Modelling Dynamic Forgetting in Distributed Information Systems

Nicolas Höning[1] and Martijn C. Schut[2]

[1] Centrum Wiskunde en Informatica (CWI), Amsterdam, Netherlands
[2] VU University, Department of Computer Science, Amsterdam, Netherlands

Abstract. We describe and model a new aspect in the design of distributed information systems. We build upon a previously described problem on the *micro-level*, which asks how quickly agents should discount (forget) their experience: If they cherish their memories, they can build their reports on larger data sets; if they discount quickly, they can respond well to change in their environment. Here, we argue that on the *macro-level*, where agents disseminate information, the coordination of these micro-level strategies of discounting can have significant consequences on the system performance if the environment is uncertain. In our proposed model, a referral network disseminates information about a disruptive environment (a service provider) to a risk-averse client agent, who uses this information to maximise his profit and then gives feedback into the referral system. We model two simple strategies to dynamically find better discounting factors, through central and decentral control. We show that with dynamic discounting rates, the system can become more reactive. We discuss interdependence of the system components in the light of differing discounting scenarios. In this work, we build on a certainty-based trust representation and operators for it in referral systems, developed by Josang [7] and Hang, Wang and Singh [13,2].

1 Introduction

In dynamic environments, information becomes less accurate over time – this is why forgetting exists, as a reflection of uncertainty in a changing world. In many distributed information systems that are modelled, the forgetting rate (in the literature, this is mostly referred to as the *discounting factor*) is set for all agents and never changes. But by definition, autonomous agents can forget (discount) old information at a rate of their choosing. Here, we propose a model to study this design aspect. We believe it is necessary to investigate the problem that the nature of these micro-level decisions can have significant consequences on the macro-level, affecting the ability of the system to accurately model the environment. This is certainly the case in sophisticated information systems, where information is not merely passed on and accumulated, but agents evaluate information before and after they pass it on. To our best knowledge, this problem has not been investigated so far.

In particular, we investigate the interactions between agents in so-called *referral networks*, in which trust forms the basis of cooperation and information sharing between agents. If such networks are employed in dynamic environments, then the agents in the networks have to react to changing information. This raises the question how each

P. Jędrzejowicz et al. (Eds.): KES-AMSTA 2010, Part II, LNAI 6071, pp. 252–261, 2010.
© Springer-Verlag Berlin Heidelberg 2010

agent must decide for itself what part of the information that it has received so far reflects the current situation best. We model these dynamics by a referral network which disseminates information about a disruptive environment (a service provider who sometimes changes his quality level from good to bad or vice versa) to a risk-averse client agent, who uses this information to maximise his profit and then gives feedback into the referral system.

Again more focused, for such referral networks, we look at the (distributed) *discounting* of information, i.e., agents in a referral network discount old information to correctly reflect the current situation, while trying to keep their evidence base (their "certainty") as large as possible (this constitutes a trade-off as both old and new information have their own value to the agent). For such discounting, one often employs a discounting factor with which to deprecate information (comparable to an interest rate on a loan).

This discounting factor affects the information directly, but indirectly it is also likely to affect other information in the system. For instance, the information is evaluated in the light of the trust of the recipient in the referrer before being passed on. The trust in the referrer itself should be subject to discounting (to become independent from malicious or badly informed referrers) and, in turn, is via feedback mechanisms evaluated on how accurate the referred information was. Here, we provide a starting point to model this seemingly complex design problem and investigate the behaviour of two simple strategies.

The contributions of this paper are twofold: firstly, we model a service provision setup in which distributed discounting in referral networks can be studied (here we extend existing work by Singh and colleagues [2,11,12,14] on certainty-based trust models for multi-agent systems). Secondly, we perform a number of computer simulation experiments, observing effects of distributed discounting strategies in different disruptiveness-scenarios.

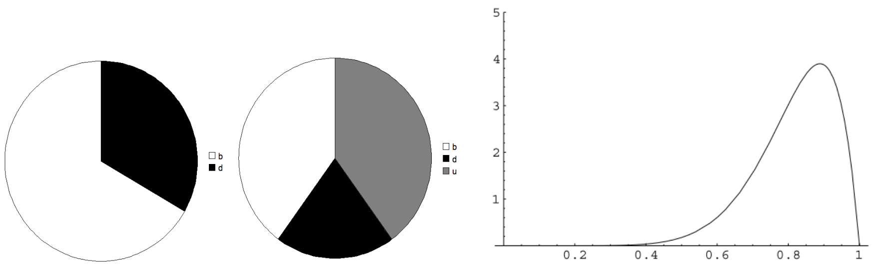
This paper is structured as follows. In the following section, we discuss relevant literature and position our work in the areas of certainty-based trust, referral networks and inter-temporal value of information (i.e., balancing the factors of recency and certainty in order to make good decisions). In Section 3 we present our model, including the referral network and the embedding of this networking between disruptive service provider and risk-averse client. In Section 4 we describe simulation experiments we performed on the system and analyse the outcomes of these experiments. We close in Section 5 with conclusions and pointers for future work.

2 Literature

In this section, we discuss related literature concerning certainty-based trust (including a complex representation of trust with an explicit notion of certainty), referral networks (especially the work of Singh and colleagues on the issue of how to propagate trust in such networks), and inter-temporal value of information (about how to effectively balance the recency and certainty of information).

2.1 Certainty-Based Trust

Trust is based on experience (e.g., with other agents or with an uncertain environment in general). The more experience a trustor collects about the trustee, the more accurate his



(a) Scalar trust vs Certainty-based trust (b) Probability distribution

Fig. 1. (a) Scalar trust vs certainty-based trust with uncertainty **u**. The ratio of belief **b** and disbelief **d** is equal in both charts. (b) An example probability distribution from [7], after 8 positive and 1 negative results.

opinion (also called a *trust report*) should become. For this, every trust representation needs to be accompanied with an update function, so that new experience somehow alters the existing opinion. A trust report can then be seen as a compressed history of experience. When agents translate their experience history into their trust reports, they compress it and thereby need to give up some of the inherent information.

Various researchers have investigated how to do such compression effectively, see [6,7,8]. While trust can be represented as a simple scalar (e.g. $\in [0, 1]$, it may be more effectively to use a slightly more complex representation in which the *certainty* is represented explicitly. From a two-fold *evidence space*, where good and bad experiences r and s are collected, we compute a three-fold *belief space*, where the third fold takes uncertainty into account. See Figure 1(a) for an illustration. Initially, without any experience, everything is uncertain. As experience is added in evidence space, uncertainty decreases in the belief space and certainty (which consists of belief plus disbelief) starts to get its share. This helps to distinguish on how much experience a trust report is actually based. By calculating over evidence space, the certainty $c \in [0, 1]$ can be derived from the integral of a probability density function (see Figure 1(b)), which we do according to [12]. Then, the probability of a good outcome $\alpha = \frac{r}{r+s} \in [0, 1]$ defines how certainty is divided into belief and disbelief.

2.2 Referral Networks

When agents make use of the trust reports provided by other agents, they form a referral network. A referral network is an information dissemination system consisting of autonomous and possibly self-interested agents. A referral is provided by agent B to agent A by giving A a report about his (B's) trust in C. The path from A to B to C is called a referral path (and could in principle also extend over several other agents). Agents who have no direct experience can only refer to other agents and are called *referrers*, while agents who can provide their direct experience are called *witnesses*.

Referral networks have been studied widely (e.g. [11,1]) and are a state-of-the-art technique to model information dissemination in multi-agent systems. Hereby, direct experience becomes indirect knowledge to its recipients. As such, trust (direct experience) becomes reputation (indirect knowledge).

For the precise formalisation of operators used in referral networks employing a certainty-based trust representation, see [9] for the path operators aggregation and concatenation (which was reconsidered in [14]) and [2] for the update operator (which was reconsidered by us in [4]). The update operator updates an agents trust in a referrer by considering two other trust reports: the referred opinion and the agents actual (ex post) opinion.

2.3 Discounting

The most basic approach to incorporate recency in one's memory is that old experience should be accounted for less, since circumstances might change over time. Discounting mostly happens with a discounting factor. For instance, here is the formula used by [6] for a simple scalar trust representation:

$$g_d(tv, ev) = d * tv + (1 - d) * ev$$

where g_d is the update function which updates the trust value $tv \in [-1, 1]$ in light of a new experience $ev \in [-1, 1]$, using a discount factor $d \in [0, 1]$. The existing tv is also discounted by d to normalise the result of g_d in the range $[-1, 1]$.

For the addition of experience in evidence space, we use a similar update function (following [2]), for r and s. Here is the update of existing r with new experience r':

$$r = r' + (1 - \beta) * r$$

Note that the choice of the discounting factor β is crucial and should depend on the situation at hand. Hang [2] experimented with varying static values for β in different situations, but we have not come across any research in which agents use dynamic values for β or choose their values themselves.

2.4 Recency and Certainty

Agents generally strive for more information in order to learn about their environment. When they are more certain in their actions, they can take higher risks and maximise their profits. The trust model described in section 2.1 enables agents to build up certainty over time, as agents interact. However, when stored experience gets forgotten, the certainty also decreases. Meanwhile, agents will have more accurate opinions if they base their reports on more recent information. As agents in a trust system try to maximise both of these values, a trade-off situation occurs: How much should old experience be discounted in order to to benefit from both certainty and recency?

Several researchers have noticed this issue (while none that we know of have focused their analysis on it). For instance, [10] note the recency/certainty trade-off when agents produce less failures with smaller history windows, but become more vulnerable to malicious behaviour which exploits small history windows. Also, in [5] a simple reputation system is modelled, in which two firms and many customers are present. Customers discount their memories of direct experience and firms choose how to invest in their next intended quality level, based on their current reputation. They show that discounting is a necessary condition for equilibrium points to be reached but that uncertainty makes it unlikely that the system reaches any equilibrium point.

To accumulate certainty is generally rewarding, unless the environment becomes too uncertain for accumulated memory to yield any valuable predictions. In that case, a focus on certainty can even be harmful as the agent(s) will be too slow to react to a disruption. In general, we do not need to speak of recency *versus* certainty: to some degree, a system should manage to have both.

3 Model

We model a referral network - a system in which a *client* needs to learn about a *service provider* and relies on referral agents for this. This network involves *witnesses* who have direct experience with the service provider and *referrers* who only deal with indirect experience. The client knows all referrers and can access them to ask for information and give them feedback. Each referrer has access to some other referrers, while only some have access to a witness. We strictly separate referral agents from witness agents for simplicity (such that an agent can only have one role).

A service provider provides a service, the quality of which can change between being good or bad. The client asks several (randomly selected) referrers he knows for information about the service quality. On each of those referral paths, he gets referred to another referrer until he finds a witness. Then, information travels from the service provider as first-hand experience to witnesses. It then becomes reputation when the client makes use of it. Note that he does not use the witnesses opinion as-is, but he incorporates the referrers opinions about the agent (referrer or witness) they referred to. Thus, the client needs to concatenate the trust values along the paths. Then, he aggregates the results from all paths into one final trust report, which represents the networks opinion. See Figure 2 for a broad overview over information flow. Note that also the client has direct experience. In this system, this is solely used to evaluate the reputation information he receives in order to give feedback.

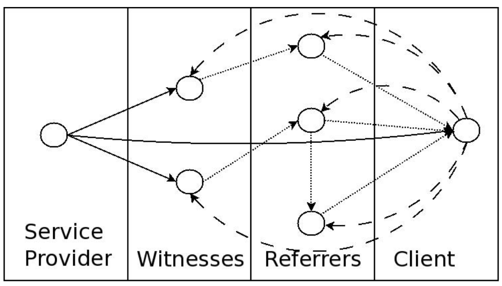


Fig. 2. The referral network in our system, where solid arrows indicate first-hand experience, dotted arrows indicate referrals, and dashed arrows indicate feedback. The four types of agents are explained in the text.

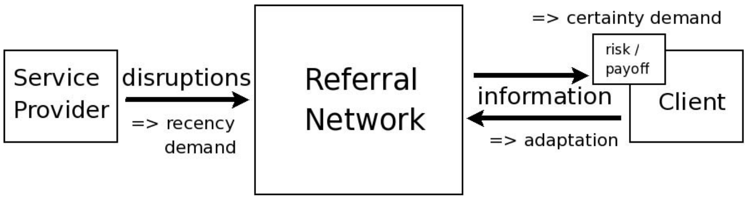


Fig. 3. Functional overview of the service provision system

We use the notion of service provision to evaluate the effectiveness of information dissemination. In particular, a client on one end of the referral network is about to use a service on the other end of the referral network and uses the referred trust information to calculate the risk he is prepared to take.

We view the service quality as an external input to the system and the clients utility as an output. Furthermore, the client might provide evaluations and feedback as input into the system. Figure 3 shows a functional system overview, modelling the referral system (referrers and witnesses) as the system under consideration, with input and output connected to the client and the service provider.

The general setting we want to investigate with this system is an uncertain, potentially disruptive and non-deterministic environment, in which agents have a *demand for recency*. For this, the disruptiveness of the service needs to be a controlled variable, alike [3]. We implement a service provider who initially provides good service. The $disruptiveness$ denotes the probability of service level disruptions between rounds (a change from good to bad quality or vice versa).

On the other end of the system, the client implementation should model that there is a *demand for certainty*. The assumption that certainty is useful has to be set into practice. Therefore, we propose that the client can accumulate a payoff, representing the usefulness of the system (to provide accurate information). The client takes a risk each round. He puts a stake into the interaction with the service provider. If the service is good, his return is twice the stake he put in, otherwise he gets nothing back.

In our model, the stake is the belief of the referral network in a good service, minus the disbelief. Thus, the stakes upper bound is the belief (if there is no disbelief) and the lower bound is zero (if disbelief is higher than belief). Note that stakes would rise with certainty, assuming the relation between belief and disbelief stays constant.

Finally, we make the agents act *adaptively*. This means that they control their own views of the world (the trust in their neighbours and – in one scenario – their own discounting behaviour). To inform the adaptation process, they need to receive feedback from the client for each of their referrals. With this, they can use the update operator (see Section 2.2) to update the trust in their neighbours and various mechanisms (to be explained in Section 4.1) to update the discounting factor β.

4 Experiment

We have observed in preliminary experiments that the disruptiveness of the environment significantly affects the effectiveness of static discounting strategies (e.g. low values for

β (the discounting parameter, explained in section 2.3) are hurtful in highly disruptive scenarios) and that this system is stable in the light of malicious witnesses and referrers [4]. We turn now to an experiment in which values for β are dynamically determined in various ways[1].

4.1 Setup

As dependent variables, we measure the client payoff, the certainty reported by the referral network, the clients stake and the values of β in referrers and witnesses. The experiment ran for 1000 iterations and was conducted 30 times, with a population of 18 agents (10 referrers, 8 witnesses). We show the averages along with the sample standard deviations.

In each run, the network structure was newly configured - agents were randomly connected such that the client knew four of the referrers, each referrer knew three other referrers and each witness was known by one referrer. The client asks all referrers for opinions (and then concatenates and aggregates their reports to make a decision), while referrers only ask one witness if they know one, or one random referrer otherwise. Each round, the client updates his feedback in the referrers and the referrers get the clients feedback, so they can update their trust in the agent they referred to.

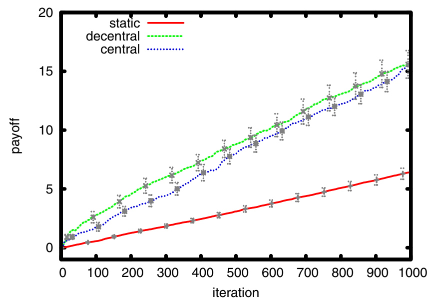
As independent variables, we control the probability of service disruptions (we use 0.05 and 0.33) and the agents adaptation of their discounting factor β in response to the accuracy of their referrals. We always let all agents begin with a value for β drawn randomly from $[0.1, 0.9]$. In the *static* scenario, all β values will stay fixed during the runs. We design two dynamic scenarios:

1. The *centrally* controlled scenario, in which the client defines the new β for all agents, according to how accurately the network described his experience.
2. The *decentrally* controlled scenario, in which each referrer and witness uses the feedback from the client to adjust his β himself. In order to do this, they keep a trust in their own history of accuracy t_o, according to which they adjust their discounting: $\beta = 1 - t_o.\alpha$ (α denotes the probability of a good outcome, see Section 2.1). When they learn through feedback that their recent referrals were accurate/misleading, they will thus lower/raise their β. In this experiment, we set the β of t_o to 0.1.

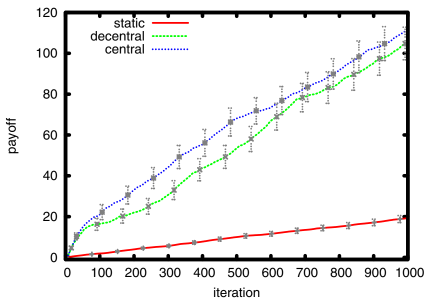
4.2 Results and Discussion

In Figures 4(a) and 4(b), we see that payoff-wise, the dynamic scenarios fare much better than the static scenario. As could be expected, it is beneficial to adjust β values according to the situation at hand. When the service quality changes, agents raise β in order to discount old information that has just become less accurate. This way, stakes rise faster in times of good service and drop faster in times of bad service, so big losses are avoided. Note also that for a disruptiveness of 0.33, much less payoff can be made than with 0.05, since much less certainty can build up in the trust reports.

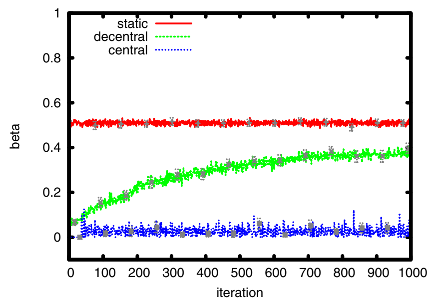
[1] The program code used to run these experiments can be openly accessed at http://subversion.assembla.com/svn/trustcertprop/tags/kes_paper/



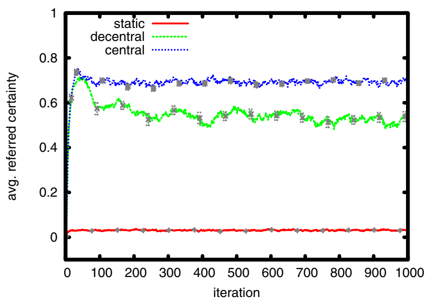
(a) Avg. payoffs for disruptiveness = 0.33



(b) Avg. payoffs for disruptiveness = 0.05



(c) Avg. β in referrers and witnesses asked in this round

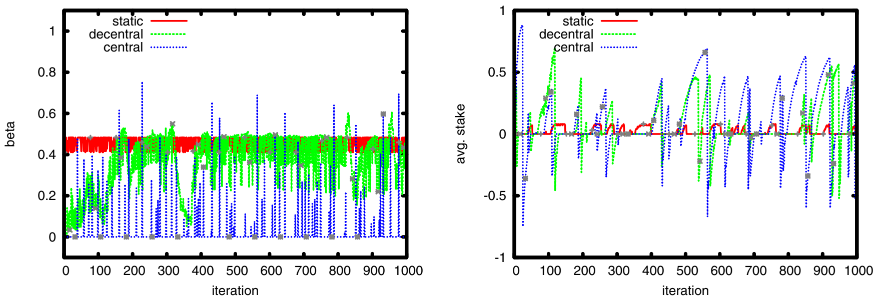


(d) Avg. certainty of the referral net's trust report

Fig. 4. Results of all runs

Interestingly, the performance between both dynamic scenarios does not differ significantly. Let us consider the inner workings a bit closer. In Figure 4(c), we see that the decentral scenario uses (on average) higher values for β. See Figure 5(a) for a the development of beta values in a sample run. While the values in the central scenario spike rapidly and then quickly return to near-zero, values in the decentral scenario are more diverse and higher. This is due to their $t_o.\alpha$ being lower, since their β is not aligned with the β of the client, whose feedback determines t_o. Thus, lower certainty is accumulated by referrers and witnesses (see Figure 4(d)). Note also that in the static scenario, very little certainty accumulates due to inflexibility of β.

In spite of higher values for β, the referral network in the decentralised scenario generally manages to give equally valuable signals for the investment behaviour of the client. See Figure 5(b) for stakes invested in a single run. Recall that the stake is calculated by $max(0, belief - disbelief)$. The certainty in the trust report of the referral network is $belief + disbelief$, so if the stakes are (roughly) equally high with lower certainty, this most likely means that in the decentralised scenario, agents refer stronger opinions (a higher value of $belief - disbelief$, which normally happens with higher values for β).



(a) Avg. β in referrers and witnesses asked in this round

(b) stakes earned/lost by the client

Fig. 5. Results of a single run

5 Conclusions

In this paper, we approach a trade-off between certainty and recency which each information system faces when its environment is uncertain. We develop previous research in certainty-based trust in referral networks into a model where this trade-off can be studied. Then we run initial experiments in which the agents dynamically find better discounting factors.

In our model, a referral network disseminates information from a disruptive environment (a service provider agent) to a risk-averse client agent, who uses this information to maximise his profit and then gives feedback into the referral system. We model two simple strategies to dynamically find better discounting factors (β) in a distributed multi-agent context. In the first scenario, the client agent chooses one β centrally. In the second scenario, all agents in the referral network choose their next β themselves, based on the clients feedback about their last referral. Both strategies are significantly better in maximising the clients pay-off than static values for β, though they seem to differ much in their behaviour.

In future work, the behaviours of these two strategies will be investigated closer in order to discover interaction effects on referral paths, and surely there can be new ideas on how to dynamically and decentrally compute discounting factors. Furthermore, one could make the agents more complex in various ways, for instance by letting them reason about future dynamics of the world. Another direction could be to investigate how incentives to agents to behave self-interested affects the recency/certainty trade-off. To model this, the client agent could be required to share parts of its generated utility as payments for referrals.

References

1. Ding, L., Kolari, P., Ganjugunte, S., Finin, T., Joshi, A.: Modeling and evaluating trust network inference. In: Seventh International Workshop on Trust in Agent Societies (2005)
2. Hang, C., Wang, Y., Singh, M.P.: An adaptive probabilistic trust model and its evaluation. In: Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems, vol. 3, pp. 1485–1488 (2008)

3. Harrington Jr., J.: The social selection of flexible and rigid agents. American Economic Review (1998)
4. Höning, N.: Discounting experience in referral networks. Master's thesis, VU University, Amsterdam (2009)
5. Huberman, B., Wu, F.: The dynamics of reputations. Computing in Economics and Finance (2003)
6. Jonker, C.M., Treur, J.: Formal Analysis of Models for the Dynamics of Trust Based on Experiences. In: Garijo, F.J., Boman, M. (eds.) MAAMAW 1999. LNCS, vol. 1647, pp. 221–231. Springer, Heidelberg (1999)
7. Jøsang, A.: A subjective metric of authentication. Lecture notes in computer science, pp. 329–344 (1999)
8. Jøsang, A., Hayward, R., Pope, S.: Trust network analysis with subjective logic. In: Proceedings of the 29th Australasian Computer Science (2006)
9. Jøsang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. Decision Support Systems (2007)
10. Keung, S.N., Griffiths, N.: Using recency and relevance to assess trust and reputation. In: Proceedings of AISB 2008 Symposium on Behaviour Regulation (2008)
11. Singh, M.P., Yolum, P.: Emergent properties of referral systems. In: 2nd International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 592–597 (2003)
12. Wang, Y.: Evidence-Based Trust in Distributed Agent Systems. PhD thesis, North Carolina State University, Raleigh (2009)
13. Wang, Y., Singh, M.P.: Trust representation and aggregation in a distributed agent system. Science (2006)
14. Wang, Y., Singh, M.P.: Formal trust model for multiagent systems. In: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI 2007), pp. 1551–1556 (2007)