

Declarative and Numerical Analysis of Edge Creation Process in Trust-Based Social Networks

Babak Khosravifar¹, Jamal Bentahar², and Maziar Gomrokchi³

¹ Department of Electrical and Computer Engineering, ² Concordia Institute for Information System Engineering, ³ Department of Computer Science,
Concordia University, Montreal, Canada
`b_khosr@encs.concordia.ca`, `bentahar@ciise.concordia.ca`,
`m_gomrok@encs.concordia.ca`

Abstract. Online social networks are enjoying drastic increase in their population and connectivity. One of the fundamental issues in these networks is trust, which is an essential factor in quality of the connections among diverse nodes in the network. To address the efficiency in the interactions among nodes, we propose in this paper a trust-based architecture applicable to maintain interactions in multi-agent-based social networks. We provide a detailed discussion over the network formation by taking into account the edge creation factors classified as homophily, confounding and influence. We systematically inspire different involving factors to observe evolution of trust-based interconnections in a microscopic manner. We also provide declarative and numerical analysis of the proposed model and its assessment and discuss the system implementation, along with simulations obtained from a number of executions compared with the broadly known frameworks.

Keywords: Trust establishment, edge creation, agent communication, social networks.

1 Introduction

During the last ten years, online social networks have been drastically enlarged. Facebook, Flickr, Yahoo! Answers are among very popular social networks that are gaining a very high traffic in terms of the users and their connectivity. In general, the impact of the features of these networks and analysis on how they form the behavior of the users have been of a great interest during the very recent years. A number of theoretical and empirical works have been proposed analyzing the users' behavior in forming the connection among them. For example, the analysis on the edge creation process between network nodes (participants), which is related to the sociality of a node, led to observe the distribution of a heavy-traffic degree of popular nodes [6,10]. In [1], the authors address the source of the correlation among agents that led them to extend their activity and create edges. In [2], the correlation between agents are analyzed in an online large scale network. In fact, the relation among the agents that just joined the

network and the agents that are already in the network is discussed. Clearly, all these proposals investigate the way the networks are formed and enlarged. However, the reasons behind the edge creations and extensions are not specifically discussed.

In this paper, we analyze the issue of edge creation from a different perspective over the social correlation among agents using a combination of declarative and numerical techniques. In fact, trust is the main issue agents consider when decide to create edges connecting them within the network. In edge creation process, we analyze diverse impacts upon the trust that is already established between two nodes and generalize the trust concept to analyze the socialization of agents that use different trust evaluation systems. There are some proposals in the literature developing frameworks to establish trust among agents. The purpose of this paper is not to develop a new trust framework, but to analyze how the trust can affect the status of an agent in a social network, which is captured by a set of decision rules in the framework. We first discuss the trust evaluation method and upon that, analyze the connectivity among agents in a microscopic approach. Some trust models in the literature consider the direct interaction of two parties [8,15,16]. Some models also rely, to some extent, on the suggested ratings provided by other agents [9,12,14]; and some others also consider the suggested ratings of the agent being evaluated [4,8]. Since agents are self-interested, it is hard to analyze an agent's likely behavior based on previous direct interactions given the fact that the collected information from other agents may be non-reliable and could lead to a non-accurate trust assessment. So far, these frameworks do not act properly if selfish agents tend to change their behaviors. Therefore, agents do not properly initiate a social activity in the sense that they cannot maintain a strong control on the environment.

In this paper, we use the model we proposed in [11] and discuss the social network related parameters (classified in [1]) such as homophily, confounding and influence on the edge creation process of the distributed agents. Homophily refers to the tendency for agents to have ties with agent who are similar to themselves. Confounding refers to the external influence where external factors correlate with the event that two agents become connected. For example, "two friends are likely to live in the same city, and therefore to post pictures of the same landmarks in an online photo sharing system" [1]. Influence is a parameter for the extent to which an agent is prompted to initiate a connection with another agent caused by an adjacent agent.

The objective of discussing these parameters is to elaborate the impacts that a proper trust adjustment framework has on the extension of agents' connectivity. Considering the cost that agents pay for edge creation, the accuracy on suitable extensions are crucial. In the proposed model, we provide an efficient assessment process in a twofold contribution. In the first contribution, agents evaluate the trust of other agents by combining their direct and indirect trust. In direct trust, the history of interactions is considered as a measure of honesty. In indirect trust, the suggested rates through some consulting agents are

considered in the measurement of trust. In the second contribution, the evaluator agent updates its belief set considering the difference/similarity between the proposed rates and the actual trustworthiness of the evaluated agent captured by its actual behavior. The update (so-called maintenance) considers the evaluated agent together with the consulting agents that have been suggested in support of the agent being evaluated. By updating the trust values, agents would recognize the adjacent agents that are worth to extend the connection with. On the other hand, agents would stay far from bad agents in terms of social connectivity. Doing so, the agents equipped with the maintenance framework, gradually recognize the more reliable interacting agents and thus, can quickly propagate the recent changes in the environment. The edge creations are done towards the active and accurate communities. Therefore, more efficient social correlation is formed among interacting agents.

The remainder of this paper is organized as follows. In Section 2, we briefly define our proposed framework as comprehensive trust assessment process, which is composed of evaluation and maintenance process. In Section 3, we define the social network parameters and environment where the interactions are taking place. In Section 4, we analyze and discuss these parameters in an experimental setting. We represent the testbed and compare our model results with two well-known trust models in terms of efficiency in trust assessment. Finally, Section 5 concludes the paper.

2 Trust Evaluation Approach

2.1 General Background

In this section, we combine declarative and numerical techniques to formalize trust and its assessment between interacting parties in the social network. As illustrative example, we consider a social network of customers and providers of some services used for service selection. In this paper, we focus on the trust assessment formulation and predefined rules that direct the edge extensions of agents interacting in the social network. Details about the used trust model are provided in [11]. To characterize the relationship between a trustor agent Ag_a (e.g. a customer) and a trustee agent Ag_b (e.g. a provider), three elements are used [3]: 1) how much the trustor agent trusts the trustee: $Tr_{Ag_a}^{Ag_b}$; 2) the number of past interactions: $NI_{Ag_a}^{Ag_b}$ or business transactions: $NT_{Ag_a}^{Ag_b}$; and 3) the time recency of the last transactions: $TiR_{Ag_a}^{Ag_b}$. Formally, we define a social network for service selection as follows:

Definition 1 (Social Network). A social network SN for service selection is a tuple $\langle C, P, \text{---}_{tc}, \text{---}_{tp} \rangle$ where C is a set of customers, P is a set of providers, $\text{---}_{tc} \subseteq C \times \mathbb{R}^3 \times C$ is a ternary relation (for labelled edges linking customers) and $\text{---}_{tp} \subseteq C \times \mathbb{R}^3 \times P$ is a ternary relation (for labelled edges linking customers to providers).

To simplify the notation for the labelled edges, if $c_i, c_j \in C$ and $v \in \mathbb{R}^3$, then $(c_i, v, c_j) \in \text{---}_{\text{ec}}$ is written as $R_{cc}(c_i, v, c_j)$. Likewise, we write $R_{cp}(c_i, v, p_k)$ instead of $(c_i, v, p_k) \in \text{---}_{\text{ep}}$ where $p_k \in P$. Our social network for service selection has two types of nodes: type 1 for customers and type 2 for providers and two types of edges: type 1 for edges between customers and type 2 for edges linking customers to providers. The edges of type 1 represent friendship relations in the network, while edges of type 2 capture business relationships. The existence of an edge of type 1 $R_{cc}(c_i, v, c_j)$ means that c_i knows (is friend of) c_j such that: $v = (Tr_{c_i}^{c_j}, NI_{c_i}^{c_j}, TiR_{c_i}^{c_j})$. The existence of an edge of type 2 $R_{cp}(c_i, v, p_k)$ means that c_i had transactions with p_k such that: $v = (Tr_{c_i}^{p_k}, NT_{c_i}^{p_k}, TiR_{c_i}^{p_k})$.

In general, each customer agent c_i is linked to a set of customers it knows and a set of providers it has interacted with in the past. A link (an edge) between two customers c_i and c_j is added to the social network when the number of interactions between them is large enough. In the same way, a link (an edge) between a customer c_i and a provider p_k is added to the social network when the number of transactions between them is large enough. The following two Prolog-like rules (having the form: $Head \leftarrow Body$: if $Body$ then $Head$) are used by c_i as decision rules to decide about adding links to the social networks where μ_1 and μ_2 are two predefined thresholds:

$$R_{cc}(c_i, v, c_j) \leftarrow v = (Tr_{c_i}^{c_j}, NI_{c_i}^{c_j}, TiR_{c_i}^{c_j}) \wedge NI_{c_i}^{c_j} > \mu_1$$

$$R_{cp}(c_i, v, p_k) \leftarrow v = (Tr_{c_i}^{p_k}, NT_{c_i}^{p_k}, TiR_{c_i}^{p_k}) \wedge NT_{c_i}^{p_k} > \mu_2$$

We note that there is no edges in this social network between providers. This does not mean that there is no social link between providers, but only the existing links (which could be collaborations or competitions) are not used in our framework. In fact, links between providers could be used to share information regarding clients trust. However, sharing such market information in a competitive setting requires incentives and other considerations such as coalition formation. Game theory and mechanism design tools could be used to analyze these considerations. However, this aspect is out of the scope of this paper.

A social link between two customers c_i and c_j (denoted by $SL(c_i, c_j)$) exists either because there is a link (an edge) between c_i and c_j ($R_{cc}(c_i, v, c_j)$) or because c_i is linked via an edge to another customer c_x , which is socially linked to c_j via a social link $SL(c_x, c_j)$. This aspect is specified using the following Prolog-like recursive rules:

$$SL(c_i, c_j) \leftarrow R_{cc}(c_i, v, c_j)$$

$$SL(c_i, c_j) \leftarrow R_{cc}(c_i, v', c_x) \wedge SL(c_x, c_j)$$

In the same way, we specify the social link between a customer c_i and a provider p_k as follows:

$$SL(c_i, p_k) \leftarrow R_{cp}(c_i, v, p_k)$$

$$SL(c_i, p_k) \leftarrow R_{cc}(c_i, v', c_x) \wedge SL(c_x, p_k)$$

2.2 Direct Trust (*DTr*)

Let $\mathcal{T}ran_{c_i}^{p_k}$ be the set of transactions between a customer c_i and a provider p_k . The direct evaluation of p_k by c_i is based on the ratings c_i gave to p_k for each past transaction ($r_l \in \mathcal{T}ran_{c_i}^{p_k}$) combined with the importance of that interaction (λ_l) and its time recency. Let n be the number of total transactions between c_i and p_k ($n = NT_{c_i}^{p_k} = |\mathcal{T}ran_{c_i}^{p_k}|$), equation 1 gives the formula to compute this evaluation.

$$DTr_{c_i}^{p_k} = \frac{\sum_{l=1}^n (\lambda_l \cdot TiR_{c_i}^{p_k} \cdot r_l)}{\sum_{l=1}^n (\lambda_l \cdot TiR_{c_i}^{p_k})} \quad (1)$$

The direct interaction is considered reliable if the history of transactions is strong enough ($n > \mu_2$). If not, the evaluation should be done through consulting with some other agents. We refer to this evaluation as indirect evaluation $ITR_{c_i}^{p_k}$. In fact, if the time recency of the transactions is less than a predefined threshold μ_3 , the transactions reflect old behaviors and thus may not reveal the accurate information. Likewise, if the number of transactions is not high enough to reflect a strong history, the evaluating agent could not rely on that. The following rule is a decision rule for c_i specifying the pre-conditions of using indirect trust denoted by $Use(ITR_{c_i}^{p_k})$:

$$Use(ITR_{c_i}^{p_k}) \leftarrow \neg R_{cp}(c_i, v, p_k) \vee TiR_{c_i}^{p_k} < \mu_3$$

2.3 Indirect Trust (*ITr*)

To perform the indirect evaluation, the customer c_i solicits information about the provider p_k from other customers, called consulting customers (denoted by the set \mathcal{T}_{c_i}), such that for all $c_j \in \mathcal{T}_{c_i}$ there is an edge $R_{cc}(c_i, v, c_j)$ in the social network. An agent c_j is added to \mathcal{T}_{c_i} if the size of this set is less than a maximum size μ_4 and the overall trust value of c_j ($\alpha Tr_{c_i}^{c_j}$) is greater than a threshold μ_5 where $\alpha Tr_{c_i}^{c_j} = DTr_{c_i}^{c_j} \cdot NI_{c_i}^{c_j} \cdot TiR_{c_i}^{c_j}$. Thus, agents are added in \mathcal{T}_{c_i} in the sense that the evaluating agent can rely on their provided information in support of the provider that is being evaluated. The following is the decision rule used to update \mathcal{T}_{c_i} :

$$\mathcal{T}_{c_i} = \mathcal{T}_{c_i} \cup \{c_j\} \leftarrow |\mathcal{T}_{c_i}| < \mu_4 \wedge \alpha Tr_{c_i}^{c_j} > \mu_5$$

To communicate and exchange trust information, agents use messages defined as follows:

Definition 2. A communication message is a tuple $\langle \alpha, \beta, c_i, c_j, M, t \rangle$, where α ($\alpha \in \{Req, Rep\}$) indicates whether it is a request or a reply communication message, β ($\beta \in \{Inf, Refuse, Not\ Have\}$) represents the type of the message as requesting information in case of initiating the communication (*Inf*), refusing to reveal information (*Refuse*), or not having the information in case of replying to a request message (*Not Have*). Agents c_i and c_j are respectively the sender and receiver of the message, M is the content of the message and finally t is the time at which the message is sent.

In order to obtain a trust value of a provider p_k ($Trust(p_k)$), the evaluator agent c_i initiates a communication message to get the information from the consulting agents. The reply from a consulting agent c_j consists of providing the trust value ($Inf(p_k)$), or informing that it does not have the required information ($NotHave$), or refusing to answer ($Refuse$). Formally, these possibilities are represented by the following rules:

$$\langle Rep, Inf, c_j, c_i, Inf(p_k), t_1 \rangle \leftarrow \langle Req, Inf, c_i, c_j, Trust(p_k), t_0 \rangle$$

$$\langle Rep, NotHave, c_j, c_i, *, t_1 \rangle \leftarrow \langle Req, Inf, c_i, c_j, Trust(p_k), t_0 \rangle$$

$$\langle Rep, Refuse, c_j, c_i, *, t_1 \rangle \leftarrow \langle Req, Inf, c_i, c_j, Trust(p_k), t_0 \rangle$$

Proposition 1. Assume c_j is collaborating with c_i . c_j will reply by $Inf(p_k)$ using the first rule iff $SL(c_j, p_k)$.

The consulting agents are also subject to check for their trust values, and indeed, the more the consulting agent is trustworthy, the more the evaluating agent can rely on the provided rating. The equation computing the indirect estimation is given by equation 2.

$$ITr_{c_i}^{p_k} = \frac{\sum_{c_j \in \mathcal{T}_{c_i}} \alpha Tr_{c_i}^{c_j} \cdot DTr_{c_j}^{p_k} \cdot TiR_{c_j}^{p_k} \cdot NT_{c_j}^{p_k}}{\sum_{c_j \in \mathcal{T}_{c_i}} \alpha Tr_{c_i}^{c_j} \cdot TiR_{c_j}^{p_k} \cdot NT_{c_j}^{p_k}} \quad (2)$$

2.4 Total Trust (Tr)

To compute $Tr_{c_i}^{p_k}$, the direct and indirect evaluations are combined according to their proportional importance. The idea is that the customer relies, to some extent, on its own history (direct trust evaluation) and on consulting with its network (indirect trust evaluation). This merging method considers the proportional relevance of each trust assessment, rather than treating them separately. To this end, c_i assigns a contribution value for the trust assessment method (ω for direct trust evaluation and $1 - \omega$ for indirect trust evaluation when $\omega < 1$). The value ω is obtained from equation 3.

$$\omega = \frac{\log(DTr_{c_i}^{p_k} \cdot NT_{c_i}^{p_k} \cdot TiR_{c_i}^{p_k})}{\sum_{c_j \in \mathcal{T}_{c_i}} \log(DTr_{c_i}^{c_j} \cdot NT_{c_i}^{c_j} \cdot TiR_{c_i}^{c_k})} \quad (3)$$

This value could exceed 1 in the case that the history is more informative than contribution of others. Basically, the contribution of each approach in the evaluation of p_k is defined regarding to: (1) how informative the history is in terms of the number of direct transactions between c_i and p_k ($NT_{c_i}^{p_k}$) and their time recency ($TiR_{c_i}^{p_k}$); and (2) how informative and reliable the consulting customers are from c_i 's point of view ($DTr_{c_i}^{c_j}$). Therefore, consultation with other agents is less considered if the history represents a comparatively higher value for ω ,

which reflects lower uncertainty. Respecting the contribution percentage of the trust assessments, c_i computes the trust value for p_k using the following rules:

$$Tr_{c_i}^{p_k} = \omega \cdot DTr_{c_i}^{p_k} + (1 - \omega) \cdot ITr_{c_i}^{p_k} \leftarrow \omega < 1$$

$$Tr_{c_i}^{p_k} = DTr_{c_i}^{p_k} \leftarrow \omega \geq 1$$

Generally, the merging method is used to obtain the most accurate trust assessment. According to the following rule, the customer agent c_i would initiate the transaction $Transact(c_i, p_k)$ with p_k if the evaluated trust is high enough (threshold μ_6), which means that the customer agent can expect a high quality of service.

$$Transact(c_i, p_k) \leftarrow Tr_{c_i}^{p_k} > \mu_6$$

2.5 Maintenance

After performing the transaction, customer c_i analyzes the quality of the received service regarding to what is expected (i.e. the evaluated trust $Tr_{c_i}^{p_k}$) and what is actually performed (so-called observed trust value $\widehat{Tr}_{c_i}^{p_k}$). To this end, an adjustment trust evaluation should be performed. When c_i decides, based on the previous rule, to transact with p_k , the number of transactions is incremented. Then the observed value is checked with the expected trust value. The corresponding update is applied on the assessed trust value depending on the difference between the observed and the evaluated values. The following rules specify the maintenance process where μ_7 is a predefined threshold and the value β is a small value in the sense that $1 + \beta$ reflects an increase and $1 - \beta$ reflects a decrease in the current value.

$$NT_{c_i}^{p_k} = NT_{c_i}^{p_k} + 1 \leftarrow Transact(c_i, p_k)$$

$$Tr_{c_i}^{p_k} = Tr_{c_i}^{p_k} \times (1 + \beta) \leftarrow |\widehat{Tr}_{c_i}^{p_k} - Tr_{c_i}^{p_k}| < \mu_7$$

$$Tr_{c_i}^{p_k} = Tr_{c_i}^{p_k} \times (1 - \beta) \leftarrow |\widehat{Tr}_{c_i}^{p_k} - Tr_{c_i}^{p_k}| \geq \mu_7$$

In general, the idea is to learn from gained experiences in the sense that observing the actual value, the agent that performed the evaluation and consulted with couple of other agents can adjust its trust in them regarding to the accuracy of information they provided. To this end, the consulting agents that provided bad trust values, which are far from the observed one, will be removed from the list of potential witnesses in the future. Likewise, the agents that reveal accurate information would be considered more trustworthy than before and would be potentially consulted in future. The evaluating agent c_i would check for each consulting agent the suggested trust value with the observed actual value (checking the difference with the threshold μ_8), and consequently updates the corresponding values.

$$\begin{aligned}
Tr_{c_i}^{c_j} &= Tr_{c_i}^{c_j} \times (1 + \beta) \leftarrow |\widehat{Tr}_{c_i}^{p_k} - Tr_{c_j}^{p_k}| < \mu_8 \\
Tr_{c_i}^{c_j} &= Tr_{c_i}^{c_j} \times (1 - \beta) \leftarrow |\widehat{Tr}_{c_i}^{p_k} - Tr_{c_j}^{p_k}| \geq \mu_8 \\
\mathcal{T}_{c_i} &= \mathcal{T}_{c_i} \setminus \{c_j\} \leftarrow |\widehat{Tr}_{c_i}^{p_k} - Tr_{c_j}^{p_k}| \geq \mu_8
\end{aligned}$$

In addition, over the recent interactions, high quality providers are recognized and thus distributed over the adjacent agents in the network. In general, using the maintenance process (for full description of algorithms, see [11]), correlated agents could increase their rate of influence to one another, which eventually would approach to a more active social network. This can be represented by an optimization problem as shown in equation 4. The minimization problem is actually inspired by the fact that each time the maintenance is progressed, there is an actual value to compare with the previously suggested trust values. To this end, the evaluating agent would learn to minimize the error such that the upcoming maintenances would be more accurate [13].

$$\min_{c_j \in \mathcal{T}_{c_i}} |\widehat{Tr}_{c_i}^{p_k} - Tr_{c_j}^{p_k}| \quad (4)$$

3 Social Network Representation

To analyze our social network for service selection, many parameters described in the literature about social networks could be considered. A detailed list of such parameters are presented in [5]. For space limit, we consider only the following parameters and provide equations to compute them in our context of trust for service selection. Without loss of generality, we would like to measure the probability (likelihood) of edge creation between a customer and a provider agent. The focus of this paper is on the study of edge-by-edge evaluation of the social network in microscopic manner. We compare the network formation of different types of agents that are using different trust establishment method and use different strategies. Hence, we effectively analyze the impact of different trust models in socializing a particular agent that joins a network and seeks to increase its overall outcome (so-called utility). We basically distinguish between different models based on their strategies of network formation in agent arrival, edge arrival and interaction maintenance process (how after-interaction parameters affect the strategies that are used in the further actions of agents).

3.1 Outdegree

Outdegree is a parameter for the extent to which an agent in the network conveys information regarding some other agents. Outdegree value from the customer's point of view, is to what extent a customer agent knows the providers. The idea is to reflect the fact that a customer that is connected to more reliable providers has a higher outdegree than a customer linked to less reliable ones.

In other words, the outdegree value reflects the extent to which an agent tries to set up and strengthen more edges connecting it to other agents. Equation 5 computes this parameter for an agent Ag , where $\alpha Tr_{Ag}^{c_j} = Tr_{Ag}^{c_j} \cdot NI_{Ag}^{c_j} \cdot TiR_{Ag}^{c_j}$ and $\alpha Tr_{Ag}^{p_k} = Tr_{Ag}^{p_k} \cdot NT_{Ag}^{p_k} \cdot TiR_{Ag}^{p_k}$.

$$D_{out}(Ag) = \sum_{c_j \in \mathcal{T}_{Ag}} \alpha Tr_{Ag}^{c_j} + \sum_{p_k \in \mathcal{T}'_{Ag}} \alpha Tr_{Ag}^{p_k} \quad (5)$$

where $\mathcal{T}'_{Ag} = \{p_k \in P \mid \exists v \in \mathbb{R}^3 \wedge R_{cp}(Ag, v, p_k)\}$

3.2 Indegree

Indegree is a parameter for the extent to which a customer in the network receives information regarding to a particular agent from some other agents. Indegree value from the customer's point of view, is the extent that the agent is known by the close agents in the network. The idea is to reflect the fact that a customer that is connected to more reliable providers has a higher indegree than a customer linked to less reliable ones. Indegree value from a provider's point of view, is the extent that a provider agent is popular in the social network that causes higher number of requests from the customer agents. In other words, the indegree value reflects the popularity of an agent in the sense that any agent would like to increase it and thus cares not to distract it. Equation 6 computes this parameter for a generalized agent Ag , that could be a customer or a provider agent.

$$D_{in}(Ag) = \sum_{c_j \in \mathcal{S}_{Ag}} \alpha Tr_{c_j}^{Ag} \quad (6)$$

where $\mathcal{S}_{Ag} = \{c_j \in C \mid \exists v \in \mathbb{R}^3 \wedge (R_{cc}(c_j, v, Ag) \vee R_{cp}(c_j, v, Ag))\}$

3.3 Homophily

Homophily is a parameter for the extent to which a customer in the network chooses to interact with a provider that is known and is already evaluated (this concept is derived from [1]). This basically raises to strengthen the correlation of adjacent agents. In the social network, agents that are known from previous interactions may tend to request for a service, which is expected to be satisfactory. This is the affect of being friend in a network. In general, it is likely that a customer agent re-selects a particular provider agent aiming to request for a new service. Thus, provider agents normally try to provide a quality service to keep their customers. The homophily of agents in the network is a factor that is not directly compared to other choices of the customer agent, that is seeking for a service. Basically it is the matter of how well-quality the provider agent would provide the new service. This means that, the customer agent's concern is to measure the probability of gaining the expected quality in the service given the fact that the provider agent has already provided a similar service to the same customer. This possibility measurement is mainly related to the indegree

value of the provider agent in the sense that a provider with high indegree value is known to be popular, so there is less chance of disturbing its popularity by providing a not promised service quality. In Section 4, we analyze this effect in more details showing that the trust models with the after-interaction policies could lead to a more accurate friendship evaluations.

Equation 7 computes the probability of selecting a provider p_k with $D_{in}(p_k)$ as indegree value. In this equation, we do not involve the trust measurement that the customer agent c_i performs for evaluating the provider agent p_k ($Tr_{c_i}^{p_k}$). The reason is that since the customer agent c_i is already in relation with the provider p_k , then based on the previous evaluation, could decide whether it worths to select this provider again. If by any chance, the previous history does not reflect the efficiency of the provider p_k , there is no point for investigating the probability of the provider's efficiency if being selected. In equation 7, the value ω is set to be the uncertainty factor (see equation 3) of the history between the customer agent c_i and the provider agent p_k . And the value β represents the coefficient set for the system inconsistency. In the trust models with after-interaction strategies, this value is dynamically modified reflecting the system accuracy level. However, without maintenance process, the value is set initially and remains fixed.

$$p(D_{in}(p_k)) = \frac{e^{\omega \log(D_{in}(p_k)+1)+\beta}}{1 + e^{\omega \log(D_{in}(p_k)+1)+\beta}} \quad (7)$$

In general, the customer c_i would request for transaction with providers that their total trust values are high enough and their homophily probability exceeds a predefined threshold μ_9 . The new rule for initiating a transaction is given as follows:

$$Transact(c_i, p_k) \leftarrow Tr_{c_i}^{p_k} > \mu_6 \wedge p(D_{in}(p_k)) > \mu_9$$

3.4 Confounding

Confounding is a parameter for the extent to which a provider as an external agent influences a customer agent to request for a particular service (this concept is derived from [1]). This influence affects some close agents in the network to set up an edge with an unknown provider under the promising conditions that the provider defines. To this end, the provider that is looking for the customers requests to interact (*Intr*) and specifies some conditions that it promises to provide (*conditions*(p_k)). The customer agent analyzes the request and thus would may accept or refuse the interaction according to the following rules:

$$\langle Rep, accept, c_i, p_k, *, t_1 \rangle \leftarrow \langle Req, Intr, p_k, c_i, conditions(p_k), t_0 \rangle$$

$$\langle Rep, Refuse, c_i, p_k, *, t_1 \rangle \leftarrow \langle Req, Intr, p_k, c_i, conditions(p_k), t_0 \rangle$$

In general, the providers that join the network, seek for the agents that are more likely to request for their services. In other words, when a provider agent

is being activated, it tries to socialize itself in the network. Thus, starting from very close customer agents, the provider agent encourages them to request for its service. To this end, the provider at the beginning acts generously in order to attract the customers and gain more popularity. Moreover, upon high quality service, the customer agents may influence their adjacent agents to request for the same service. So, the provider agent takes the outdegree value of the customer agents into account and based on the interaction span of the customer agents, it provides high quality services.

In confounding factor, the probability of activating an agent c_i with a provider agent p_k is computed in equation 8. As it is assumed that the provider p_k is unknown to the customer c_i , the customer agent would evaluate the social trustworthiness value of the provider. Given the fact that the trust measurement requires some information from the other adjacent agents, the customer agent takes the entropy value into account in order to partially consider the indirect trust value ($ITr_{c_i}^{p_k}$) and the rest for the popularity of the provider agent. Thus, the customer c_i first evaluates the provider p_k and then considers the p_k 's in-degree value together with the network inconsistency level. If the information obtained for evaluating p_k is not enough, the entropy value ω would be high, so that mostly the trust evaluation part would be considered. This would normally cause to lower the overall probability of activation.

$$p(c_i, D_{in}(p_k)) = \omega \times ITr_{c_i}^{p_k} + (1 - \omega) \times \frac{e^{\log(D_{in}(p_k)+1)+\beta}}{1 + e^{\log(D_{in}(p_k)+1)+\beta}} \quad (8)$$

The providers whose probability is greater than a predefined threshold μ_{10} would likely transact with the customer if the two conditions set in the previous rule (Section 3.3) are satisfied. The new rule to decide about initiating a transaction is given as follows:

$$Transact(c_i, p_k) \leftarrow Tr_{c_i}^{p_k} > \mu_6 \wedge p(D_{in}(p_k)) > \mu_9 \wedge p(c_i, D_{in}(p_k)) > \mu_{10}$$

3.5 Influence

Influence is a parameter for the extent to which an agent is prompted to initiate a request caused by an adjacent agent (this concept is derived from [1]). This could take place in a friendship of agents that distribute the idea of some services to be requested. When an agent needs to request a particular service from a provider, it may have already set up an edge with that provider, so the evaluation can be done, or may need to set up a new edge upon which could obtain the service. This is the affect of getting encouraged by a friend in a network. In general, it is likely that a person does action because his friend has already done it. Thus, it is the matter of activation of a new edge, which is set up between a customer agent and the provider agent, that has already been requested for a service by the customer's adjacent agent (friend).

In the confounding factor, we mentioned that when a typical provider agent advertises its service to a couple of adjacent customer agents, it considers that some of the customers may propagate its quality of service to their adjacent

agents, which could lead to more service requests. On the other hand, the customer agent that is being prompted to take a service produced by a particular provider, needs to evaluate both the advertising adjacent agent c_j ($DTr_{c_i}^{c_j}$) and the provider itself p_k ($ITr_{c_i}^{p_k}$). Equation 9 computes the influence-based probability of activation of a customer agent c_i regarding to taking the service produced by a provider agent p_k . In this equation, ω_{c_j} is the entropy value related to the information c_i has and thus could rely on, and ω_{p_k} is the entropy value related to the information that c_i has about the provider p_k .

$$p(c_i, c_j, D_{in}(p_k)) = \omega_{c_j} \times DTr_{c_i}^{c_j} + (1 - \omega_{c_j}) \times \Theta \quad (9)$$

where

$$\Theta = \omega_{p_k} \times ITr_{c_i}^{p_k} + (1 - \omega_{p_k}) \times \frac{e^{\log(D_{in}(p_k)+1)+\beta}}{1 + e^{\log(D_{in}(p_k)+1)+\beta}}$$

Finally, the decision about initiating a transaction with a provider can be given considering the previous conditions (Section 3.4) and the fact that the influence probability is greater than a predefined threshold μ_{11} . The final rule is then specified as follows:

$Transact(c_i, p_k) \leftarrow$

$$Tr_{c_i}^{p_k} > \mu_6 \wedge p(D_{in}(p_k)) > \mu_9 \wedge p(c_i, D_{in}(p_k)) > \mu_{10} \wedge p(c_i, c_j, D_{in}(p_k)) > \mu_{11}$$

4 Experimental Results and Related Work

In this section, we describe the implementation of proof of concept prototype. In the implemented prototype, agents are implemented as *Jadex*^{©TM} agents. Like in [7], the testbed environment is populated with two agent types: (1) *service provider agents*; and (2) *service consumer agents*. The simulation consists of a number of consequent Runs in which 200 agents are activated and build their private knowledge, keep interacting with one another, and enhance their overall knowledge about the environment. Depending on the agent interactions, agent may extend their connections hoping to be more socialized. However, there is always the chance of investing on wrong agents that lead to no outcome. Here, we distinguish agents by the service (or information) quality that they provide. Table 1 represents four types of the service providers we consider in our simulation: good (15% of the population), ordinary (30% of the population), bad (15% of the population) and fickle (40% of the population). The first three provide the service regarding to the assigned mean value of quality with a small range of deviation. Fickle providers are more flexible as their range of service quality covers the whole possible outcomes. Upon interaction with service providers, service consumer agents obtain utilities and consequently rate the quality of the providers (for simplicity, we assume only the consumers are interconnected to the provider agents). In the simulation environment, agents are equipped with different trust models in the sense that their edge creation policies are different. In the proposed model, we try to establish a trust mechanism where an

agent, firstly can maintain an effective trust assessment process and secondly, accurately updates its belief set, which reflects the other agents likely accuracy. In order to observe the impact of each contribution, we compare the proposed model with other trust models in two perspectives. In the first comparison, we use the agents that only perform a direct trust assessment process. We refer to this group of agents as *Direct Trust Group (DTG)*. In the second comparison, we use the agents that (in addition to the direct trust assessment mechanism), perform maintenance process for evaluating the consulting agents in order to increase their information accuracy. We refer to this group of agents as *Maintenance-based Trust Group (MTG)*. The reason of decomposing the proposed model to two groups is to focus on the efficiency of each model, which enables us to analyze the impact of each contribution on the accuracy of the agent in edge creation process. In order to discuss the proposed model's overall performance, we compare it with BRS ¹ [9] and Travos ² [14] trust models. These models are similar to the proposed model in the sense that they do consider other agents' suggestions while evaluating the trust of some specific agents and discard inaccurate suggestions aiming to perform most efficient edge creation. The detailed description of these models is provided in [4]. Here, we basically distinguish [10] between different models based on their strategy of network formation in agent arrival, edge arrival and interaction maintenance process (how after-interaction parameters affect the strategies that are used in the further actions of agents). In the rest of this section, we discuss the impacts of efficient parameters in the edge extension of agents and elaborate how different trust mechanisms effectively deal with these impacts.

Table 1. Simulation summarization over the obtained measurements

Service provider type	Density in the network	Utility range	Utility SD
Good	15.0%] + 5, +10]	1.0
Ordinary	30.0%] - 5, +5]	2.0
Bad	15.0%] - 10, -5]	2.0
Fickle	40.0%] - 10, +10]	—

Provider Popularity. We start the discussion by the probability of selecting the providers over their different popularity values. As we discussed earlier, the indegree value of a node reflects their popularity in the social network. Thus, we could conclude that the chance of selection for a particular service provider agent would be proportionally relevant to its indegree value (ordinary selection attitude). However, the trust evaluation method together with its distribution process would affect this probability of selection. Illustrated in figure 1, the BRS

¹ BRS trust model collects the after-interaction ratings and estimates the trust using beta distribution method. This trust model ignores the ratings from such agents that deviate the most from the majority of the ratings.

² Travos trust model is similar to BRS in collecting the after-interaction ratings and estimating the trust using beta distribution method. But Travos ignores the ratings from agents that provide intermittent reports in the form of suggestions.

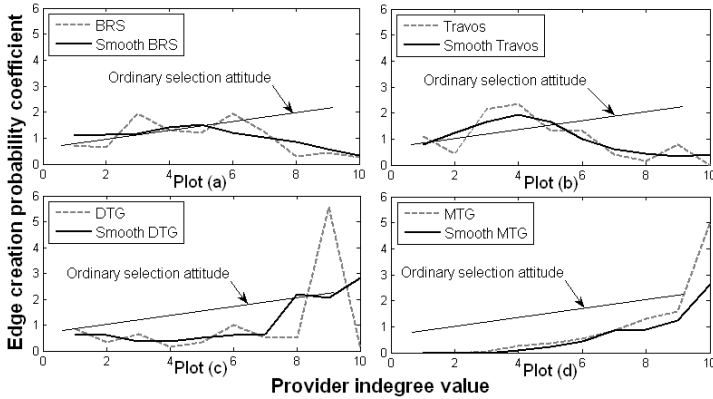


Fig. 1. Probability of edge creation with provider agent vs. the provider's indegree value

agents act independently of the mentioned probability as the BRS agents do not consider the popularity of the provider. Travos agents also do not consider such value. However, the probability of selection of the popular providers increase as they take less risk of changing their behaviors and thus perform satisfactory services, which would lead to their selection. In general, because of inaccuracy detection feature of Travos agents, the percentage of selection of provider agents with high indegree value increases in a gentle manner. At some certain point, the selection of popular providers are coming down (see *plot b*). This is explained by the fact that a popular provider has large number of recommenders that provide diverse range of information to the agent that is trying to evaluate the provider. This diversity would lead to confusion state due to high deviation of reports (the state that this system would generalize the majority of the information that is obtained and could be inaccurate), which in Travos would cause the drop of the suggestions and thus the selection would be less. The proposed model agents (*DTG* and *MTG*) follow the information propagation feature as the adjacent agents influence each other to select the high quality providers. There is a difference in the slope of selection graph in *MTG* and *DTG* models. This is explained by the fact that agents in the *MTG* group are characterized by the maintenance process that enable them to recognize high quality provider agents and thus their accuracy in influencing adjacent agents are more than regular *DTG* agents. In general, since the maintenance feature does not exist in *DTG* group, the customer agents loose the track of high quality provider agents, and thus the probability of selection would not increase so fast.

Interacting Agents Age. In general, in the defined testbed, the agents that are obtaining a high quality service are encouraged to distribute their experience to other adjacent agents (influence others). This activity of agents would basically get increased over the time, or say over the age of the agent. In figure 2, we have compared the activity of different groups of agents by comparing edge extension of the agents (outdegree value). Without loss of generality, the edge extension

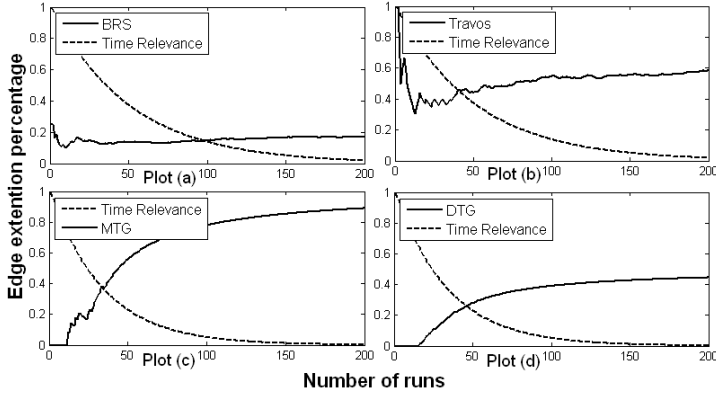


Fig. 2. Agent edge extension vs. the agents age

is proportionally related to the accuracy of agent in detecting the high quality providers. In BRS model, the extension over the time is not increasing as the agent gets involved with high number of adjacent agents and would be difficult to effectively extend the social activity, so more or less would be independent of the age of the agent. Travos and *DTG* models are increasing, however relatively with small slope. In *MTG* group, because of the maintenance process the agents would be encouraged to initiate a request to high quality service providers and thus extend their activity. In this graph, the slope is relatively large as over the time, the agent could manage to categorize the providers that could possibly act beneficially for the agent, and thus would enlarge his activity area. In figure 2, the second line represents how fast the agents would drop the previous data and use the recent data for their analysis. This dropping factor is also relevant to how active an agent is and thus, to what extent there would be available resource that agents could drop obsolete data. *DTG* and *MTG* group use the same dropping feature ($TiR(\Delta t_{Ag_a}^{Ag_b})$), which is derived in equation 10. Variable λ is an application-dependent coefficient. In some applications, recent interactions are more desirable to be considered (λ is set to relatively large number). In contrast, in some other applications, even the old interactions are still valuable source of information. In that case, a relatively smaller value to λ is used.

$$TiR(\Delta t_{Ag_a}^{Ag_b}) = e^{-\lambda \log(\Delta t_{Ag_a}^{Ag_b})} \quad \lambda \geq 0 \quad (10)$$

Homophily-Confounding-Influence. We would like to go further into the details of the selection history in terms of the microscopic social network affects (homophily, confounding, and influence) and illustrate them in figure 3. in this section, we observe the diverse impacts of homophily, confounding and influence features on each group in the sense that we would capture their edge creation reasons. Note that the edge creation is not the important issue, however, the concern is to extend to the agents that are known to be trustworthy. Therefore, we elaborate the overall outcome of different agents at the following. The

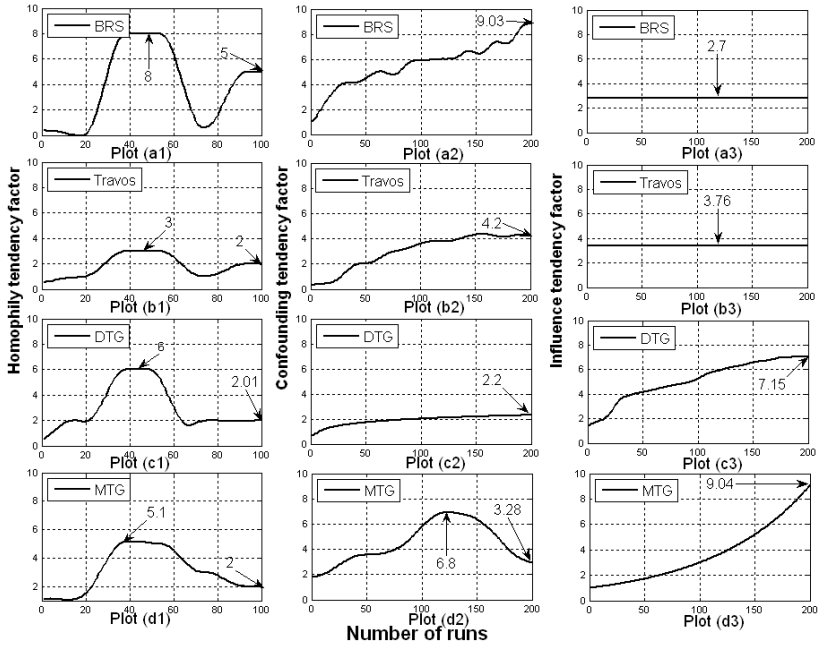


Fig. 3. Overall comparison of the proposed model with BRS and Travos in terms of (a) Homophily; (b) confounding; and (c) influence factors

homophily aspect would be caused by the friendship relation of the agents that have history interaction between them. This is a very general case in the sense that consumer agents over the time would get to know and select the provider agents. If the interacted service is satisfactory for the agent, then the consumer agent may re-select the same provider agent in some future. BRS agents are the ones that mostly rely on the homophily affect in the sense that they keep the history of the interaction in order to re-evaluate the provider agent. The providers that remain trustworthy would be selected over the time. As it is clear from plot *a1*, once the providers change their policies, the selection of them would be affected so fast, as the BRS agents recognize that they should start seeking for the appropriate friends. Travos agents also rely on the previous history and re-select the previously interacted service providers (see plot *b1*). However, over the time the reports regarding to the accuracy of the providers would be divergent, which would lead to refuse the selection. The same reason is the case for *DTG* and *MTG* group (shown in plots *c1* and *d1*). These agents to some extent rely on the previous history and select the providers. After some certain time, these agents also recognize the inconsistency in the evaluation process of the history interacted providers. Overall, *DTG* and *MTG* agents evaluate the providers in a very accurate manner. The accuracy that Travos, *DTG* and *MTG* agents have cause the decremented manner after some certain time.

Confounding factor reflects the extent to which the provider agents advertise their service to the consumers (could be new or previously serviced ones). This feature also affects BRS group, as they start evaluating the advertising provider, and thus extend their activation area. Plot *a2* indicates that the BRS group are easy to involve in interaction with the advertising provider agent. Travos agents act in the same way as the provider agents could induce them to take their service. However, Travos agents are considering this case less, because they investigate the previous reports related to the advertising provider and doubt on the inconsistent ones (see plot *b2*). In general, the BRS and Travos agents accept the confounding-related interactions over the time, and thus their graph has an increasing manner. But in *DTG* and specially *MTG*, the agents would not accept this service all the time, as over the time, once the network inconsistency level increases, these agents would have confusion in accepting the confounding-related affect caused by unknown service providers (see plots *c2* and *d2*). *MTG* agents would accept this option from the providers, but since they are equipped with a maintenance process, they would distribute the performance of the providers to the adjacent agents, which would lead them to get to know the network faster than the other models. This would let the *MTG* agents to select the best providers, and thus would drop the request from most of the unknown agents while they are already in a good accuracy level.

Influence factor is mostly used by active agents, while they obtain service and tend to distribute the efficiency of the interaction to the adjacent agents. Since BRS agents independently select the providers, the influence is not a factor for these agents (plot *a3*). Travos agents would act almost independently, however the Travos agents are encouraged by the reports they obtain for the evaluation of a particular provider agent (plot *b3*). *DTG* group would be encouraged with the same factor as Travos agents. Upon evaluating provides, the *DTG* agents would consider the reports obtained from adjacent agents and recognize outstanding service provided by the provider that is just served an adjacent agent (see plot *c3*). The influence-related interactions are mostly initiated among *MTG* group, shown in plot *d3*. This is explained by the fact that the *MTG* group are equipped with maintenance feature, which enables them to reason about the accuracy and efficiency of the obtained services and propagate the information to the adjacent information.

General Performance. Considering all the involved features, at the end we compare the models in general perspective, starting good provider selection efficiency. In such a biased environment, the number of good providers are comparatively low. Therefore, the agents need to perform an accurate trust assessment to recognize the best providers. As it is clear from the Figures 4, plots *a1*, *b1*, and *c1*, *DTG* agents function better than other models (Travos and BRS). The reason is that in this model, agents are assessing the credibility of the providers using other agents suggestions depending on their credibility and to what extent they know the provider. Afterwards these agents rate the provider, which would be distributed to other agents upon their request (relatively in plots *a2*, *b2*, and *c2* the comparison of fickle selection percentage, and in *a2*, *b2*, and *c2*,

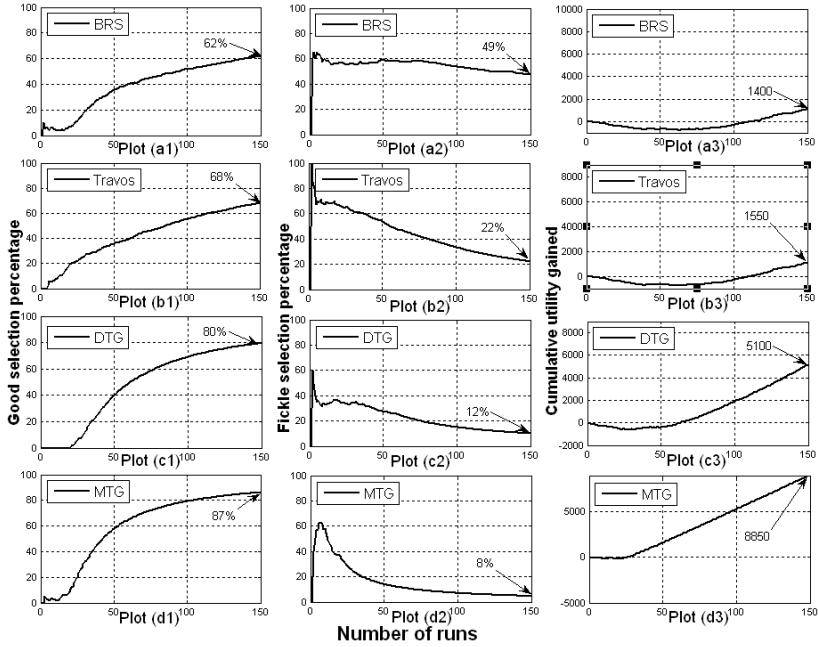


Fig. 4. Overall comparison of the proposed model with BRS and in terms of (a) good selection percentage; (b) fickle selection percentage; and (c) cumulative utility gained

the gained cumulative utility is shown). Not excluding the fact that *DTG* agents are considering partial ratings for consulting agents, we state that they weakly function when the environment contains agents that do not truthfully reveal their believes. *MTG* agents in addition to the direct trust assessment, provide incentives for consulting agents, which encourages them to effectively provide the information aiming to gain more utility. Plot *d1* shows that *MTG* agents outperform other models in best provider selection. This is expressed by the fact that *MTG* agents recognize the best providers ensuring that the best selected provider would provide the highest utility. Relatively plot *d2* shows an outperform in fickle selection and consequently higher cumulative utility in plot *d3*.

In BRS model, the trustor agent in the assessment process uses beta distribution method and discards the ratings that deviate the most from the majority of the ratings. Concerning this, BRS is comparatively a static trust method, which causes a low-efficient performance in very dynamic environment. In general, if a BRS agent decides to evaluate an agent that he is not acquainted with, he considers the majority of ratings, which are supposed to be truthfully revealed about the trustee agent. In such a case that the trustee agent has just changed his strategy, the trustor agent would loose in trust assessment and does not verify the accuracy of the gained information. Therefore, as illustrated in figure 4, plots *a1*, the BRS agents would have less percentage of good providers selection, relatively higher percentage of fickle providers selection (plot *a2*), and consequently lower gained cumulative utility (plot *a3*).

Travos [14] trust model is similar to BRS in using beta distribution to estimate the trust based on the previous interactions. Travos model also does not have partial rating. Hence, the trustor agent merges his own experience with suggestions from other agents. However, unlike BRS model, Travos filters the surrounding agents that are fluctuating in their reports about a specific trustee agent. To some extent, this feature would cause a partial suggestion consideration and thus, Travos agents would adapt faster comparing to BRS agents. Rates concerning the good and fickle selection percentage shown in figures 4, plots *b1* and *b2* reflect higher efficiency of Travos compared to BRS. However, Travos model considers that agents do not change their behavior towards the elapsing time. These missing assumptions affect the accuracy of trust estimation in a very biased environment (lower gained cumulative utility in plot *b3*).

5 Conclusion

The contribution of this paper is the detailed investigation of a trust-based multi-agent architecture in edge creation and correlation formation in social networks. The analysis of this issue is done by combining both declarative and numerical techniques. The established trust is provided by the proposed framework, that is briefly explained here. The trust assessment procedure is based on integrating suggestion of consulting agents, objectively enhancing the accuracy of agents to make use of the information communicated to them. The surveillance over the surrounding environment makes distributed agents eager to extend their activity area by interacting with high quality agents. In the proposed framework, maintenance process considers the communicated information to judge the accuracy of the consulting agents in the previous trust evaluation process. The ex-interacted analysis allows the agents to propagate the recent and accurate information to their adjacent agents, which is considered as homophily and influence factors in edge creation process.

Our model has the advantage of being computationally efficient as it takes into account the important factors involved in extending the activity zone of agents. Moreover, we have done a detailed empirical analysis over the edge creation and behavior of agents over their age, while they are equipped with different trust mechanism protocols. The proposed mechanism efficiency is compared with other related models to prove the capabilities of the proposed model. Our plan for future work is to advance the assessment model to enhance its efficiency by considering more efficient learning algorithms. In the maintenance process we need to elaborate more on the optimization part, trying to formulate it in the sense to be adaptable to diverse situations. We need to consider more extensions towards having links and correlations between provider agents and thus, we need to deal with the selfish actions that providers may perform under the assumption of having social links with other providers. Game theory and mechanism design are the most promising techniques to be investigated for such an issue. Finally, we plan to maintain more detailed analysis in comparison with other models to capture more results reflecting the proposed model capabilities.

Acknowledgements

This work is supported by Jamal Bentahar's funds from Natural Sciences and Engineering Research Council of Canada (NSERC: 341422-07), Fonds québécois de la recherche sur la nature et les technologies (FQRNT: 2008-NC-119348) and Fonds québécois de la recherche sur la société et la culture (FQRSC: 2007-111881). We would also like to thank the anonymous reviewers for their very interesting comments.

References

1. Anagnostopoulos, A., Kumar, R., Mahdian, M.: Influence and correlation in social networks. In: Proceedings of the 14'th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 7–15 (2008)
2. Backstrom, L., Huttenlocher, D., Kleinberg, J., Lan, X.: Group formation in large scale networks: membership, growth, and evolution. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 44–54 (2006)
3. Bentahar, J., Khosravifar, B., Gomrokchi, M.: Social network-based trust for agent-based services. In: Proceedings of the 23rd IEEE International Conference on Advanced Information Networking and Applications (AINA), Symposium on Web and Mobile Information Services, pp. 298–303 (2009)
4. Bentahar, J., Khosravifar, B.: Using trustworthy and referee agents to secure multi-agent systems. In: Proceedings of the 5th IEEE International Conference on Information Technology: New Generations (ITNG), pp. 477–482 (2008)
5. Buskens, V.: Social Networks and Trust. Kluwer Academic Publishers, Dordrecht (2002)
6. Christakis, N.A., Fowler, J.H.: The spread of obesity in a large social network over 32 Years. *The new England Journal of Medicine* 357(4), 370–379 (2007)
7. Dong-Huynh, T., Jennings, N.R., Shadbolt, N.R.: Fire: an integrated trust and reputation model for open multi-agent systems. *Journal of Autonomous Agents and Multi-Agent Systems* 13(2), 119–154 (2006)
8. Dong-Huynh, T., Jennings, N.R., Shadbolt, N.R.: Certified reputation: how an agent can trust a stranger. In: Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS), Japan, pp. 1217–1224 (2006)
9. Jesang, A., Ismail, R.: The beta reputation system. In: 15th Bled Electronic Commerce Conference e-Reality: Constructing the e-Economy (June 2002)
10. Leskovec, J., Backstrom, L., Kumar, R., Tomkins, A.: Microscopic evolution of social networks. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 462–470 (2008)
11. Khosravifar, B., Gomrokchi, M., Bentahar, J., Thiran, Ph.: A maintenance-based trust for Open multi-agent systems. In: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 1017–1024 (2009)
12. Sabater, J., Paolucci, M., Conte, R.: Repage: REputation and ImAGE among limited autonomous partners. *Journal of Artificial Societies and Social Simulation* 9(2) (2006)

13. Siddiqi, S.M., Boots, B., GGordon, G.J.: A Constraint generation approach to learning stable linear dynamical systems. In: Proceedings of Advances in Neural Information Processing Systems, NIPS (2007)
14. Teacy, W.T., Patel, J., Jennings, N.R., Luck, M.: Travos: trust and reputation in the context of inaccurate information sources. *Autonomous Agents and Multi-Agent Systems* 12(2), 183–198 (2006)
15. Xiong, L., Liu, L.: PeerTrust: supporting reputation-based trust for peer-to-peer electronic communities. *Journal of IEEE Transactions on Knowledge and Data Engineering* 16(7), 843–857 (2004)
16. Wang, Y., Singh, M.P.: Formal trust model for multiagent systems. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI), pp. 1551–1556 (2007)