Impact of Trust on the Performance of a Recommendation System in a Social Network*

Stefano Battiston, Frank E. Walter, Frank Schweitzer*

Chair of Systems Design, ETH Zurich, CH-8092 Zurich

*corresponding author, email: fschweitzer@ethz.ch

Abstract

Social agents naturally use their social and professional networks to filter information by trustworthiness. In this paper, we present a model of an automated distributed recommendation system on a social network and we investigate how the dynamics of trust among agents affect the performance of the system. Agents search their social network for recommendations on items to be consumed and the propagation of the query through agents at several degrees of separation enhances the efficiency of their search. Moreover, agents have heterogeneous preferences so that trust between neighbours can be used to filter information coming from remote agents. We identify the range of the density of the network and the degree of heterogeneity of agent preferences in which trust improves the performance of the recommendation system.

1 Introduction

The development of today's information society confronts agents with an information overload on products and services which makes their choice among products and services a demanding task. Recent research in computer science has addressed this issue and a number of technologies, such as recommendation systems, have been proposed [Montaner, López, & de la Rosa, 2003]. Recommendation systems assisting agents in their choice mostly fall into two classes: content-based methods suggest items by matching agent profiles with characteristics of products and services, while collaborative filtering methods measure the similarity of preferences between agents and recommend what similar agents have already chosen [Shih & Liu, 2005]. Often, recommendation systems are centralised and exhibit a bias which is due to them not being independent of the products and services that they provide recommendations on.

Additionally, the diffusion of information technologies in business and social activities results in intricate networks of electronic relationships. In particular, economic activities via electronic transactions require the presence of a system of trust and distrust in order to ensure the fulfilment of contracts [Blaze, Feigenbaum, & Lacy, 1996], [Castelfranchi & Falcone, 1998], [Zacharia & Maes, 2000], [Abdul-Rahman & Hailes, 2000], [Mui, Mohtashemi, & Halberstadt, 2002], [Guha *et al.*, 2004]. Trust plays a crucial role in the functioning of such socio-economic networks, not only by supporting the security of contracts between agents, but also because agents rely on the expertise of other trusted agents in their decision-making.

* This paper was also presented at the Workshop "Trust in Agent Societies" at the Fifth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2006), Hakodate, Japan, 09.05.2006

Along these lines, some recent works have suggested to combine distributed recommendation systems with trust and reputation mechanisms [Montaner, López, & de la Rosa, 2002], [Massa & Bhattacharjee, 2004], [Palau *et al.*, 2004]. Because of the fact that both building expertise and testing items available on the market are costly activities, individuals in the real world attempt to reduce such costs through the use of their social/professional networks.

Such complex networks, in particular their structure and function, are the subject of an extensive and growing body of research across disciplines [Newman, 2003]. Social networks have received special attention [Battiston & Catanzaro, 2004] and it has been shown that their structure plays an important role in decision making processes [Garlaschelli *et al.*, 2003], [Battiston, Bonabeau, & Weisbuch, 2003].

2 Description of the model

2.1 General outline

The issue we address in this paper is whether it is desirable and feasible to have an automated distributed system that enables agents in a social network to share knowledge and expertise with other trusted agents. In the following, we use the notion of trust in the sense of similarity of preferences of two agents. The major challenge to face is that agents have heterogeneous preferences: any two agents may have different preferences on the same item, meaning that they experience different utilities or degrees of satisfaction from the consumption of the item.

In this paper, we present a simple model of a recommendation system in which agents search their social network for recommendations on items to be purchased. The social network is static in the current version of the model; the case of an evolving network will be investigated in a further work. When facing to purchase an item, agents query their neighbourhood for recommendations on the item to purchase. Neighbours in turn pass on a query to their neighbours in case that they cannot provide a reply themselves. In this way, the network replies to a query of an individual by offering a set of recommendations. One to deal with these recommendations would be to choose the most frequently recommended item. However, because of the heterogeneity of preferences of agents, this may not be the most efficient strategy in terms of utility. Thus, we explore means to incorporate knowledge of trustworthiness of recommendations into the system. However, it is also reasonable to expect that if agents are too heterogeneous in their preferences, trust relationships cannot be established or they are of little use. Therefore, by means of analytical calculations and computer simulations, we investigate under which conditions and to what extent the presence of a trust system enhances the performance of a recommendation system on a social network.

2.2 Illustrative example

The situation we want to model could be illustrated by the following caricature: a consumer needs to buy a bottle of Swiss wine and, being new to the country, does not know which one to choose. Therefore, he contacts his friends and asks them for advice. They either have a piece of advice or they pass the question on to their friends. After some time, the consumer receives a number of recommendations, say 6, for specific brands of Swiss wine to choose. For instance, 3 recommendations could suggest brand a, another 2 brand b and 1 brand c. How can the consumer make the best use of the recommendations? He might choose brand a because it is the most recommended, but he may also know that brand c has been recommended by a friend of a friend who is known to be a real expert in wines. Let us assume that the consumer decides for brand a because it is the most frequently recommended. However, upon consumption, he discovers that this brand does not match his taste. At the next time when he goes shopping for wine, the agent will give less importance to the recommendations of those agents that recommended brand a and may even try brand b or c. What people intuitively do in real life is to keep a mental map of the level of trust that they have towards the advice of friends in a particular context. However, this is a difficult task when the market offers thousands of product and service categories as well as dozens of brands in each category.

2.3 Agent's preferences

In this work, we investigate when and to what extent trust can improve the satisfaction of consumers that use a recommendation system. Satisfaction of consumers in our model is defined through the utility that these consumers realize by following certain recommodations. To formalize the model, we consider a set of N agents (denoted by $i, j, k, l, ...$) who need to consume items from a set of size M (denoted by $a, b, c, d, ...$), classified in L categories. Categories correspond to product types, while items correspond to specific brands. Hence, a product type (such as the Swiss wine from the example scenario) is available from several brands (which are the specific manufacturers). Each agent i has a predetermined and static preference f_{a_i} on each item a; where f_{a_i} is a real value from the range $[-1, 1]$. Negative values correspond to dislike, positive values to fondness for a certain item. This corresponds to the fact that we, as the modeler, know the value of satisfaction (or preference) of a consumer for a certain brand and this does not change over time. The consumer himself does not know its preference until he consumes the item. This assumption can be relaxed in future versions of the model.

Then, the utility of agent i at time t is defined as the averaged sum over preferences of items consumed until that time:

$$U_i(t) = \frac{1}{t} \sum_a f_{a_i} \tag{1}$$

The performance of the system at time t is then evaluated by the average of the utilities of all agents:

$$\Phi(t) = \frac{1}{N} \sum_i U_i(t) \tag{2}$$

Agents have a finite memory so they know their preference value on the last m items that they have consumed and they can give advice on any of these items. An example of a memory of an agent is illustrated in table 1.

	Item	Category	Preference	
1	5	83	0.749	← Newest
2	4	96	0.230	
3	6	25	0.830	
4	3	44	-0.649	
5	7	59	0.561	
...	...	...	...	
$(m-3)$	8	59	-0.378	
$(m-2)$	1	26	-0.962	
$(m-1)$	9	07	-0.348	
m	0	95	0.523	← Oldest

Table 1: Each agent has a memory of size m in which he stores m 3-tuples consisting of a category, an item, and the corresponding preference value. The memory is a queue, i.e. the newest entry makes the oldest entry fade away.

For further purposes, we eventually introduce a measure of similarity $\omega_{i,j}$ between the preferences of two agents,

$$\omega_{i,j} = \sum_{a} \left(1 - |f_{a_i} - f_{a_j}|\right) \tag{3}$$

2.4 Agent's social network

Agents are connected in a social network that we assume to be a static random graph. In further work, we will consider other network topologies and the network will evolve in time. Each agent i is connected to a few other agents. Because a random graph, if dense enough, has "small world" properties [Watts & Strogatz, 1998], an agent can reach almost everybody in a few hops, so he will always get a recommendation, even for very rare categories. The problem is, can the agent trust the recommendation, especially if it comes from a neighbour of order K (i.e., the length of the path to the recommending agent is K)? In figure 1 agent i receives a recommendation from agent j. If each agent assigns a value of trust to his first neighbours and updates the trust values over time according to the appropriateness of the recommendations received, then a value of trust can also be assigned to the path between any two agents.

2.5 Agent's recommendations

Over time, agents have to buy items from several categories for which they need recommendations. Therefore, they have to send queries to their peers asking what would be the best item in each category. Their goal naturally is to choose the items that correspond to highest values of preference for themselves. Each query reaches a number of agents that have to process it (either respond or pass it over to their neighbours). It therefore has a cost that we assume to grow with the number of agents involved in the process.

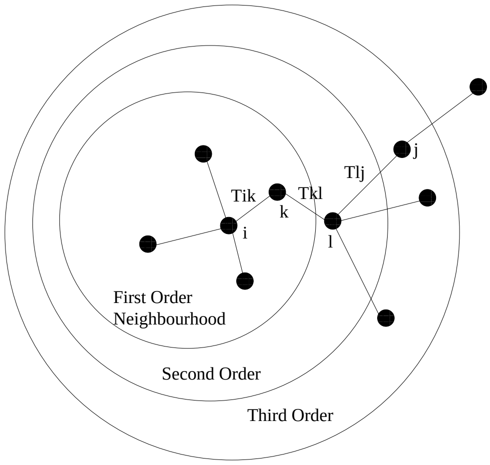

Figure 1: Agents are connected in a social network and they ask their peers for recommendations on items. Each agent i assigns a value of trust T_{ik} to his first neighbours k. In this way, a value of trust can be assigned to the path between any two agents i and j.

The following protocol is used in the search for recommendations: During a time step, each agent, chosen in random order without repetitions, sends one query about a category chosen at random to his peers and receives a set of r responses in return. A **query** consists of:

1. the id of the agent i that has issued the query
2. the id of the category c of items that the query is on

If an agent receiving a query has information on the category, it sends it back and considers its task done. Otherwise, it passes on the query to its (other) neighbours. In case of a **cycle** in the network, an agent may receive the query it sent itself or a query that it has already received and processed; it then does not take any action. In essence, this is a breadth-first search (BFS) of the social network of agent i.

As a **result of one query** agent i receives a **set of responses** (whose size s satisfies $s \geq 0$ and $s < N - 1$). In most cases, there is at least one response for a query. More precisely, the parameters controlling the density of the network and the memory of the agents can be set in such a way that, for most queries, there is at least one response.

A **response** consists of:

1. the id of the item a recommended
2. a preference value u_a according to the agent that responds to the query
3. a value of trust τ_a associated with the recommendation (see 4)
4. the id of the agent j that responds (only in case of direct responses, otherwise, if the responding agent is not among the first neighbours of the querying agent, its identity remains unknown to the querying agent)

The **value of trust** τ_a associated to the item a is the product of the trust from agent to agent along the path from the querying agent i to the recommending agent j:

$$\tau_a = \prod_{(k,l) \,\in\, \mathrm{path}(i,j)} T_{k,l} \tag{4}$$

where $T_{k,l}$ is the current trust associated with a link on the path between agents i and j.

2.6 Agent's decisions

The querying agent uses the set of responses obtained from the network in its decision to make a choice of the next item to purchase. In absence of trust, we assume the probability that an agent chooses an item a is proportional to the number of recommendations received for it. That is, if an item appears repeatedly in the set of recommendations, it consequently has more chances to be chosen. In presence of trust, we instead assume that the agents choose the items in a way that takes trust into account.

This is accomplished in the following way: If the set of responses agent i received for its query its not empty, it contains k recommendations r_l; i.e., it consists of k ids of the items, k values of preferences $[u_{i_1}, ..., u_{i_k}]$ and k values of trust associated with the path that the recommendations came along, $[\tau_{i_1}, ..., \tau_{i_k}]$. Agent i then chooses among the recommendations r_l for category c with a probability given by the logit function of the products of value of the recommendationand trust:

$$P_{i,r} = \frac{\exp(\beta \tau_r f_{r_i})}{\sum_j \exp(\beta \tau_r f_{r_j})} \tag{5}$$

The intuition behind the formula is as follows. A recommendation has good chances to be chosen if both the trust along the path and the preference value of the item in the recommendation (according to the agent that recommended the item) are high. If the item is in the memory of the agent itself, then the preference value is known (and the trust is set to 1, i.e., each agent has infinite trust in itself). If one item appears several times with both high trust and high value, it naturally has more chances to be chosen. The items for which there is no recommendation have a small, but non-null probability to be chosen. The parameter β measures the tendency to follow a recommendation. The smaller the value of β, the more the chances are equalized among items (recommended or not). If $\beta = 0$, all items are chosen with the same probability. If $\beta \gg 1$, the same item will be chosen again with high probability. In this sense, β measures the degree of *explorativity* of agents.

2.7 Agent's trust dynamics

As commonly assumed in models of trust mechanisms [Montaner, López, & de la Rosa, 2002], trust of agent i towards another agent j evolves in time as a discounted sum of the similarity between the preferences experienced by the agent and those communicated by the recommenders. Hence, we define trust as a value $T_{i,j} \in [0, 1]$, where 0 indicates *no* trust and 1 *complete* trust. For all i, j trust is initialized as $T_{i,j} = 0.5$. Agent i updates its trust only towards its first neighbours j and only if i has chosen an item directly recommended by j or i has used a recommendation which came through j.

For the update dynamics, an approach inspired by [Montaner, López, & de la Rosa, 2002] is used:

$$T_{i,j}(t+1) = \frac{1}{2}\left[1 + \tanh\left\{\beta \tilde{T}_{i,j}(t+1)\right\}\right] \quad (6)$$

where

$$\tilde{T}_{i,j}(t+1) = \gamma \tilde{T}_{i,j}(t) + r_{i,j,a} \quad (7)$$

and

$$r_{i,j,a} = (1 - |f_{i,a} - u_a|) \quad (8)$$

$r_{i,j,a}$ is a measure of the similarity of the preferences of agent i and the recommendation u_a of the responding agent j with respect to item a. It is 1 if the agents identical preferences and -1 if they have opposite preferences. The similarity value affects the trust, however we have to assure that trust is bounded in the interval $[0, 1]$. This is obtained by introducing an intermediate unbounded variable $\tilde{T}_{i,j}$, which is a discounted memory of the similarities between the two agents under consideration. The parameter $\gamma \in [0, 1]$ is a discount factor which can be interpreted as the inverse of a memory time. $\tilde{T}_{i,j}$ is then rescaled into the interval $[0, 1]$ by means of eq. 6. β is the same parameter as used in the decision making process, eqn. (5).

3 Results of computer simulations

In this section, we discuss the results obtained through computer simulations and analytical approximations of the model. The core observation made was that recommendation systems in trust-based networks outperform majority-based recommendation systems within a range of

- *network density*: if the network is not dense enough, agents receive replies with recommendations for only on a fraction of the items that they send queries about; as well as
- *preference heterogeneity*: if agents are too homogeneous, there is no need for filtering the recommendations; if agents are too heterogeneous, they cannot find other agents to act as suitable filters for them.

We obtained these results by computer simulations and analytical approximations using the following simplifications of the model: The preferences f_{a_i}, instead of being continuous variables from the range $[-1, 1]$, take only the discrete values of either +1 or -1. Then, for a given number of items in a category, there is only a finite number, k, of possible assignments of preferences to the items in the category. These assignments we call *profiles*. We assume that the knowledge about categories is distributed among agents in such a way that each agent has knowledge about an equal fraction of categories, thus mimicking the role of a memory.

In our simulations we consider for simplicity the case of $k = 2$ profiles, $N = 100$ agents and $M = 2$ items, i.e., there are two possible profiles in the population, $v_+ = [1, -1]$ and $v_- = [-1, +1]$. We define the fraction of agents characterized by a given profile as $n_+ = N_+/N$ and $n_- = N_-/N = (1 - n_+)$. Further, for the social network we assume a random graph with a given number of agents, N, and a given total number of links, ℓ. The network density is then defined as $p = 2\ell/N(N-1)$. In our model agents are connected randomly with respect to their profile.

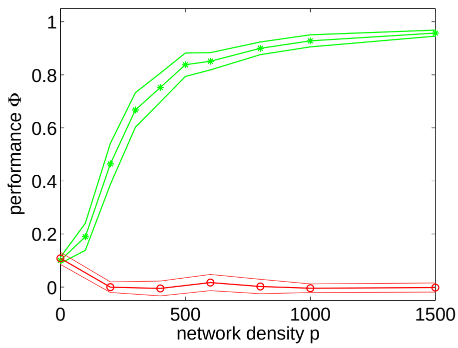

Figure 2: Performance as a function of the the network density. The curves show the mean value and the standard deviation obtained from 10 runs.

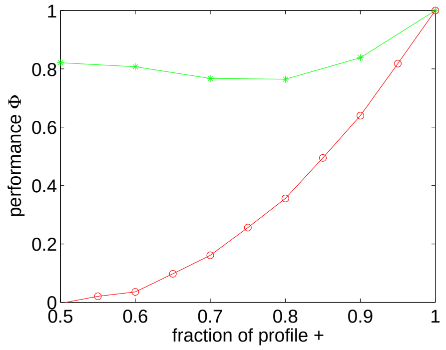

Figure 3: Performance as a function of the the heterogeneity of preferences.

In Figs. 2, 3 the performance of the system, Φ, eqn. (2), is plotted against two crucial variables of the model, network density p and preference heterogeneity of agents, expressed by n_+. In each figure, the performance with trust (green line) is compared to the performance without trust (red line). The latter one means that agents choose randomly among the recommendations received, whereas in the case with trust agents follow the decision making and trust update process outlined above.

In Fig. 2, we see that without trust the performance is zero on average, because random choices lead to an equal distribution of "good" and "bad" (with respect to preferences) products. Furthermore, for low network densities, the probability of an agent not receiving a recommendation is higher because there might be disconnected components in the network structure. For an increasing network density the performance with trust rapidly increases until it reaches a saturation.

In Fig. 3, we see that an equal distribution of both profiles, 0.5, without trust results in zero performance again, as explained in the previous paragraph. If there is an increasing fraction of agents with a certain profile, then the performance increases even without trust, because despite random choices the majority gives matching recommodations. Therefore, for an equal distribution of both profiles the difference between the cases with and without trust is greatest.

4 Analytical approximation

4.1 Similarity between agents

For the above mentioned assumptions, we can obtain the following analytical treatment of the model. First, the expected value of similarity, ω, can be calculated. $\omega_{i,j}$ was defined in (3). We now consider the average of $\omega_{i,j}$ over a set E of pairs of agents:

$$\langle\omega\rangle = \langle\omega_{i,j}\rangle = \frac{1}{|E|}\sum_{i,j\in E}\omega_{i,j} \tag{9}$$

where $|E|$ is the cardinality of the set. With such definitions, the similarity of two agents is 1 if their preferences over the products are identical, and -1 if their preferences over the products are always opposite, and 0 if half of their preferences are identical and half are opposite.

Because agents are connected randomly, for any pair of chosen agents i, j, the probability that their profiles are both v_+ or both v_-, or mixed is: $(n_+)^2$, $(n_-)^2 = (1 - n_+)^2$ and $(n_+)(n_-) + (n_-)(n_+) = 2(n_+)(1 - n_+)$. Correspondingly, the values of $\omega_{i,j}$ are $1, -1$ with probability $(n_+)^2 + (1 - n_+)^2, 2(n_+)(1 - n_+)$ respectively. The expected value of ω of the similarity over a large set of pairs of agents is then:

$$\begin{aligned}\langle\omega\rangle &= n_+^2 + (1 - n_+)^2 - 2(n_+)(1 - n_+) \\ &= 4n_+^2 - 4n_+ + 1\end{aligned} \tag{10}$$

If profiles are evenly distributed among agents ($n_+ = 1/2$), then $\langle\omega\rangle$ is 0 for large N. If instead, agents have only one profile, $n_+ = 1$, then trivially $\langle\omega\rangle = 1$.

In the general case of M items per category, the number of possible profiles can be calculated as follows. If we assume that for each category there is an *even* number M of items, the preference of agent i on item a in category c is $f_{a_i,c} = \pm 1$ with the constraint $\sum_a f_{a_i,c} = 0 \ \forall i, c$. With this constraint, the number of possible profiles with the vectors $v_i = [f_{1_i,c}, ..., f_{M_i,c}]$ is

$$k = \frac{M!}{(M/2)!\,M/2!} \tag{11}$$

This is the number of possible distinct ways of placing $M/2$ entries (± 1) in a vector of length M. Each profile occurs with frequency n_l. There are then k^2 different possible values of similarity ω, each occurring with frequency $n_l n_m$ $(l, m = 1...k)$. The expected value $\langle\omega\rangle$ can then be computed knowing the frequencies of the profiles in the population. If all profiles are evenly represented (with frequencies $1/k$) then, by symmetry, for large N, $\langle\omega\rangle$ is still 0.

We further assume that the vectors v_i are the same for each category, therefore the number of profiles is independent of the number of categories.

4.2 Stationary trust dynamics

We can show that the dynamics of trust among any two connected agents i, j reaches a stable fix point dependent on the respective preference profiles. Neighbouring agents with identical preference profile will, over time, develop a trust value close to 1 while those with profiles completely anti-symmetric to each other will develop a trust value close to 0.

Consider agent i and one of its neighbours j. Over time j provides or transfers recommendations to i about different items. When agent i chooses a item for which it has received a recommendation from j, it updates its level of trust towards j by the term $r_{i,j,a} = (1 - |f_{a_i} - u_a|))$ (see eq. (8)). This term depends on the item a and is therefore subject to a random process. Moreover, it should be noticed that, if the recommendation comes directly from j, the value u_a communicated to i in the recommendation coincides with the preference value f_{a_j} of j. In this case $r_{i,j,a} = (1 - |f_{a_j} - f_{a_j}|))$ holds. Otherwise, the value u_a communicated to i is the preference value according to the recommender. However, after a number of updates of the trust values we can assume that if i chooses this recommendation it is because the value of trust associated with it is high. This in turn implies that there is a pathway of high trust connecting the origin of the recommendation and the neighbour j. In other words, we can, after a certain time, expect u_a to be close to f_{a_j}.

In the so-called mean field approach, one approximates the fluctuating term of a stochastic equation with its expected average. Within certain limits, the solution provides very useful insights. We will apply the mean field approach as follows. We assume $r_{i,j,a} = (1 - |f_{a_i} - f_{a_j}|))$ for all recommendations. We further approximate the term with its average over time, which tends to its average over the items and therefore to $\omega_{i,j}$. In this approximation, we can solve eq. 7 and the dynamics of $\tilde{T}$

$$\tilde{T}_{i,j} = \gamma \tilde{T}_{i,j} + r_{i,j,a} \simeq \gamma \tilde{T}_{i,j} + \omega_{i,j} \tag{12}$$

has a fix point in:

$$\tilde{T}^{\star}_{i,j} = \frac{\omega_{i,j}}{1-\gamma} \tag{13}$$

The corresponding value of trust among any two connected agents is then

$$T^{\star}_{i,j} = \frac{1}{2}\left[1 + \tanh\left\{\beta \tilde{T}^{\star}_{i,j}\right\}\right] \tag{14}$$

For example, with $\gamma = 0.8$ we have: $\tilde{T}^{\star}_{i,j} = -5, 0, 5$ for the values of similarity $1, 0, -1$; if now $\beta = 1$ we also have $T^{\star}_{i,j} = 0.9999, 0.5, 4.5e - 005$.

Despite the existence of fixed points, the trust dynamics is very sensitive to a wrong recommendations from trusted agents. The corresponding value of trust decreases fast and keep decreasing if good recommendations are not provided.

4.3 Random Graph Structure and Critical Density

It is well known that in a random graph (à la Erdös-Rényi) of N nodes and ℓ links, a *giant connected component* appears for values of $\ell > (N-1)/2$, meaning that the probability that the network is connected tends to 1 for large N [Newman, 2003]. Equivalently, this also means that there is at least one pathway between any two randomly chosen nodes. In our model, agents are connected in a random graph and have different preference profiles, distributed randomly according to some frequency distribution. In the case of $M = 2$, we have discussed the probability of a pair of agents having the same or opposite profiles, which we can interpret here as expected frequency of links among agents with the same profile. We can then ask what is the critical density of links (randomly drawn among agents of any profile) in the network such that there is (in the limit of many agents) at least one pathway between any two agents with a same profile. In this situation, a querying agent is able to receive recommendations from all other agents with its same profile along pathways which involve only agents with its same profile (and therefore with high trust). We denote $\ell_{++} = \ell n_+^2$ to be the number of links among any two agents with same profile v_+. The condition for the existence of a giant component becomes then:

$$\ell_{++} = \ell n_+^2 > (N-1)/2 \rightarrow \ell > \frac{N-1}{2n_+^2} \qquad (15)$$

The formula holds in general for any number of profiles in the systems, replacing n_+ with the frequency n_i of the $i^t h$ profile.

4.4 Performance

As described in the decision making process, at each time step, as a result of a query for a given category, an agent receives a set of recommendations: $r_1, ..., r_k$, associated with trust values $\tau_{r_1}, ..., \tau_{r_k}$. The agent then faces a choice on a basket of options which includes the items for which the agent has a value in memory as well as the recommendations. The items for which the agent has a value in memory are associated with $\tau_r = 1$ (complete trust). Each option is chosen by agent q with probability P_{q,r_i} proportional to $\exp(\beta\tau_{r_i})u_{r_i} = \exp(\beta\tau_{r_i})$. Each option, if chosen, leads to experience the preference value of the associate item, denoted as f_{i,r_i}. Recall that there can be more recommendations for a same item, which implies that the chances to choose that item adds up. Agents reply to a recommendation only if for the requested category they have an item in their memory for which their own preference value is positive, namely $+1$ in the simplified case. Then $u_{r_i} = 1$ for all r_i in the set.

If an agent q makes one choice on a given set of recommendations r, each selected with probability $P_{q,r}$ as defined above, then the expected preference value from the choice is:

$$E(f_q) = \sum_r f_{q,r} P_{a,r} = \frac{\sum_r f_{q,r} \exp(\beta \tau_r)}{\sum_r \exp(\beta \tau_r)} \tag{16}$$

Recall that $r_{i,j,a} = 1 - |f_{q,a} - f_{r,a}| = \pm 1 \leftrightarrow f_{i,a} = \pm 1$. If the agent q follows exactly once the recommendations of agent r on each of the M items of a category that have high preference for agent r, then it is easy to check that its average utility $U_{q,r}$ coincides with its value of similarity to r:

$$U_{q,r} = \frac{2}{M} \sum_{a \text{with } f_{r,a} = +1} f_{q,a} = \frac{2}{M} \sum_a r_{q,r} = \omega_{q,r} \tag{17}$$

In reality, in the dynamics of the model, recommendations from agent r on the item of a category are received as a stochastic process. However, over time we can reasonably approximate the average utility $U_{q,r}$ with the value of similarity among the profiles of the two agents. Moreover, the probability that agent q follows a recommendation on a item does not depend on the item itself, but only on the trust towards the recommender agent. This probability is $P_{q,r} = \exp(\beta \tau_{q,r})$.
If β is larger than 1, then only the terms $\exp(\beta \tau_{q,r})$ with $\tau_{q,r} \sim 1$ matter. We recall that $\tau_{q,r} = T_{q,r}$ for first neighbours, otherwise $\tau_{q,r}$ is the item of the trust along the path from r to q. If τ_{q,a_k} 1, where a_k is a higher order neighbour of q connected by a path through the first neighbour a_n, then it must also be τ_{q,a_n} 1. Therefore we can replace the value of trust towards a_k with the one towards a_n and write the expected utility as:

$$E(f_q) = \frac{\sum_r \omega_{q,r} \exp(\beta T_{q,r})}{\sum_r \exp(\beta T_{q,r})} \tag{18}$$

The values of similarity $\omega_{q,r}$ and trust $T_{q,r}$ between two agents depend only on their respective preference profiles. In particular the dynamics of trust yields $T^{\star} = \omega/(1-\gamma)$. There is a finite number of combinations of profiles, and their probability of occurrence depends on the density of the network. These probabilities can be computed from the frequency of occurrence of each profile in the population (we have assumed that profiles are assigned randomly to the agents). Therefore, the probability of occurrence of each value of similarity ω, $\nu(\omega)$ is known and we can order the terms of the sum above by values of similarity rather than by index of agents:

$$E(f) = \frac{\sum_\omega \omega \exp(\frac{\beta\omega}{1-\gamma}) \nu(\omega)}{\sum_\omega \exp(\frac{\beta\omega}{1-\gamma})} \tag{19}$$

Each term $\exp(\frac{\beta\omega}{1-\gamma})\nu(\omega)$ is the probability of an agent choosing the recommendation from an other agent with a given similarity value ω multiplied by the probability that such a similarity value occurs among the agent q and the recommenders. This formula allows to predict the expected utility of the system as function of the density of the network and the distribution of the profiles of preferences among the agents. The formula does not take into account to what extent the agents have knowledge on the categories, as a population or individually. This will be the object of further work. In the regime of $N \gg L$, the empirical results confirm our analytical approach.

5 Conclusions

In our model, the preference experienced by an agent when consuming an item recommended by another agent can be more or less close to the preference communicated by the recommender. As commonly assumed in models of trust mechanisms [Montaner, López, & de la Rosa, 2002], trust of an agent towards another agent evolves in time as a discounted sum of the similarity between the preferences experienced by the agent and those communicated by the recommenders. In order to decide among several recommended items that match his request to the network, an agent chooses stochastically with a probability given by a logit function of the current trust towards each agent. This is a well-established way in economics to resolve the exploitation-exploration dilemma in networks of economic agents [Weisbuch, Kirman, & Herreiner, 2000]. Assuming preferences on items to be discrete random variables $\{\pm 1\}$, the model can be solved analytically with a mean-field approximation as we have shown.

In the computer simulations, we furthermore found the following results: For homogeneous preferences, the performance of the system is the same with trust and without trust. Indeed, if all agents have the same preferences, they will get the 'right' advice (i.e, with a high preference value) from any query.

For mildly heterogeneous preferences, trust matters and the system with trust yields better performance than the system without trust. In fact, any agent is able to find some other agents with similar preferences in his neighbourhood; the agent is then able to build up a trust relationship with them and consequently they can filter those recommendations which are trustworthy for him.

For highly heterogeneous preferences, the system with trust still performs better than the system without trust, but to a much lesser extent. Indeed, very few agents find other agents with similar preferences in their neighbourhood. Therefore, no trust relationships build up and the filtering of recommendations by trustworthiness is not effective.

In conclusion, this model allows to quantitatively analyse in which regimes of network density and heterogeneity of agent preferences a trust mechanism can effectively improve a recommendation system on a social network. In this model, the network is static; real social networks co-evolve in time with trust among agents. Therefore, this model is a initial step towards a more realistic model in view of possible experiments or implementations.

References

Abdul-Rahman, A., and Hailes, S. 2000. Supporting trust in virtual communities. In *Proceedings of the 33th Annual Hawaii International Conference on System Sciences*. IEEE Press.

Battiston, S., and Catanzaro, M. 2004. Statistical properties of board and director networks. *European Journal of Physics B* 38(345).

Battiston, S.; Bonabeau, E.; and Weisbuch, G. 2003. Decision making dynamics in corporate boards. *Physica A*.

Battiston, S.; Weisbuch, G.; and Bonabeau, E. 2003. Spread of decisions in the corporate board network. *Advances in Complex Systems* 6(4).

Blaze, M.; Feigenbaum, J.; and Lacy, J. 1996. Decentralized trust management. In *Proceedings of the IEEE Symposium on Security and Privacy*. IEEE Press.

Castelfranchi, C., and Falcone, R. 1998. Principles of trust for MAS: Cognitive anatomy, social importance, and quantification. In Demazeau, Y., ed., *Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS'98)*, 72–79. IEEE.

Gambetta, D. 2000. *Trust: Making and Breaking Cooperative Relations*. Electronic Edition, Oxford University.

Garlaschelli, D.; Battiston, S.; Castri, M.; Servedio, V. D. P.; and Caldarelli, G. 2003. The scale-free topology of market investments. *Physica A* 350(2-4):491–499.

Gray, E.; Seigneur, J.-M.; Chen, Y.; and Jensen, C. D. 2003. Trust propagation in small worlds. In *First International Conference on Trust Management*, 239–254.

Guha, R.; Kumar, R.; Raghavan, P.; and Tomkins, A. 2004. Propagation of trust and distrust. In *WWW 2004*, 403–412.

Marsh, S. 1994. *Formalising Trust as a Computational Concept*. Ph.D. Dissertation, University of Stirling.

Massa, P., and Bhattacharjee, B. 2004. Using trust in recommender systems: An experimental analysis. In *iTrust 2004*, 221–235.

Montaner, M.; López, B.; and de la Rosa, J. L. 2002. Developing trust in recommender agents. In *AAMAS 2002*, 304–305.

Montaner, M.; López, B.; and de la Rosa, J. L. 2003. A taxonomy of recommender agents on the internet. *Artificial Intelligence Review* 19(4):285–330.

Mui, L.; Mohtashemi, M.; and Halberstadt, A. 2002. A computational model of trust and reputation for e-businesses. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*. IEEE Press.

Newman. 2003. The structure and function of complex networks. *SIREV: SIAM Review* 45.

Palau, J.; Montaner, M.; López, B.; and de la Rosa, J. L. 2004. Collaboration analysis in recommender systems using social networks. In *CIA 2004*, 137–151.

Shih, Y.-Y., and Liu, D.-R. 2005. Hybrid recommendation approaches: Collaborative filtering via valuable content information. In *HICSS*.

Watts, D. J., and Strogatz, S. H. 1998. Collective dynamics of 'small-world' networks. *Nature* 393:440–442.

Weisbuch, G.; Kirman, A.; and Herreiner, D. 2000. Market organisation and trading relationships. *Economic Journal* 110(463):411–36.

Yu, B., and Singh, M. 2000. A social mechanism of reputation management in electronic communities. In *Proceedings of the 4th International Workshop on Cooperative Information Agents, The Future of Information Agents in Cyberspace*, 154–165. Springer-Verlag.

Zacharia, G., and Maes, P. 2000. Trust management through reputation mechanisms. *Applied Artificial Intelligence* 14:881–908.