entity to act dependably, securely and reliably within a specified context.

In [24], Josang et al. makes a difference between *reliability trust* as a subjective probability, defined according with Gambetta [19] and the *decision trust* as being the extent in which one party is willing to depend on something or somebody in a given situation with a feeling of relative security, even though negative consequences are possible.

In [12] Falcone and Castelfranchi presents a cognitive view about trust, which is applied in the context of task delegation. When delegating a task, an agent A might evaluate the trust it places in another agent B, considering the different beliefs it has about B: (1) the competence belief: B is competent to do the task, (2) the disposition belief: B actually will do what A needs, (3) the dependence belief: A believes at least that it is better to rely on B for the task than not to rely on it, (4) the fulfillment belief: A beliefs that the task can be done, (5) the willingness belief: B intends to do what it has been proposed to do, (6) persistence belief: B is stable enough in this intentions, (7) the self-confidence belief: A should belief that B knows it can do the task and (8) motivation belief: B has some motive to help A.

The above cognitive approach is worth for consideration in grid systems as a theoretical foundation for empowering grids with trust management, considering the task delegation and resource management requirements of grids. But, when implementing trust management mechanisms, a lot of studies employed the subjective probabilistic view, as being more suited to a computational approach.

2.1.2 Reputation

Reputation is what is generally said or believed about a persons or things character or standing [24]. They argue that reputation is a mean of building trust, as one can trust another based on a good reputation. Therefore, reputation is a measure of trustworthiness, in the sense of reliability.

According to Abdul-Rahman and Hailes [1], a reputation is an expectation about and agent behaviour based on information about or observations of its past behaviour.

This last definition emphasizes the two main sources for building the reputation of an entity: the past experience and the collected referral information. Yu and Singh [53] go further and identify the challenges a reputation management system should address: (1) how an agent rates the correspondent based on the past interaction history, (2) how an agent finds the right witnesses in order to select the referral agents and (3) how the agent systematically incorporates the testimonies of those witnesses.

Other authors argue that reputation is solely gathered from the social network in which the agent is embedded [43]. Therefore, trust can be built from (1) the confidence an agent derives from past interaction and (2) the reputation the agent acquires from the social network [37]. The first source of trust is named *direct trust* while reputation represents an *indirect trust* source.

We will allow reputation to be assessed both from past experience and referrals. Therefore, reputation-based trust systems can be classified in 2 main categories: systems that use only direct trust measures and systems that use both direct and indirect trust.

2.2 Properties of reputation-based trust models

Regarding grid systems, the CoreGrid survey material on Trust and Security [6] acknowledges about the importance of trust management in grids and presents how trust is bring in using security issues. Up to date, reputation based models are barely considered for classical grid systems. As the long term future of the grid is to provide dynamic aggregation of resources, provided as services between businesses, new architectures and detailed mechanisms for bringing together arbitrary resources are required. These architectures should federate security and trust, as ones of the most significant issues[3]. On the basis of the OGSA architecture [11],

- WS-Trust defines a protocol by which web services in different trust domains can exchange security tokens for use in the WS-Security header of a SOAP message.
- WS-Federation describes how to use WS-Trust, WS-Security and WS-Policy together to provide a federation between security domains.

Therefore, in classical grids, trust is achieved through security mechanisms. Attempts like the ones of [29, 32, 50] are among the few to use reputation tools for managing virtual organizations as in grids. Other approaches ([5, 38]) tackle mobile ad-hoc networks. But, the most ones ([9, 21, 28, 31, 47, 52, 55]) address resource management as in P2P applications.

These attempts are mainly based on the developments and requirements identified in P2P systems, as P2P are the models most closed to the fully dynamic and distributed resource management requirements envisioned by the future grids.

Several properties are common to most reputation-based trust models, without any regard of their applicability:

- The computational model: Because grids are based on the distributed computational model, the first property of interest is if the trust mechanism is centralized or decentralized. Centralized models have the disadvantage of a single-failure point, therefore, regarding desktop grids, decentralized systems would be preferable. In classical grids, where security is achieved through certificates and central certification authorities exist, a centralized model could also be of interest. In such systems, one can think to a reputation service in order to be interrogated about the reputation of a user or, more generally, of a resource. This reputation service in this case is a point of centralization.
- Metrics for trust and reputation: when referring to a metric for trust and reputation we consider the value that express the reputation (and trust) of an entity as provided by the reputation mechanism. We should make a distinction between the reputation value of an agent and the feedback one is required to provide at the end of a transaction. Continuous metrics are considered more expressive than discrete ones. Usual, these values are scaled between -1 and 1, or between 0 and 1. If the reputation scheme uses values scaled between 0 and 1 these values can have the meaning of a probability.
- Type of reputation feedback: reputation information might be positive or negative one. Some systems are based on collecting both type of information with regard to an entity, while other systems are based only on negative / positive information. Regarding an accomplished transaction, the reviewer can supply with binary, discrete or continuous values. Again, continuous values are more expressive but for the sake of simplicity, a lot of approaches use discrete feedback and later on aggregates this feedback in continuous reputation or trust.
- Reliability: the trust model should help the users to defend themselves against malicious information, including trust values propagated by other users into the system. The system is reliable if this property is accomplished. Almost all researchers reported that their reputation-based system is reliable for their specific problem under study.

With regard to P2P applications, the following properties might be of interest [49]:

- Local control: in decentralized applications, data are stored at various nodes in the system. As global trust might be stored at the entities in the system, is important not to allow those entities to change the trust and reputation values they maintain. Local control property will have the value yes for those models accomplishing this property.

- Bandwidth cost: in P2P applications, bandwidth is of great importance, as peers communicate via message transfer. In a reputation-based trust system, peers might exchange reputation information, which can increase the bandwidth cost of the network. We would like to have the lower possible bandwidth cost. When a referral network is used to acquire reputation information and if a P2P distributed approach is considered for data storage, the bandwidth cost is increased.

- Storage cost: in a decentralized architecture, the nodes on the grid store trust information about other nodes. One would desire to have as few as possible data replication, and therefore, the storage at each node for trust information should be as less as possible. In a centralized setup, usually, the trust data is stored in the central node and storage cost is less important at the node level. We should acknowledge that the storage cost increases linearly with the number of entities in the system.

- Scalability: the trust model should scale with the number of nodes. Bandwidth and storage costs also increase with new nodes added to the grid, but the trust model should be built in such a way to scale well. We reported the scalability property according with the size of the experiments the authors performed in their papers.

With regard to grid systems we consider the following two properties as of particular importance:

- SLA or QoS negotiation: some reputation models are directly applied for negotiation of service level agreement (SLA) or quality of service (QoS) between 2 parties like a service consumer and producer. In most of the cases, the items to be negotiated and how each party fulfilled the agreements on the specific items part of a SLA are directly incorporated in the direct trust component.

- Trust aggregation: we refer to trust aggregation if the model allows to aggregate trust on an organizational basis. This property is of great importance in the context of VO formation and operation as allows one (1) to obtain the trust and reputation for a VO based on the individual trust on its members or (2) to infer the trust or reputation for an individual based on the trust and reputation of organizations the individual belongs to.

Table 1 depicts a summary of how the models we detailed on section 3 accomplish with these properties. Where information about a specific property was not found on the underlying research, we used the *na* (not available) notation.

3 Reputation systems

In this section we will describe the main reputation systems built up-to-date in the research community. We will start our discourse with a short introduction in the game-theoretical foundations for reputation models.

3.1 The game-theoretical approach for reputation

From the theoretical point of view, economics approaches the problem of reputation in a game-theoretical framework. Agents (players) are continuously playing the same game. When an agent plays the game repeatedly in the same way, it is assumed that the player builds a reputation for playing certain kinds of actions and the rest of the players will learn this reputation. The main concern is when and whether a long-lived player can take advantage of a small probability of a certain type or reputation to effectively commit him to playing as if he were that type [15]. A related question is if reputation models will help one to pick and choose among the many equilibriums of an infinitely repeated game.

Considering long-lived players playing repeatedly the classical prisoner‘s dilemma game, allowing for incomplete information about players type and allowing for building a reputation, the theory can prove that the long-term outcome of the game will be “to cooperate” for both players, although the sole short and long term Nash equilibrium is to defect [33]. Also, in games with only one long-lived player and short-living opponents, considering reputation in the presence of incomplete information lets the theory to prove that the “intuitive” outcome will happen. In [14], Fudenberg and Levine analyzed the chain-store game with this respect. These are good examples to demonstrate how effective reputation can be in gaining a bigger payoff from incomplete information situations where some agents have to consider a decision making.

Game theory usual helps one to demonstrate that is worth to consider reputation information when analyzing the outcome of some competing situations with incomplete information. It is also worth to notice that game theory usual considers reputation as being built only from previous experience of the player within a specific context.

	Centralized	Metric for trust and reputation	Type of feedback	SLA or QoS negotiation	Trust aggregation	Local control	Bandwidth cost	Storage cost	Scalability
eBay	yes	discrete, infinite	discrete	na	no	na	na	na	5
Amazon	yes	[1,5]	discrete	na	no	na	na	na	5
Abdul-Rahman & Hailes [1]	no	discrete	discrete	na	no	no	5	5	na
Singh et al. [48]	no	VSM	na	QoS verification	no	no	5	3	3
Despotovic & Aberer [9]	no	[0,1]	na	QoS verification	no	yes	5	5	3
Josang' subjective logic [25]	no	[0,1]	binary	na	no	no	5	3	na
Yu & Singh [53]	no	[0,1]	continuous	na	no	no	5	3	3
SPORAS and HISTOS [54]	yes	[0,3000]	continuous	na	no	na	na	na	3
REGRET [42]	no	[-1,1]	continuous	QoS verification	yes	no	3	3	5
Ramchurn et al. [36]	no	[0,1]	discrete	SLA-oriented	yes	no	3	3	na
FIRE [22]	no	[-1,1]	continuous	QoS verification	no	no	4	3	3
Gupta et al. [21]	semi	discrete, infinite	na	QoS verification	no	na	3	1	5
TrustMe [47]	no	na	na	na	no	yes	4	1	4
EigenTrust [31]	no	[0,1]	binary	na	no	yes	5	5	3
PeerTrust [52]	no	[0,1]	continuous	na	no	yes	5	5	3
P-Grid [2]	no	na	negative information	na	no	yes	5	5	4
NICE [46]	no	[0,1]	continuous	na	no	yes	3	3	4
GridEigenTrust [50]	yes	[0,1]	continuous	QoS verification	yes	no	1	na	na
PathTrust [32]	yes	(0,1]	binary	na	no	na	na	na	4
Jurca & Faltings [27]	semi	[0,1]	binary	QoS verification	no	na	3	3	4

Table 1: Summary of comparison between reputation-based trust systems

Such approaches were considered for analyzing sensitive options a reputation system designer might have. E.g. Dellarocas [8] proved that a reputation system should not update the reputation of players immediately after a transaction finishes. Rather, if the players' reputation is updated with a-priori established time frequency, the players can learn the reputation of opponents in the game and more cooperation can be induced.

3.2 Reputation in Internet sites

Internet sites mainly use sumation-based reputation systems. These systems are based on counting all votes or grades an entity receives. The votes can be simply counted on the behalf of the user or they can be averaged or weighted. As summation-based reputation systems are mainly used in e-commerce marketplaces, they are mostly centralized. Their big advantage is the simplicity of the reputation scheme. This makes the reputation value to be easily understood by the participants and allows a direct conversion between reputation assessment and trust. The most widely known reputation system of this kind is eBay. Other systems are Amazon, Epinios, BizRate etc.

3.2.1 eBay

The most simplistic approach for assessing reputation is the summation scheme of eBay. eBay [1] is an auction-based e-commerce site for sellers and buyers with millions on items to bid for. The reputation management system is a transaction based one. After the end of an auction, the buyer and the seller have the opportunity to rate each other's performance with either 1 (positive), 0 (neutral) and -1 (negative). The reputation of a user is the sum on these individual feedback and it is a common knowledge into the system. The system stores and manages the reputation centrally. New users receive no reputation and a user may leave the system and rejoin with another identity. The advantage of this reputation scheme is that the reputation measure is easily understood by the participants and therefore, the reputation information can be quickly transformed in a trust knowledge.

In eBay, most of the feedback is positive. Sellers receive negative feedback only 1% of times and buyers 2% [40]. Therefore, the negative information is the most valuable one in the reputation database. Josang et al. [24] classifies this reputation system as primitive, but, as Resnick et al. [41] proves, this primitive reputation system is validated by its long time existence, acknowledging the Yhprums Law: systems that shouldnt work, sometimes do, or at least work fairly well.

Similar feedback summation methods were proposed in other e-commerce websites. Beside summation, averaging the feedback or weighting it was considered.

3.2.2 Amazon

In the Amazon[2] bookstore, reputation is assigned to books and to reviewers. Regarding the books, the reputation of a book is the average score the book received from its reviewers. A reviewer can assign to a book between 1 and 5 stars. Each reviewer has its own reputation. Each time a review is considered helpful by a user, the reviewer receives a vote. The reviewers are ranked based on the votes they received from users.

Similar with eBay, the Amazon reputation system is a centralized one, the reputation is a common knowledge and it has the advantage of simplicity. Anyway, in Amazon, the reputation does not have such a great impact on the whole marketplace, as it can only affect the buying decision, not the price at which the transaction happens. It is also expected that reviewers to receive positive feedback, but, unlike in eBay, Amazon does not display how many negative votes a reviewers received. Amazon reputation system is not a transactional one, as one can vote a reviewer even without buying the item under review. Epinions [3] is another system that offers reviews about products and services in a similar way like Amazon.

This kind of reputation systems is not too valuable for our concern in grids, as they do not allow reputation to directly influence the transaction execution in the system.

3.3 Reputation models based on referrals network

Building trust can be based not only on the past interactions between entities but, also considering the social networks the entity belongs to and the referrals the entity can obtain using the social network. In multi-agent research, Singh et

[1] http://www.ebay.com
[2] http://www.amazon.com
[3] http://www.epinions.com

al. [48] defined the concepts of agent communities and social networks. The members of an online community provide services and referrals for services to each other. A participant in a social network has reputation for both *expertise* (providing good services) and *sociability* (providing good referrals). This approach is widely considered in the agent research community.

The referrals network described by Singh et al. [48] is an agent abstraction for the trust model proposed by Abdul-Rahman and Hailes [1]. Both approaches propose a reputation learning model that updates the sociability reputation of a user according with the outcome of the interaction with that user.

A lot of research was developed considering this approach. The items under study are the way the reputation is stored in the system, how referral information is aggregated, which learning model is used. This section will develop this sort of referral systems.

3.3.1 Abdul-Rahman and Hailes

Abdul-Rahman and Hailes [1] propose a model for computing the trust for an agent in a specific context based on the experience and recommendations. Like with the summation models, trust values are discrete: very trustworthy, trustworthy, untrustworthy and very untrustworthy. Each agent stores the trust values for the agents she interacts with, therefore, the trust model is distributed. Each agent also stores the recommender trust with respect to another agent. The recommender trust value are semantic distances applied for adjusting the recommendation in order to obtain a trust value. They propose a method for evaluating and combining recommendations and updating the trust value. As the model is based on the set theory, each agent has to store all history of past experiences and received recommendations. On a system with a lot of participants and frequent transactions, each agent should have a large storage with this respect. Regarding the network traffic, this is caused by the messages exchanged between agents in order to get reputation information.

The authors provide an example of applying the reputation management scheme, but no computational analysis is provided.

3.3.2 Singh et al.

Considering the same basic assumptions as Abdul-Rahman and Hailes [1], Singh et al. [48] further refine the referral model. Therefore, they assume an agent places queries for services and the responses are of two types: service expertise and referral. Each query and response is expressed as a vector of required expertise. Responses are evaluated based on the similarity between the received service expertise answers and the received referrals weighted with the trust (sociability) in that agent. According with this representation in the vector space model (VSM), each agent has to store its area of expertise and the models of the peers, including peers‘ expertise and sociability. Each agent updates the peers expertise after verifying (by experience) the QoS provided by that peer. If the QoS is bad, therefore, for the whole chain of agents who referred the peer under discussion the sociability measure is decreased. Periodically, each agent decides which peers are worth to keep in its internal model. Therefore, the storage space at each agent is kept in a reasonable limit.

Authors tested the model in a simulated environment with 20 to 60 agents with expertise in 5 fields. The average number of networks was selected as being 4. The main results are the following: (1) the quality of the social network improves over time, (2) the social network stabilizes at an improved quality, (3) when referrals are given, the quality of the system is high than without referrals and (4) a new agent added to an existing stable network will drift toward the neighbours from which it receives improved quality. Another result reported by the authors regards the existence of some privileged agents in the network with a bigger number of neighbours. If this assumption is fulfilled, the overall quality of the system could be improved.

We can observe that with small number of participants in the network, using reputation mechanisms a gain can be obtained in the quality of service assured. This can have some applicability to classical grids, but the strong requirement is that resource and service selection to be an automated task. More, the system is self-organizing and once achieved maturity, a new member is well accommodated by the system. The privileged agents might be assimilated with the central nodes of a classical grid.

3.3.3 Despotovic and Aberer

Although the work [9] of Despotovic and Aberer refers to P2P networks, because it fully employ the referral network as being a source for obtaining recommendations, we categorize this paper as belonging to this last category. They

use a probabilistic approach for assessing peers' trustworthiness in a P2P network. They assume the existence of two interaction contexts: *a direct relationship* where a destination node performs a task and *recommendations* when the destination node acts as a recommender of other nodes in the network. Rather than considering the standard P2P architecture, the graph of nodes is built by linking peers who have one the above-mentioned relationships. Standard models usual weight a recommendation by the trustworthiness of the recommender. Instead, they model the ability of a peer to make recommendations, which is different from the peer trustworthiness.

Each peer j has associated innate probabilities for performing honest or dishonest with others. Other peers when asked about the performance of j may again lie and misreport. Assuming a probability l_k that a peer p_k lies, one can derive the probability of observing a good or bad report from peer k about peer j. Given a sample of independent reports about peer j, one can compute the likelihood behaviour of j, which in turn, depends on the internal probability of agent j for performing honest. Maximizing this likelihood, one can obtain the probability associated with a peer.

The authors state that good predictions can be obtained with 10-20 reports (direct referrals) retrieved. A peer i can learn the misreporting probability l_k by previous experience with peer k, by asking peer k to report about the service quality that peer produced in bilateral direct interactions. Therefore, a full probabilistic model is obtained for predicting the probability of a peer to be honest.

The setup was simulated in an environment with 128 peers, varying number of random direct interactions (from 20 to 100) and varying percentage of liars (0.1 to 0.5). The mean absolute prediction error is low when the proportion of liars is small. The worse results are obtained when half of the population lies and the number of direct interaction is reduced (20).

The authors entered further details, as considering several services provided by the peers of agents and a normal distribution for each peer with regard to the provided QoS. The average QoS provided by a peer is internal to its model. They analyze the following pattern of behaviour: a service provider j provides a service of quality x to a peer j. If j is honest, then it will report the quality x when requested. On the other hand, j will be liar and will report a quality chosen randomly from a normal distribution. Within this setup, they show that a maximum likelihood estimation method can be used to accurately predict the future performance service providers, given the reports are based on their past provided qualities.

Testing this second setup in a network with 128 peers, with 10 to 50 interactions per peer, proportion of liars varying from 0.1 to 0.4, 4 services available on the network and the standard deviations of peers performing a service being 0.3, they obtained good misclassification rates regarding the expected service quality.

This approach is worth for consideration in both classical grids and desktop grids. For desktop grids, the approach does not make too much network traffic as only a small number of recommendations are used for each computation. The model each peer stores locally is not too big, being only the misreporting probability each peer learns about its partners. Also, as simulations proved, good predictions can be obtained, increasing the fault tolerance of the system.

Regarding the usage of the model in classical grids, one can predict the misclassification rate for the expected QoS for a service provided by group of peers (which could be a virtual organization), by employing the probabilistic model described above.

3.4 Belief-oriented trust

These models keep valid the basic assumptions of the referral networks. They refine the above described models by introducing a more sophisticated technique for computing the trust value. The main starting point is that trust is a human belief involving a subject and an object and the trust in a system is a subjective measure. Because of the imperfect knowledge about the reality, one might only have an opinion about trusting an object and this opinion could be a belief, disbelief and uncertainty [25]. The roots of this approach are in the Dempster-Shafer theory of evidence. More, this approach is consistent with the theory of Marsh [35], allowing the existence of two thresholds for expressing trust and untrust beliefs. Trust values are continuous in this case and the storage model can be distributed at the levels of the nodes in the system.

3.4.1 Josang subjective logic

The Josangs subjective logic [25] is a trivalent one, an opinion could have 3 degrees of values: belief (b), disbelief (d) and uncertainty (u), with

$$b + d + u = 1 \qquad \text{with } \{b, d, u\} \in [0, 1]^3$$

Assessing b, d and u from the previous experience of the agent with the object of the trust (which can be another agent or a resource) can be done using the beta distribution function, which is applicable in a space where every event can be successful or unsuccessful.

The subjective logic of Josang introduces the following operators, which can be applied at the internal level of the agent in order to produce the internal trust model.

- the conjunction operator in order to infer a conclusion about a proposition, having two opinions about that proposition
- the consensus operator between independent and dependent opinions
- the recommendation operator, allowing the agent to include in the inference chain the recommendations received from a referral.

The main contribution of Josang' subjective logic is a clear representation of the logic each node in the network should posses in order to manage the experience and the received referrals.

In [26] the author shows the compatibility between the subjective logic and the PGP authentication system, demonstrating the usage of the trust values in grid-like networked environments (based on layered certifications).

They applied this reputation mechanism for improving service discovery in a P2P environment in [51], combining the reputation computation with distributed hash-table routing structure. In this development, referrals are not used, they pursue building the reputation only based on experience and received feedback.

3.4.2 Yu and Singh

The model of Yu and Singh [53] could be more expressive that Josang's one as they allow continuous values in order to assess the outcome of a transaction. According with the Marsh approach, Yu and Singh considers two thresholds (low and high) for assessing the belief or disbelief in the trusting relationship. On their model, they directly use the Dempsters rule of combination in order to aggregate 2 belief functions built on different evidences. This operator has the same meaning as the conjunction operator in the Josang's model.

Considering the referrals network approach previously presented in Singh et al. [48] and solely based on the Dempsters rule combination operator adapted to this environment, they fully describe an agent local decision model for selection of a transaction partner. In order to keep the referral graph restricted (because longer the referral chain is, less reputed is the obtained information) they introduced a depth limit of the referral graph.

They extended their previous experiments to a bigger number of agents (100 to 500), keeping the same vector-based information space for the expertise and the same average number of neighbours. They introduced a new parameter in the experiments: the cooperativeness factor an agent, after selected, might accept to perform a transaction with a certain degree. Computing the overall reputation of the agents in the simulated experiments, they reached the conclusion that overall reputation stabilizes to an equilibrium value. They also simulated the behaviour of a single agent who at the beginning was very cooperative thus gaining a very good reputation. After that, if its cooperativeness factor was reduced to simulate the abuse of having a high reputation, it was proved that its reputation decreased rapidly.

The experiments of Yu and Singh are valuable especially for P2P-based grid communities as they demonstrated formally that the predicted informal behaviour of an agent will really happen.

3.5 Agent-based approaches

Generally, the agent research community sees the agent paradigm as a good formalization for a wide variety of distributed systems, including grids, semantic web, pervasive computing and P2P [22]. The most important property on which they base their discourse is the openness propriety of multi-agent systems, the fact that the agents are self-interested, proactive, know only a local part of the acting world and no central authority restricts the behaviours of all agents. This section will review the main reputation models developed by agent research community.

3.5.1 SPORAS and HISTOS

The systems proposed by Zacharia and Maes in [54] were one of the first attempts to build a reputation-based system to overcome existing trust problems in e-commerce on-line applications. Their ideas were incorporated in later reputation-based trust models.

First, they proposed the SPORAS system, based only on direct transaction ratings between users. Users rate each other after a transaction with continuous values from 0.1 to 1. The ratings received for a user are aggregated in a recursive fashion, obtaining a reputation value that scales from 0 to 3000 and a reputation deviation to assess the reliability of the reputation value. The recursive formula for updating the reputation is based on the following principle: users with very high reputation will experience much smaller rating changes after each update and ratings are discounted over time. The time discount model was further used in FIRE [22].

HISTOS takes into account also the social network created between users through performing transactions. The reputation model for user A_0 from user A_i point of view takes in the consideration all paths on the social network graph between these 2 users. Only paths with positive (greater than 0.5) ratings are considered. As a rating is more far away from the user under discussion, its influence to the total social network rating is lower. The same kind of social network was used after that in the approaches of [9, 46].

When evaluating SPORAS and HISTOS, the authors reported better results than the classical eBay and Amazon approaches. Although, their results were outperformed by more recent studies.

3.5.2 REGRET

In [42] Sabater and Sierra propose a model, named REGRET that considers the following dimensions of the reputation: *the individual dimension*: which is the direct trust obtained by previous experience with another agent, *the social dimension* which refers to the trust of an agent in relation with a group and *the ontological dimension* which reflects the subjective particularities of an individual.

Their model focuses on SLAs between two parties, with several variables under interest. Each agent stores a local database with impressions regarding the accomplishment of an agreed value of a SLA. The impression values are marked with time stamps, are continuous and might be positive, negative or neutral. The subjective reputation of an agent with respect to another agent is computed against a pattern of SLA possible variables and takes into account the impressions stored in the local database weighted with a time discount factor. The reliability of the subjective reputations depends on the number and the variability of the impressions used to compute the reputation. For assessing the individual dimension, the above-described subjective reputation is computed.

For assessing the social dimension, first, the agent aggregates its subjective reputation for the agents members of a target social group. This is the assessment of the agents previous experience with a target group. Second, the subjective reputations of all agents in the same group with our agent are aggregated to obtain the group subjective reputation for a target agent. Third, an overall subjective reputation between groups is obtained by aggregating all subjective reputations between agents belonging to the groups under discussion. The reputation for the social dimension is obtained by aggregating all 3 components described above, including the individual dimension as a 4^{th} component. In all aggregations, weighs are used to reflect the importance the agent puts in one or another component of the aggregation. These weights might change during the agent lifetime

The ontological dimension is computed considering the internal ontological model an agent has with regard to a service. Therefore, one agent might be a good seller if he delivers on date, at an agreed product price with a certain quality. The ontological knowledge of an agent is composed by the aggregation structure between variables and their corresponding weights. To compute the subjective reputation for the ontological dimension, the agent aggregates the individual subjective reputations for the variables part of the structure of a desired variable in a SLA.

This model is simple and allows one to express easily the reputation for the individual experience and the group-related reputation and to compose services by aggregation. The service composition is a well-desired property of grid systems. Besides the internal agent database, another database is required at the level of a group in order to store the group-based reputation. This is a mean of centralization. Although the authors do not say by which mechanism one agent is said to belong to one group this is viewed as a drawback in [49], with regard to grid technologies, one might consider the nodes organization as being the group of that node. The belonging of a node to a group is, therefore resolved by authentication in grid systems.

Another drawback in the opinion of [49] is the lack of referrals traffic. This is indeed a drawback in P2P approaches, as only part of the social information (the one between involved groups) is considered when assessing the general trust, but in classical grids this could be an advantage, as is hard to imagine that a node in an organization will

easily inquire another node in another organization for a reference. Also, the lack of the referral queries decreases the bandwidth cost. The composition of a group reputation by aggregating the individual members reputation might represent a mean of cheating, if most members of a group are unfair ones. But in classical grids one can assume with a high likelihood the good intention of the participant nodes.

Overall, the REGRET system could be of interest for classical grids as allows a way of aggregating the reputation at the level of a group and approach the service composition, which is a particularity of grid systems.

3.5.3 Ramchurn et al.

Taking the assumptions of [42] as valid, Ramchurn et al. [36] further refine the system by detailing more on the terms of a contract (service level agreement - SLA). Their intention is to build a trust model to be directly used in negotiation of a contract. As they see the negotiation process as a successive exchange of offers and counter-offers, they argue that a trust model can short the length of the negotiation and can assure better negotiated values.

The premises of the model are the following: the whole society is composed by groups of agents and each agent is part of one group. Some power-based relations exist between groups. Two agents negotiate on a contract made by several issues, for each issue the negotiation would establish a common accepted value. Agents get some utility after the execution of a contract. Each partner in a contract should have some expectations about the outcome values of the issues. In the environment, all agents must fulfill some societal rules - common to all agents, group rules - common only to agents in a particular group and institutional rules - coming from the interaction environment in which 2 agents negotiate and execute the contract. Each agent stores a history of the agreed contracts and the context (made by the rules) at the time when a contract was negotiated. The trust model is composed by two components: *confidence* - accounting for the direct trust (obtained only by the agents experience) and *reputation* - accounting from the trust obtained from the social environment.

The confidence of an agent in an issue x handled by another agent is a measure of the certainty which allows the first agent to expect a given set of utility deviations to be achieved after the second agent will fulfill the contract. Bigger the confidence is, smaller the expected deviations are. Confidence can be bad, average and good, each of these linguistic label being associated a fuzzy utility function that maps utility deviations in the set $[0, 1]$. The confidence levels for an agent with respect to a contract issue is evaluated from the history of the past interactions, by building a probability density function of the agents utility variation. They employ a similarity function between contract values in order to filter out cases from the history which are not relevant to the actual negotiation context. With this approach they tackle the problem of an agent performing well in a long history of small transactions and after that, cheating in a big and very valuable one transaction [40].

Regarding the reputation, they have a similar view as in [42]. They do not consider the problem of obtaining the reputation from the social environment, assuming that some method exist for getting it (like asking for referrals or the existence of a central reputation service for the group). The reputation measure is continuous, between $[0, 1]$ and reflects the first agents view about a second agent reputation in handling an issue of a contract with respect to a qualifying confidence level. The group reputation is aggregated as in the REGRET model, but considering a bigger weight for more powerful groups. Reputation measure is useful for an agent without prior transaction experience as it can base its negotiation process on it. Confidence and reputation are aggregated in order to obtain the final trust model.

The model principles do not differ too much from the one of Sabater and Sierra [42], but it has the advantage of entering more in the details of the establishment of a SLA. It can be worth for the grid community, as the authors show how trust can be incorporated in the SLA bilateral negotiation process by permitting the adjustment of the proposed values for the issues of a contract in a more reliable way. Even more, when the trust that the negotiation partner will supply the agreed values in the contract is low leading to negative utility expectations, the model shows how an agent can further require more issues in the contract (as a new quality certification) in order to secure a positive utility expectation.

3.5.4 FIRE

In the conception of Huynh et al. [22] a trust model has to (1) take in consideration a wide variety of information sources, (2) the agents should be able to evaluate the trust for themselves (distribution and local control) and (3) the trust model should be robust to lying. They address the first 2 requirements, building a trust model based on 4 different types of trust: (1) interaction trust, (2) role-based trust, (3) witness reputation and (4) certified reputation.

The interaction trust is built considering the previous agents experience, as in the REGRET model. The ratings of a previous transaction are continuous, selected from $[-1, 1]$, only the last H ratings with regard to an issue and another agent are stored in the local database and when aggregating the previous ratings, a time discount function is employed. The role-based trust models the trust resulting from the role-based relationships between 2 agents (e.g. owned by the same company, the relationship between a service provider and its users etc). They propose some rules in order to assess the role-based trust. These rules are of the following form: if 2 roles are considered, a rule expresses the expected performance between agents belonging to these 2 roles and the confidence in the above-assessed expectation. The witness reputation is obtained from the social network of the agent, following a referral process like the one proposed by Yu and Singh [53]. Therefore, queries are required to be propagated through the network in order to compute the witness reputation, which implies a higher bandwidth cost. The certified reputation of an agent consists of a number of certified references about its behaviour on a particular task. Each agent stores its own certified reputation like the references one has on her resume, and when other agent wants to see them, the agent makes its references available. As one agent will reveal the references its has about its previous tasks, it will have the incentive to present only good references, therefore it makes sense to store only the best reference obtained after fulfillment of a transaction.

To obtain the trust value for an agent, one has to aggregate each piece of reputation mentioned above. The authors propose to weight each component as to reflect the emphasis the model puts assigns for each of the information sources above. The weights are normalized. Each trust value is accompanied by a reliability value which, in turn, is composed of two measures: (1) a rating reliability computed on the basis of the weight given for certain component, measuring the quality of the reliability and (2) a deviation reliability measuring the volatility of rating values and therefore, the certainty of the accomplishment of an agreed SLA.

They showed that each component of the model adds an improvement in how reliable and fast an agent finds its partners in transactions. More, they compared the model with a centralized approach (which is supposed to perform better as the whole amount of information is available in one central point) and demonstrated comparable performance levels.

We think that this model is the most complete one from the agent research point of view, combining the advantages of the previously described models of Sabater and Sierra and Ramchurn et al. Only few additional costs are involved, as more model components require more storage and the witness reputation requires a bandwidth cost.

3.5.5 ART Testbed

With the intention to unify the researches with regard to reputation-based trust, a group of researchers from several universities launched the ART Testbed competition, supplying with a testbed for unifying the experiments related with reputation models [17, 18]. The first edition of the contest took place during AAMAS 2006 conference in Hakodate Japan with 14 registered agents, but the idea emerged in 2004 and get contour during spring 2005 with the papers of Fullam et al. presented at AAMAS 2005.

The testbed [17] provides an environment for a limited number of competing agents 6, which have limited expertise in providing some services (evaluation of paintings), and which have to gain as many utility as possible (in terms on money) by performing the service during several rounds of the game. Agents can respond to a service request and might be engaged in exchanging opinion and reputation information. Opinion information regards the agents opinion about the value of a service and the reputation information regards the agents trust in a third agent. Services are assigned to agents by the simulation engine. Therefore, the agents should concentrate only in the social-based reputation model. With this respect, the testbed is valuable as it provides a mean of experimentation for modeling the trust obtained by direct experience and referral trust obtained by gossip through the social network.

Till now, some papers ([16, 30, 45]) were already being produced based on the testbed. But, instead of achieving the goal of experimenting existing strategies in a unified world, these papers focus on the specificity of this environment.

In [30], Kafali and Yolum add a new factor to the reputation model: the self-confidence of the agent, as being the number of times an agent is asked to produce a reputation or an opinion. In their experiments they used agents equipped only with a direct trust model (based on the past transaction experience) and on a mixed model combining direct trust with reputation-based trust. An agent has also to consider its strategy when responding to reputation requests. An agent might respond sincerely to all reputation requests, thus being recognized as an expert and allowing other agents to gain more or might consider to respond only to those agents who performed sincerely in a previously reputation exchange. They found that the most beneficial strategy is to consider the reputation-based trust as part of the trust model and to respond sincerely to all reputation requests.

Sen et al. [45] investigates the existence of cooperation opportunities as part of the testbed setup. They argue that a trustful behaviour should lead towards cooperation between individuals supplying complementary expertise for the overall long-term goodwill of the community. They demonstrate that in the actual environment setup, agents do not have incentive to cooperate on the basis of trust and they propose an improvement in this direction: to change the client share function. They also show by experimentation that such a setup based on trust management can lead for the cooperation between self-interested agents and conclude that an effective trust management scheme should (1) allow agents to be inclined to help someone that has a potential to provide help, (2) allow comparisons between different cooperation costs, (3) be able to flexible adjust inclination to cooperate based on the current work-load and (4) be responsible to changes in types of tasks and types of expertise in the population.

Fullam and Barber [16] see reputation exchange as a mean on learning the trustworthiness of the agents. They apply the q-learning method as decision support. In this method, each agent is rewarded for each action it takes. Therefore, rewards are assigned for requesting and providing opinions and for requesting and providing reputation. The opinion and reputation values are selected according with the actual rewards an agent possesses. Their experiments show that learning agents gains more than non-learning or cheating agents, while it seems that the reputation model has only a little influence to the overall behaviour of the learning agent.

The novelty of this approach is the fact that the trust and reputation profile of the agents in the society is memorized in the form of related rewards. These rewards replace the well-known trust and reputation measures. Their approach is more a competing game-theoretical one. They are not concerned about the overall gains of the game or about the total welfare produced, but rather about the agent who will win the game.

Although one objective of the testbed was to provide a mean of experimentation for reputation methods, it seems that only very few experimentation were pursued on the testbed. Instead, authors focused on its game-theoretical property, trying to win the game rather than to observe the behaviour of a particular already developed reputation model. The paper of [45] revealed some weaknesses of the testbed, from the agent research perspective. From the grid point of view, we can say that the testbed is not too valuable, as it can accommodate only a very small number of participants and the total length of a game do not allow building large history of transactions. More, the testbed focuses only on direct trust obtained by experience and indirect trust obtained by referrals, the other existing types of trust being not present in the testbed. The testbed does not allow trust aggregation, as in the model of REGRET [42], nor SLA negotiation as in the model of Ramchurn et al. [36] Therefore, its suitability for evaluation, with respect to grid research is very limited.

3.6 P2P approaches

In P2P systems, one main concern is the identification of malicious peers that provides misleading services. Trust models might prevent such behaviour and might improve the reliability and fault tolerance of the system. In a P2P approach, the challenge is how to aggregate the local trust values without a centralized storage management and facility. Beside, two kinds of questions are addressed by P2P approaches: what trust metric should be considered and how to store reliable and securely the trust values across the network.

P2P approaches are more suitable for fully decentralized grids, like desktop grids, which come closed with P2P. Regarding their suitability for classical grids, they are quite far from the classical grid problems like SLA and QoS negotiation, or virtual organization management. But, as we will see, ideas from P2P approaches were considered by the grid community, allowing those ideas to be improved by some degree of centralization.

3.6.1 Gupta et al.

Gnutella-like P2P file sharing systems are among the most popular P2P networks. They are fully decentralized and unstructured and file sharing is their objective. In [21], Gupta et al. proposes a reputation system to track back the past behaviour of users and to allow drawing up decisions like who to serve content to and who to request content from. They base their system on the internal properties of such a network, where the most important activities are content search and content download. One objective of the proposed reputation system is to give an idea about the level of participation of the peers in the system. The reputation system proposed by Gupta et al. [21] is a transaction-based one, rather than the user-based approach of TrustMe [47], described in the following subsection.

In this model, the reputation of a peer depends on (1) its behaviour assessed in accordance with the contribution of the peer to content search and download and (2) its capability expressed in terms of processing power, bandwidth, storage capability and memory. Each peer in the network gets credit for (1) processing query response messages, (2)

serving content and (3) sharing hard-to-find content in the network. Content serving and sharing hard-to-find content are assessed based on the quality of the service provided (in terms of the bandwidth and file size). For each download, a peer reputation is debited with a similar amount as for serving the same content. The reputation score is simply a summation of the receiving credits with or without deducing the debits.

Each peer could maintain and compute its reputation locally. But, because there is a misbehavior threat with regard of this operation, a reputation computation agent (RCA) is provided for the P2P network with the goal of keeping track of transactions and of the credits and debits that flows in the network. A peer might choose to participate in the reputation system then it will need to cooperate with the RCA, or might stay apart of the reputation system in this case its reputation is minimal (0). The RCA maintains a transaction state of the system keeping track the full list of transactions and points to be granted for those transactions for a period of time. Each peer communicates with the RCA based on the classical public key cryptography exchange mechanism. After each transaction, each peer reports the transaction to RCA. From time to time the peers contact RCA for being granted with credit for their transactions. The RCA is a central point of failure only for the reputation management scheme. Therefore, the functionality of the P2P network will not be affected if the RCA fails, as it only adds with a supplementary functionality.

The system is simplistic, but covers well the properties of the target P2P network and does not interfere with the standard usage of a Gnutella-like network. Although some misbehavior is still possible as peers might report incorrect transaction details, the system tries to reduce the incentive of multiple identities because a new coming peer always receives no reputation. Some experiments were reported, showing the effectiveness of the reputation system.

This reputation system might have some importance for grid research as it presents a reputation scheme that gives score for desired behaviour and penalizes undesired one therefore, pushing toward cooperative behaviour. Also, it shows how issues part of QoS delivered can be included in the reputation.

3.6.2 TrustMe

TrustMe [47] is another approach for decentralized and unstructured P2P networks. Rather than the approach of [21] which is a transaction-based one, TrustMe is a user-based approach, adopting the principle of obtaining references about a peer, before engaging in a transaction with that peer. Broadly, TrustMe functions in the following manner: each peer is equipped with a couple of public-private key pairs. Trust values of a peer (B) are randomly stored at another peer (THA) in the network. Any peer A interested in the trust value of a peer B broadcast the query on the network and the THA peers replies this query. Based on the received trust value, peer A decides to enter or not in interaction with peer B. After interaction, peer A files a report for peer B indicating the new trust value for B and therefore, THA can modify the trust value of B accordingly. TrustMe uses a smart public key cryptography mechanism to provide security, reliability and accountability. It is assumed that somehow, peer A updates the trust information for peer B and broadcast back this information to its storage located at peer THA.

TrustMe lets free option for selecting the trust measure and focuses on developing a secured message exchange protocol for protecting the information and its sources in the network. Some properties of their proposed protocol are: persistence, no central trusted authority needed, small decision time and ease of contribution. It is out of the scope of this paper to develop the details of message exchanges protocol in TrustMe. But, it is worth for consideration as an alternative way of enforcing trust in a decentralized P2P network.

Comparing this approach with the one of [21], the bandwidth cost is increased, as each peer has to deal also with requests relating reputation besides its usual tasks for responding to search and download queries.

3.6.3 EigenTrust

According to Kamvar et al. [31], the following issues are important in P2P reputation system: (1) self-policing: no central authority should exist and the peers should enforce the ethical behaviour by themselves, (2) anonymity: peer reputation should be associated with an opaque identifier, (3) the system should not assign profit to newcomers, (4) minimal overhead and (5) robust to malicious collectives of peers.

Their approach is based on the notion of transitive trust: a peer i have a high opinion of those peers who have provided it good services and therefore, peer i is likely to trust the opinions of those peers. The idea of transitive trust leads to a system where global trust values correspond to the left principal eigenvector of a matrix of normalized local trust values.

Kamvar et al. [31] considers that each peer stores locally its trust values for the rest of the peers [31]. They do not enforce a method for obtaining these trust values, but they suggest the trust values could be obtained by evaluating

each previous transaction between peers thus being a form of direct trust. Each peer normalizes these trust values obtaining values in the interval $[0, 1]$, 1 being assigned to the most trusted peer. In order to obtain a global view of the network, as in [53], each peer can ask referrals from its neighbours regarding a third peer. The received trust values can be aggregated using the local trust values for the neighbor as weights. Therefore, using one set of queries that is investigating the neighborhood graph on a distance of 1, a peer can obtain a trust vector including witnesses of first order. Iterating and querying the neighbours of the neighbours, the global trust vector becomes much refined. Kamvar et al. proved that by further iterations, the global trust vector converges to a value that is unique for the network and is the left principal eigenvector of the initial matrix of normalized trust values [31]. Therefore, by a repeated query process, each agent can obtain the global trust vector, while still storing locally only its own trust values regarding the rest of the peers. This model has also a remarkable probabilistic interpretation, as a peer might interrogate its neighbours with the probability given by the neighbors local trust value. In order to make the model more resistant to collusion, they propose to consider the founders of the network as a-priori trusted nodes and at each iteration step, to take a part of the trust as being the trust given by these nodes. Addressing the distribution of the storage of the data, the paper lets each node to store also its global trust number part of the global trust vector, besides the normalized trust values. Doing this, the initial a-priori trusted nodes get lost in the network anonymity, making the model more reliable.

Kamvar et al. [31] addresses also some issues which are specific to P2P architectures and are not in the scope of trust management, as how to avoid that a peer to wrongly compute its global trust value. A replication scheme is proposed, allowing each peer to compute the global trust value for other peers in the network.

Regarding the usage of the trust values, they propose to select the peer who will supply a service on a probabilistic basis, taking the selection probability as being a mixture between the global trust value of the peer offering the service and the local value stored at the requesting peer regarding the peer who supplies the service.

Doing some extensive experiments, they showed a good performance of the trust model in the P2P setup. Although, they could not totally reduced the failure rate in the system, but the improvements are significant.

3.6.4 PeerTrust

PeerTrust [52] is based on 5 important parameters contributing to a general trust metric. The 5 parameters considered are: (1) the feedback a peer obtains from other peers, (2) the feedback scope counted as the number of total transactions that a peer has with other peers, (3) the credibility factor for the feedback source, (4) the transaction context factor discriminating between mission-critical and non-critical transactions and (5) the community context factor for addressing community-related characteristics. In fact, as revealed by their general trust metric formula, the trust metric for a peer is composed by the community context factor metric and the weighted satisfaction received for previous transactions.

The weighted feedback received for previous transactions internalizes the first four information sources mentioned above. Regardless of the feedback scheme used by the peers, the feedback should translate into a continuous $[0, 1]$ numerical satisfaction measure, accounting for the first 2 information sources. For assessing the credibility, a first choice they propose is to use recursively the existing trust values of the peers, building an averaged TVM metric. The second choice is to construct the credibility measure from the similarity of satisfaction vectors collected from other peers that interacted with both peers involved in a transaction. The transaction context factor could be in fact a time decay weighting function, allowing that more recent transaction to have a bigger influence. The community context factor can have a very big importance in the model, and its main intention is to provide with a way of convincing peers to give feedback for past transactions. Therefore, they propose as a measure the proportion between the number of the transactions for which a feedback is given and the total number of transactions of that peer. Regarding the distribution of the trust model, each peer has a trust manager that is responsible for feedback submission, for trust evaluation and a database that stores a portion of the global trust data. Several distributed algorithms are proposed for computing the various formulas required by the trust model.

They performed some simulation over several P2P setups with varying number of peers in order to find the effectiveness of the proposed formulas and algorithms. They also considered a defective behaviour of a part of peers. They concluded that the similarity-based approach for measuring the credibility is more efficient than a recursive trust based approach is a setup with malicious peers. When trust-based peer selection is employed in a collusive approach with the similarity-based measure for peers credibility, better results and bigger transaction rate is obtained in comparison with a standard setup without trust-based peer selection.

The importance of the paper is the demonstration that a trust-based mechanism for partner selection in a transaction

is worth for consideration in a P2P approach. Also, they demonstrate that usage of 3rd party information for building credibility (reputation indirect trust) is much valuable that only the own-existing experience. Rather than based on own evaluation of the experience, the model bases on feedback, taking its inspiration from eBay.

With regard to the usage of the model in classical grids, not too many things can be said, as the model does not tackle the problem of QoS and SLA negotiation. The trust measures developed part of the model could be of interest as they proved to be effective is a P2P approach. They also do not tackle the problem of peers belonging to different organizations, which is of interest in classical grids. Although, the model has a great importance regarding desktop grids, as it uses a full P2P approach.

3.6.5 P-Grid

In P-Grid, Aberer and Despotovic [2] see reputation as an assessment of the probability that an agent will cheat. To compute reputation they use data analysis of former transactions. Their trust is binary; an agent can perform a transaction correctly or not. They consider that usual trust exists, and therefore, they disseminate only dishonest information as relevant. They name this information as complains. Therefore, agent p after detecting the malicious behaviour of agent q will store a complaint $c(p, q)$. The total trust of an agent p is defined as the number of complains the agent p stores multiplied with the number of complains about agent p stored by other agents. High values for this trust value indicated the fact that the agent is not trustworthy.

The global trust model is very simplistic and in this approach the main challenge is to store complains in a distributed manner in the network. The P-Grid solution is selected. A P-Grid is a virtual binary search tree, each leaf in the tree being associated with a node from the network. Each node stores data items for which the associated path is a prefix of the data key and also some routing information for directing a search to a complementary node. This P-Grid structure supports 2 operations: insertion of a new node with its related information and query for complains data about an agent. Search is done in $O(\log n)$ time and the storage space required at each agent scales also with $O(\log n)$.

For insertion of a new node, the same insert method is replicated for a number of times, chosen according with the supposed proportion of cheating agents. For locally computing the trust, an agent asks several queries for the same data and after that, she aggregates the received responses, according with the frequency a witness agent is found. The decision regarding whether an agent is trustworthy or not is chosen according with the following heuristics: if an observed value for complaints exceeds the general average of the trust measure too much, the agent must be dishonest.

They evaluated their trust scheme on a population of 128 agents, with different number of cheaters in it and a big number of interactions (6400 or 12800). Good quality for the trust measure is obtained, and this quality can be increased only by increasing the data replication in the P-Grid. The scheme has also the quality of distinguishing well the cheating agents. Therefore, the advantage of the method is that it allows taking decisions regarding peers interactions in an increased number of cases, increasing the reliability of the P2P network.

The method is quite suited for P2P approaches and also for decentralized desktop grids. With regard to a standard grid, anyway, the usability of the method is under question, as it does not address the QoS and SLAs and not also the virtual organization formation.

3.6.6 NICE

The NICE approach [46] targets a specific P2P network implementation, the NICE1 platform. In their view, the trust value of a node B at a node A is a measure of how likely the node A believes a transaction with node B will be successful. They adapted the idea of the social network described in the agent-based approaches to the structure and the security requirements of a fully decentralized P2P network, equipped with a PKI infrastructure. Each agent comes to the system with a pair of public and private keys and the messages are signed by the peers who are creating them. Therefore, after each transaction between a peer client A and a servant B, the peer A generates a cookie with its perceived feedback (trust value) for the transaction. Trust values scales from 0 to 1. Peer A sends the cookie to B and peer B can store the cookie as a reference of its effectiveness in other transactions. Peer B can decide which cookies to store and how long to store such a cookie. More, each peer could posses its own algorithm for updating and storing the trust values it receives from transaction partners.

When a peer A deliberates to enter a transaction with peer B, a cookie might exist between A and B and in this case, this cookie contains the trust peer A has for B. Or previous transactions did not already exist or were discarded. In this case, A will ask its partners about having cookies for B and the partners will continue to spread the request into

the network till a path between A and B is established. As a response to its request, peer A will collect the cookies that link it to B and therefore, will have the graph structure of the social network. On this graph structures paths between A and B are evaluated either by selecting the minimum trust value on the path or by multiplying the trust values. Therefore, the strongest path can be selected. Refinements mechanisms are presented with regard to generating cookies requests. One of them is to allow users to store negative cookies. It is obvious that after a defective transaction, when peer A will generate a cookie for peer B with a low trust value, peer B will simply discard the cookie, as it does not help him. But instead, peer A can retain the cookie as a blacklist, and never entering transactions with peer B.

Experimenting with the system in various setups, the authors proved that the method scales well. Allowing each user to store a maximum 40 cookies and the outdegree (number of peers that receives the same message) of a cookie query to 5, they showed that querying at most 3 nodes in depth is enough to obtain a good representation for the social network. The total number of peers varied from 512 to 2048. When considering also malicious peers in the system (peers that do not follow the NICE trust protocol), a robust cooperative group emerged in the system. As they demonstrated, number of trust-related queries that are forwarded into the network is kept low; therefore, the total bandwidth overhead is minimal. As cookies are small and a peer does not have to memorize too many cookies, the memory requirements are kept also reasonable.

This approach shows how ideas from multi-agent research can be successfully employed in P2P computation. As the NICE is concerned with resource bartering, this environment comes closer to a fully distributed and decentralized grid.

3.7 Incentive compatible approaches

As we have seen in the previous sections, trust can be obtained both from direct interactions and via a third party source. After a transaction is finished, agents have to report about the result. Most studies assumed that agents report truthfully such information (eBay, Amazon, [1, 22, 26, 31, 32, 36, 42, 48, 50, 51, 53]). When considering indirect third party sources to account for the reputation of an agent, again, the third party agent might lie and report incorrect information.

Some of the studies listed before analyzed in some extent the agents truthfulness and how robust the proposed reputation schemes are to such attacks. In this section we will shortly list these results and therefore, we will present the incentive-compatible reputation mechanism of Jurca and Faltings [27], whose design was guided exactly by these considerations.

Despotovic and Aberer [9] experimented their system against liar agents and reported good results when the number of liars is low and there are enough agent interactions. But, the performance gets worse as half of the population lies and the number of direct interactions is reduced. Agents are let to deduce the misreporting probability from their direct interactions.

Regarding the ART testbed setup, Kafali and Yolum [30] barely reported that playing honest when responding to reputation requests is the most beneficial strategy. Fullam and Barber [16] did not investigate the effects of coordinated lying strategies.

In P2P systems, the designers usually do not allow peers to store theirs trust values and the storage model is distributed and replicated through the all network. Most of them consider this trust storage scheme as enough for protecting against lying nodes. In P2P systems like Gnutella, Gupta et al. [21] recognizes an increased possibility of collusion when debits are not considered as part of the reputation measure. Cheating like reporting fake feedback is not possible in this setup, because the reputation points are uniformly given per transaction basis. In TrustMe [47], the authors designated a majority voting protocol in order to assure the reliability of the trust values communicated in the network. In PeerTrust [52] the authors experimented with opportunistic cheating players but they only reported which of their proposed trust scheme performs better. In P-Trust [2] the data replication scheme protects against lying.

Jurca and Faltings [27] proceed with a game-theoretical approach when developing an incentive-compatible reputation mechanism. They argue that it is not in the best of an agent to (i) report reputation information because it provides a competitive advantage to others; (ii) report positive ratings because the agent slightly decrease its own reputation with respect to the average of other agents and therefore, reporting negative ratings the agent will increase its own reputation. They base their model on the classical Prisoner Dilemma played iteratively. They acknowledge that an incentive-compatible mechanism should induce side-payments that make rational for agents to share reputation. These side payments are managed by a set of broker agents called R-Agents that buy and sell reputation information. The interaction protocol is as it follows: before a transaction, the agents select a R-Agent whom they ask about reputation. Each agent asks the R-Agent about the reputation of the partner and pays for this information. After finding

out knowing the reputation of the partner, the agent can decide to engage in the transaction (play the game) or stay apart. If both agents decided to play the game, they enter a contract negotiation stage where they agree about the transaction terms and after that, they do the transaction and receive the payoffs for the transaction. From the payoffs, they can determine the behaviour of the partner in the transaction and submit a report to the selected R-Agent. After submitting the report, they will get a payment for this from the R-Agent. The agents also update their view about the effectiveness of the R-Agents regarding the reputation transactions. Payoffs obtained by transactions and by selling reports to the R-Agents are not interchangeable.

Regarding the payments an agent receives from a R-Agent, they selected the following payment scheme: if agent A reports about agent B behaviour and the report is the same as the next report received about agent B, in this case agent A will receive a positive payment for the report, otherwise nothing. They proved that in the case that the joint probability of lying inside the population is less than 0.5, the agents will be rational by reporting truthfully to R-Agents.

R-Agents are points of centralizing information in the system. It is possible that some R-Agents to have more accurate information than other R-Agents. Therefore, is important for usual agents to learn how to select R-Agents when requesting reputation information about transaction partners. A q-learning scheme is proposed for selection of the R-Agents, each R-Agent being selected according with the maximum expected reward value.

They experimented with this setup and showed that agents that use reputation information before engaging in a transaction accumulated much wealth that agents that did not use reputation information. 40% of bad transactions were eliminated through the usage of the reputation incentive mechanism. More, introducing liar agents in the world, they showed that these agents finished by loosing money, while the trustful agents performed well.

The authors extended the model for pricing services in P2P networks [28] and for improving the service level agreement in the web services world [29]. In [29], they considers groups of customers (like silver, gold and platinum customers) being serviced by providers and submitting binary feedback for the received QoS. The reputation of a provider is therefore the average positive feedback submitted by members of a customers group. Therefore, reputation is identical with the QoS delivered to a group of customers. The reputation mechanism also uses some trusted nodes that submit high trusted reputation reports. To assure that customers will report truthfully about the received QoS, as in the previous work, they consider side-payments for each valid report submitted to the reputation mechanism. Providers have the incentive to supply with the advertised QoS because some penalty payments are considered in the case they missed to accomplish the established SLA. The size of the penalty payments is computed taking into account the reputation of the provider.

These last papers ([28, 29]) are worth for consideration for the Grid community as they show directly how principles of rational behaviour from economics and game theory can be used to put incentives on the grid players to behave for the goodwill of the community. More, although the presented models have some degree of centralization, this is not a drawback in what concerns classical grids, as entity owners can behave as R-Agents for memorizing the reputation of the players.

4 Using reputation in grids

Up to date, there are only few approaches of reputation models to grid systems. Reputation models can bring with more dependability in the grid by tackling the sabotage-tolerance problem or by improving the resource allocation and scheduling in grids. In either cases, the usage of reputation models affects the notion of trust in the grid computing environment, allowing the system to construct the *soft* version of trust. Sabotage-tolerance problem is specific to desktop grids and there are several approaches with this regard. In classical grids, models described in the previous section were applied mainly in resource management with respect to virtual organization formation and evolution phases. In this section we will describe these approaches.

4.1 The sabotage tolerance problem in desktop grids

One problem of big interest in P2P environments, especially in Desktop grids, is the one of sabotage tolerance. In Desktop grids, task allocation is usually done based on the master-worker model [44]. The master submits jobs to workers and after receiving the results, it should have some mechanisms to assess the veridicity of those results, i.e. the computation was performed fairly and the results are correct. Workers have incentive to sabotage the computation in order to get more credit in the system [4].

Among other techniques, reputation-based ones were considered for sabotage-tolerance. Replication with majority voting [4] is the main technique considered for sabotage-tolerance in Desktop grids. Each task is replicated to several workers and if the majority of them produce the same result, that result is accepted as the valid one. This technique is very robust when there are plenty of computational resources and the percentage of saboteours is small. Other techniques, including reputation-based ones need to be considered when resources are scarce and when the proportion of saboteours is important. Techniques for approaching sabotage tolerance will be the topic of another technical report. In what follows, we will describe two important reputation-based techniques for sabotage tolerance.

Sarmenta [44] introduces the *credibility-based fault tolerance* and provides with a framework for combining the benefits of voting (replication), spot-checking or other mechanisms. The idea is to attach credibility values to different objects in the system, as workers, results, result groups and work entries. The credibility of a worker depends on its past observed behavior (i.e. number of spot-checks it passed, the average error rate or the number of results verified by replication). New workers receive less credibility than workers that passed some verification. The credibility of a worker determines the credibility of its results, which affects the credibility of the result group they belong to and the credibility of the result groups are used to assess the credibility of a work entry. The credibility of a work entry is an estimate of the probability to obtain a correct result from it. The usage of the credibility is based on the following principle: if we only accept a result when the conditional probability of that result being correct is at least υ, then the probability of accepting a result as being correct, averaged over all work entries would be at least υ. Therefore, computing and maintaining credibilities over the objects in the system is like estimating the total accuracy of the tasks. We forward the reader to consult the paper of [44] in order to obtain the credibility metrics for each associate fault tolerance method. Credibility is used to reduce either the number of votes required for a replicated task, the replication itself or the number of spot-checkers. Sarmenta [44] proved that credibility-based fault tolerance works well in the case when replication is very costly, i.e. when the total fraction of saboteurs is big.

As used by Sarmenta, credibility is a sort of reputation applied to sabotage-tolerance. It has the basic characteristics of trustworthiness and the credibility has also the meaning of a probability. More, it is built from past experience and credibility of a work entry is built by aggregation.

In [55], Zhao and Lo propose a trust-based scheduling for desktop grid environments. The principle behind is "trust and verify" which combines a trust-based management scheme with a standard result verification mechanism used in such computational environments. The trust-based scheduling of [55] works as follows: the task scheduler fetches tasks from the task queue and selects a set of trusted hosts using the reputation system. The reputation system contains a trusted list of candidate peers each with a trust rating and an optional black list with malicious nodes. A reputation verification module selects some hosts from the trusted list and inserts quizzes with known answer among the tasks that are batch submitted to the hosts. The reputation verification module decides about the number of quizzes to be send to a peer and about their difficulty, based on how reputed is that peer. Peers with a higher reputation need less verification than less reputed peers. Consequence of quiz result verification, the reputation verification module updates the trust rating for the verified peers and sends back the results to the reputation system. Quiz-based verification can be replaced with voting-based replication schemes. Several trust models are proposed for the computation of the trust values, like local trust systems (each peer maintains its own view about the trust of other peers), EigenTrust [31], NICE [46], gossip-based reputation systems (based on asking referrals) or a global centralized reputation systems.

Improvements in the accuracy of the system are observed when using a reputation management technique combined with another sabotage tolerance technique. When confronted with malicious behaviour, quizzes together with a global centralized reputation management system recover and converge faster in order to keep a failure free environment.

4.2 Reputation in classical grids

This subsection will describe the approaches that employed reputation models in classical grids tackling resource management through virtual organizations. Service level agreements and quality of service negotiation are of particular interest.

4.2.1 GridEigenTrust

Based on the MSc thesis of B.E. Alunkal, Laszewski et al. [50] exploits the beneficial properties of EigenTrust [31], extending the model to allow its usage in grids. They integrate the trust management system as part of the QoS

management framework, proposing to probabilistically pre-select the resources based on their likelihood to deliver the requested capability and capacity.

They took the basic framework of EigenTrust and adapt it for grid requirements, resulting the GridEigenTrust model. First, to integrate trust in a QoS management, trust should be related to multiple existing contexts. If we discuss about grids, we need to address entities, organizations and virtual organizations. Considering 2 organizations which entities interact, a trust table will store the direct trust between the organizations, for each context of the transactions. The global trust between organizations at time t is computed by weighting the direct trust table entry with a time decay weight. The trust relationship of organization i for another organization j for a context c is obtained by aggregating the direct trust between these two organizations with the reputation of organization j, weighted with normalized values. The global trust or reputation of an organization j is computed by obtaining recommendations from a 3^{rd} organization and by aggregating the received recommendations with the direct trust values, applying the time decay function specific for the given context. This value is normalized as to scale to $[0, 1]$.

Considering a hierarchical organization of the entities, the trust of an organization will be computed based on the trust of belonging entities. The trust of a virtual organization will be computed based on the trust of the internal organizations. The updated trust of an entity is the weighted average between the old trust of the entity weighted with the time decay measure and the trust of the organization to which the entity belongs to, weighted with the importance (grade) of the entity in the organization. A new organization that just joins the grid may be assigned a low trust or a trust with similar organizations, already part of the grid. The reliability trust of an organization could be obtained by normalized weighted sum of the direct experience and the global trust in that organization. To this weighted sum they also add the grade that users from trusting organization assign to entities part of the trusted organization.

These global reliability trust values are used as normalized trust values in the EigenTrust model, being therefore, used to compute by iteration the global trust vector of the virtual organization. As the P2P architecture of Kamvar et al. [31] is no more of interest, a reputation service manager will perform all trust computation. The reputation service is composed by a data collection manager, a storage manager, a reputation computation manager and a reputation reporter.

The approach of Laszewski et al. [50] is one of the few from literature to propose a reputation service as a way to improve QoS management in grids. Although they present the design of the system, they do not present experiments in order to prove the efficiency of the approach.

4.2.2 PathTrust

PathTrust [32] is a reputation system proposed for member selection in the formation phase of a virtual organization. Because virtual organizations represent one of the main abstraction of the grid [13], we described PathTrust as a grid-related reputation system.

To enter the VO formation process, a member must register with an enterprise network (EN) infrastructure by presenting some credentials. Besides user management, EN supplies with a centralized reputation service. At the dissolution of the VO, each member leaves feedback ratings to the reputation server for other members with whom they experienced transactions. The feedback ratings can be positive or negative ratings. The system requires each transaction to be rated by the participants.

PathTrust arranges the participants in a graph structure similar with the one of NICE [46] or agent-based social networks [48]. Each edge in the graph is weighted with the trust between the nodes at the ends of the edge. This trust is computed by accounting the number of positive feedback let by participant i for participant j and subtracting the number of negative feedback weighted by the report between the total positive feedback and total negative feedback participant i has submitted. If the report is less than 1 that is i submitted more negative feedback, then the weight is 1. The above trust value is normalized by the total number of transactions and therefore, it is less than 1. To distinguish between no transactions experience at all and some existing experience, the trust value is lower bounded by some small value (0.001). The weight of a path in the graph is the product of the weights of the edges that compose that path. As in NICE [46], for assessing the reputation between 2 nodes in the graph, the PathTrust algorithm selects the path with the maximum weight. Like in the EigenTrust [31] approach, the trust value is seen as the probability of selecting a participant from the list of possible alternatives.

They evaluated the PathTrust scheme against the EigenTrust algorithm and against attacks by reporting fake transactions in the system. It seems that with EigenTrust, a cheater can gain more profit than with PathTrust. The second test they performed was against random selection of participants. The results show that EigenTrust looses its advantage over random selection once cheating was introduced in the system. This loss occurs also with PathTrust, but is

much lower. Therefore, to prevent cheating, the authors propose the usage of a transaction fee.

PathTrust is one of the first attempts to apply reputation methods to grids by approaching VO management phases. They approached only partner selection and did not tackled organizational aspects. Their model still lacks of dynamics, as the feedback is collected only at the dissolution of the VO. But, the advance in the field is given by the fact that ideas from previous research were successfully transferred in the area of virtual organizations and grids.

5 Conclusion

Grids pool together resources of various kinds and from various providers. Assuring a trusted computational environment is one of the fundamental requirements in Grid computing. Up-to-date, a lot of efforts were directed toward building trust using security mechanisms. But, as the Grids evolves in the direction of P2P computing and business usage, in the context of a fully transparency and automation at the level of resource-to-job assignments, reputation-based techniques for building trust come into discussion. This paper reviewed the existing research in the area of reputation management, carried out in various fields of computing: Internet, e-commerce, agent systems, P2P and grids. We identified the most important properties a designer has to consider when approaching a reputation management system, depending on the context of applicability.

In general, models based on rational behaviour principles from economics as the one of Jurca and Faltings [27] are worth for consideration as they allow nodes to behave autonomously and still to keep stability and goodwill in the society. Of course, the assumption that trust is a belief [26] and has some degree of uncertainty needs to be incorporated in the model. In the context of classical grids, centralized or semi-centralized approaches are still valid. One has to consider reputation aggregation in the context of virtual organizations, as in the approach of [42]. Other requirement to be considered in the case of classical Grids is the SLA and QoS negotiation. Models that emphasize on SLA [22, 29, 36] are worth for consideration.

For P2P systems and desktop grids, decentralized solutions are required. The approach of Zhao and Lu [2005][55] reports goods results for failure detection with reputation mechanisms. One has to consider memory and bandwidth costs in such networks when devising a reputation management scheme because model distribution incurs such drawbacks. Some reputation management schemes reported good results with respect to these requirements ([21, 46]). Including reputation acquired from the social network is valuable as some papers reported high trust induced in comparison with models using only direct reputation information [52].

In the context of Grid systems, not too many reputation-based approaches are in the market. We can not recommend a reputation system as being the best of all Grid requirements, the design of the reputation mechanism being hardly dependent on the solution used for implementing the Grid middleware, how services are expressed in the Grid and how they are distributed.

In Grids, further research should concentrate on addressing resource selection and job allocation using algorithms that incorporate reputation of entities. Considering the virtual organization concept as the main abstraction of the grid, a reputation model should at least accomplish the trust aggregation and SLA and QoS negotiation requirements. Regarding desktop grids, we think that job allocation can be improved with the usage of reputation models, mainly in the case of untrusted environments with high failure rates or big number of saboteours. Usage of reputation models can reduce the gap that currently exists between classical grids and desktop grids, making desktop grids trustable and allowing them to be used as the classical grids are.

References

[1] A. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. In *HICSS '00: Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 6*, page 6007, Washington, DC, USA, 2000. IEEE Computer Society.

[2] K. Aberer and Z. Despotovic. Managing trust in a peer-2-peer information system. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 310–317, New York, NY, USA, 2001. ACM Press.

[3] M. Ahsant, M. Surridge, T. Leonard, A. Krishna, and O. Mulmo. Dynamic trust federation in grids. In *iTrust2006: Proceedings of the 4th International Conference on Trust Management*, volume 3986 of *Lecture Notes in Computer Science*, pages 3–18. Springer, 2006.

[4] David P. Anderson. Boinc: A system for public-resource computing and storage. In Rajkumar Buyya, editor, *5th International Workshop on Grid Computing (GRID 2004), 8 November 2004, Pittsburgh, PA, USA, Proceedings*, pages 4–10. IEEE Computer Society, 2004.

[5] S. Buchegger and J.Y. Le Boudec. Self-Policing Mobile Ad-Hoc Networks by Reputation. *IEEE Communication Magazine*, 43(7):101–107, 2005.

[6] CoreGrid. D.ia.03 survey material on trust and security. Technical Report D.IA.03, CoreGrid, October 2005. http://www.coregrid.net/mambo/images/stories/IntegrationActivities/TrustandSecurity/d.ia.03.pdf.

[7] Chrysanthos Dellarocas. Reputation mechanism design in online trading environments with pure moral hazard. *Info. Sys. Research*, 16(2):209–230, 2005.

[8] Chrysanthos Dellarocas. How often should reputation mechanisms update a trader's reputation profile? *Info. Sys. Research*, 17(3), 2006.

[9] Z. Despotovic and K. Aberer. Probabilistic prediction of peers' performance in p2p networks. *Engineering Applications of Artificial Intelligence*, 18(3):771–780, 10 2005.

[10] M. Deutch. Cooperation and trust: Some theoretical notes. In *Nebraska Symposium on Motivation*, pages 275–319. Nebraska University Press, 1962.

[11] I. Foster et al. The open grid services architecture, version 1.0. Technical Report GFD-I.030, GGF, 2005.

[12] R. Falcone and C. Castelfranci. Social trust: A cognitive approach. In Cristiano Castelfranchi and Yao-Hua Tan, editors, *Trust and Deception in Virtual Societibes*, pages 55–90. Kluwer Academic Publishers, 2001.

[13] I. Foster, C. Kesselman, and S. Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal of High Performance Computing Applications*, 15(3):200–222, 2001.

[14] D. Fudenberg and K.D. Levine. Maintaining a reputation when strategies are imperfectly observed. *Review of Economic Studies*, 59(3):561–579, July 1992.

[15] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, 1991.

[16] K. Fullam and K.S. Barber. Learning trust strategies in reputation exchange networks. In *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 1241–1248, New York, NY, USA, 2006. ACM Press.

[17] K. Fullam, T. Klos, G. Muller, J. Sabater, A. Schlosser, Z. Topol, K.S. Barber, J.S. Rosenschein, L. Vercouter, and M. Voss. The agent reputation and trust (art) testbed competition game rules (version 1.0). Technical Report TR2004-UT-LIPS-028, The University of Texas at Austin, 2004.

[18] K. Fullam, T. Klos, G. Muller, J. Sabater, Z. Topol, K.S. Barber, J.S. Rosenscheim, and L. Vercouter. The agent reputation and trust (art) testbed architecture. In *Frontiers in Artificial Intelligence and Applications*, volume 131, pages 389–396. IOS Press, 2005.

[19] Diego Gambetta. *Trust: Making and Breaking Cooperative Relations*, chapter Can We Trust Trust?, pages 213–237. Department of Sociology, University of Oxford, 1988. http://www.sociology.ox.ac.uk/papers/gambetta213-237.pdf.

[20] T. Grandison and M. Sloman. A Survey of Trust in Internet Applications. *IEEE Communications Surveys and Tutorials*, 3(4), September 2000.

[21] M. Gupta, P. Judge, and M. Ammar. A reputation system for peer-to-peer networks. In *NOSSDAV '03: Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video*, pages 144–152, New York, NY, USA, 2003. ACM Press.

[22] T.D. Huynh, Nick R. Jennings, and N.R. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 13(2):119–154, 2006.

[23] S. Jones and P. Morris. Trust-ec: requirements for trust and confidence in e-commerce. Technical Report EUR 18749 EN, European Commission Joint Research Centre, 1999.

[24] A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 43(2):618–644, March 2007.

[25] A. Jøsang and S.J. Knapskog. A metric for trusted systems. In *Proceedings of the 21st National Information Systems Security Conference, (NIST-NCSC 1998)*, 1998.

[26] Audun Jøsang. An algebra for assessing trust in certification chains. In *Proceedings of the Network and Distributed System Security Symposium, NDSS 1999, San Diego, California, USA*. The Internet Society, 1999.

[27] R. Jurca and B. Faltings. An incentive compatible reputation mechanism. In *2003 IEEE International Conference on E-Commerce Technology*, page 285, Los Alamitos, CA, USA, 2003. IEEE Computer Society.

[28] R. Jurca and B. Faltings. Reputation-based pricing of p2p services. In *P2PECON '05: Proceeding of the 2005 ACM SIGCOMM workshop on Economics of peer-to-peer systems*, pages 144–149, New York, NY, USA, 2005. ACM Press.

[29] R. Jurca and B. Faltings. Reputation-based service level agreements for web services. In B. Benatallah, F. Casati, and P. Traverso, editors, *ICSOC*, volume 3826 of *Lecture Notes in Computer Science*, pages 396–409. Springer, 2005.

[30] O. Kafali and P. Yolum. Trust strategies for art testbed. In *The workshop Trust in Agent Societies at AAMAS 2006*, 2006.

[31] S.D. Kamvar, M.T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 640–651, New York, NY, USA, 2003. ACM Press.

[32] F. Kerschbaum, J. Haller, Y. Karabulut, and P. Robinson. Pathtrust: A trust-based reputation service for virtual organization formation. In *iTrust2006: Proceedings of the 4th International Conference on Trust Management*, volume 3986 of *Lecture Notes in Computer Science*, pages 193–205. Springer, 2006.

[33] D.M. Kreps and R. Wilson. Reputation and imperfect information. *Journal of Economic Theory*, 27(2):253–279, August 1982.

[34] D.W. Manchala. E-commerce trust metrics and models. *IEEE Internet Computing*, 4(2):36–44, 2000.

[35] S. Marsh. *Formalizing Trust as a Computational Concept*. PhD thesis, University of Stirling, 1994.

[36] S. Ramchurn, N.R. Jennings, C. Sierra, and L. Godo. Devising a trust model for multi-agent interactions using confidence and reputation. *Applied Artificial Intelligence*, 18(9-10):833–852, 2004.

[37] S.D. Ramchurn, D. Huynh, and N.R. Jennings. Trust in multi-agent systems. *Knowl. Eng. Rev.*, 19(1):1–25, March 2004.

[38] Y. Rebahi, V. Mujica, and D. Sisalem. A reputation-based trust mechanism for ad hoc networks. In *ISCC '05: Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC'05)*, pages 37–42, Washington, DC, USA, 2005. IEEE Computer Society.

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

[40] P. Resnick and R. Zeckhauser. *Advances in Applied Microeconomics*, volume 11, chapter Trust Among Strangers in Internet Transactions: Empirical Analysis of eBay's Reputation System. Elsevier, 2002.

[41] P. Resnick, R. Zeckhauser, J. Swanson, and K. Lockwood. The value of reputation on ebay: A controlled experiment. *Experimental Economics*, 9(2):79–101, June 2006.

[42] J. Sabater and C. Sierra. Regret: a reputation model for gregarious societies. In *Fourth Workshop on Deception, Fraud and Trust in Agent Societies*. ACM Press, 2001.

[43] J. Sabater and C. Sierra. Review on computational trust and reputation models. *Artif. Intell. Rev.*, 24(1):33–60, 2005.

[44] Luis F. G. Sarmenta. Sabotage-tolerance mechanisms for volunteer computing systems. *Future Gener. Comput. Syst.*, 18(4):561–572, 2002.

[45] S. Sen, I. Goswami, and S. Airiau. Expertise and trust-based formation of effective coalitions: an evaluation of art testbed. In *The workshop Trust in Agent Societies at AAMAS 2006*, 2006.

[46] Rob Sherwood, L. Seungjoon, and B. Bhattacharjee. Cooperative peer groups in nice. *Computer Networks*, 50(4):523–544, 2006.

[47] A. Singh and Ling Liu. Trustme: Anonymous management of trust relationships in decentralized p2p systems. In *P2P '03: Proceedings of the 3rd International Conference on Peer-to-Peer Computing*, pages 142–149, Washington, DC, USA, 2003. IEEE Computer Society.

[48] M.P. Singh, Bin Yu, and M. Venkatraman. Community-based service location. *Commun. ACM*, 44(4):49–54, 2001.

[49] G. Suryanarayana and R.N. Taylor. A survey of trust management and resource discovery technologies in peer-to-peer applications. Technical Report UCI-ISR-04-6, Institute of Software Research, UC Irvine, 2004.

[50] G. von Laszewski, B.E. Alunkal, and I. Veljkovic. Towards reputable grids. *Scalable Computing: Practice and Experience*, 6(3):95–106, 2005.

[51] R. Wishart, R. Robinson, J. Indulska, and A. Jøsang. Superstringrep: reputation-enhanced service discovery. In *ACSC '05: Proceedings of the Twenty-eighth Australasian conference on Computer Science*, pages 49–57, Darlinghurst, Australia, Australia, 2005. Australian Computer Society, Inc.

[52] Li Xiong and Ling Liu. Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering*, 16(7):843–857, 2004.

[53] Bin Yu and M.P. Singh. An evidential model of distributed reputation management. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 294–301, New York, NY, USA, 2002. ACM Press.

[54] G. Zacharia and P. Maes. Trust management through reputation mechanisms. *Applied Artificial Intelligence*, 14(9):881–907, 2000.

[55] Shanyu Zhao, Virginia Lo, and Chris GauthierDickey. Result verification and trust-based scheduling in peer-to-peer grids. In *P2P '05: Proceedings of the Fifth IEEE International Conference on Peer-to-Peer Computing (P2P'05)*, pages 31–38, Washington, DC, USA, 2005. IEEE Computer Society.