A Graph-based Approach for Promoting Honesty in Community-based Multiagent Systems

Georgia Kastidou[1], Robin Cohen[1], and Kate Larson[1]
{gkastidou,rcohen,klarson}@cs.uwaterloo.ca

David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, ON, N2L3G1, Canada

Abstract. We present a graph-based heuristic approach for reducing the number of communities that are queried, when communities share information about their agents, in multiagent settings. In particular, our approach exploits the consistency among the advice of the queried communities resulting in a more competitive environment in which communities are inclined to be honest. As argued in this paper, providing a method for communities to obtain sufficiently valuable information about agents with a reduced number of queries is an important element of many systems. Exploiting consistency among good advisors will further promote honest behaviour since fewer communities will be consulted and thus resources will be restricted to only the most reliable sources. As a result, our proposed selection process contributes to the development of an effective overall framework for sharing agent reputation ratings in community-based environments.

1 Introduction

Consider the scenario where there are agents representing the interests of users and there are multiple communities that an agent can join (e.g a P2P file sharing system). By community, we mean a collection of agents that co-exist for a specific purpose. Consider as well that agents may be participating in multiple communities simultaneously and/or may be migrating from community to community. Our aim is to design an efficient framework that enables communities to share information about the reputability of their agents. The key idea of our research is to promote truthfulness amongst agents and communities with the ultimate goal of increasing the social welfare of each community that participates in the mechanism. Agents should be inclined to be good citizens within their communities because their reputation will be shared; communities need to be inclined to honestly share the reputation ratings of their agents in order to be able to benefit from the information provided by the other communities in the system.

Given that communities are primarily self-interested in some cases they have incentives to misreport the evaluation of their agents. For example, a community

C can be reluctant to provide a truthful evaluation for one of its very good agents to another community C'. This is due to the fact that if the community C' accepts the agent then C might have to share the resources, that the agent contributes, with C'. Given that the agent's resources can be limited, this can result in a decrease of the agent's contribution to C.

To address this problem and promote truthfulness, we propose the use of a graph-based heuristic which exploits the consistency among the advice of candidate advisors and reduces the number of communities that are queried each time a community seeks information about a prospective agent in such a way that all the communities are inclined to provide truthful reports. Reducing the number of communities that are queried is important since each time a community requests information from another community it has to provide some resources. We argue that exploiting consistency among good advisors will further promote honest behaviour since fewer communities will be consulted and thus resources will be restricted to only the most reliable sources. This will create a more competitive environment between the advisor communities which will be inclined to be truthful in order to increase the probability of being asked in the future.

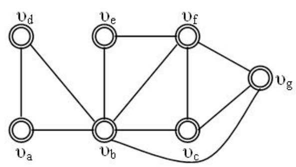
2 Model

Let C_i denote community i and let a_j denote agent j. We assume that if a_j is a member of community C_i then C_i can observe and judge the quality of agent a_j. In particular, we assume that community C_i maintains a *reputation model* for all member agents, and is able to assign a *reputation rating* r_j^i to agent a_j, where $r_j^i \in \{Good, Poor\}$. If agent a_j wishes to join community C_i, then before welcoming a_j, community C_i will contact the communities in which a_j is currently, or was previously, a member, and will request information regarding a_j. We denote the set of these communities as $S(a_j)$. We will refer to the community C_i as the *recipient community* and the communities in $S(a_j)$ as the *advisor communities*. We assume that the set $S(a_j)$ is provided by the agent a_j.[1]

In this paper we aim to identify a subset, S', of a set of candidate advisors in a way that will not compromise the quality of information C_i receives regarding the reputation of an agent, while at the same time provide strong incentives for all the participant communities to be truthful. We argue that this can be achieved by exploiting the consistency among communities which provide good quality of information.

In order to exploit the consistency among the advice of the advisor communities we consider a graph-based heuristic. First, for each community $C \in S(a_j)$ we identify the set of communities with which it appears to be consistent in providing advice and which also appear to be consistent with each other, then we select the community with the 'strongest' set, and finally from this set we select

[1] In the future we plan to develop a reasoning mechanism that will reward truthful agents to a certain extent, even if they had an apparently justified poor contribution in a percentage of communities in which they have been members, and it will penalize agents that appeared to have not disclosed some of these communities.



v	$Pr^+(v)$	$Pr^-(v)$	$Expr(v)$
v_a	0.75	0.7	0.65
v_b	0.7	0.8	0.55
v_c	0.74	0.85	0.68
v_d	0.68	0.77	0.74
v_e	0.85	0.77	0.56
v_f	0.67	0.78	0.75
v_g	0.76	0.65	0.69

$e=(v,w)$	$Pr^+(e)$	$Pr^-(e)$	W_e
(v_a,v_b)	0.7	0.07	0.6
(v_a,v_d)	0.8	0.05	0.7
(v_b,v_c)	0.65	0.03	0.5
(v_b,v_d)	0.8	0.08	1
(v_b,v_e)	0.8	0.05	0.8
(v_b,v_f)	0.9	0.01	0.7
(v_b,v_g)	0.55	0.05	0.8
(v_c,v_f)	0.7	0.06	0.9
(v_c,v_g)	0.6	0.1	0.8
(v_e,v_f)	0.7	0.06	0.6
(v_f,v_g)	0.8	0.08	0.3

Fig. 1. An example of a Consistency Graph G_{C_i} with its weights.

the ‘strongest’ community. The latter community is the first candidate community to ask. Then we remove it and repeat the procedure in order to select the second candidate and so on.

More specifically, each community C_i constructs a graph G_{C_i} that maintains partial information regarding the consistency among the information that other communities have provided to C_i in the past. In particular, we refer to the graph $G_{C_i} = (V, E, Pr^+(V), Pr^-(V), Pr^+(E), Pr^-(E))$ as the *Consistency Graph* of the community C_i. Each vertex v_k of the *Consistency Graph* G_{C_i} represents an advisor community C_k from which the community C_i has requested information in the past. Each vertex $v_k \in V$ is described by a tuple $(Pr^+(v_k), Pr^-(v_k), Expr(v_k))$, where $Pr^+(v_k)$ represents the probability an agent a will be a ‘Good Contributor’ in C_i given that C_k characterized it as a ‘Good Contributor’, $Pr^-(v_k)$ represents the probability an agent a will be a ‘Poor Contributor’ in C_i given that C_k characterized it as a ‘Poor Contributor’, and $Expr(v_k)$ represents the total number of times the community C_k was asked by C_i. An example of a *Consistency Graph* is depicted in Fig. 1. For instance, the community which owns the graph in Fig. 1 has received information in the past from the communities $\{C_a, C_b, C_c, C_d, C_e, C_f, C_g\}$.

The existence of an edge $e \in E$, $e = (v_j, v_k)$, indicates that in the past the community C_i has requested simultaneously and received information from the communities C_j and C_k regarding at least one agent. Each edge $e \in E$ is described by a tuple $(Pr^+(e), Pr^-(e), W_e)$ (Fig. 1), where $Pr^+(e)$ represents the expected probability the communities C_j and C_k will provide consistent information regarding a ‘Good Contributor’, $Pr^-(e)$ represents the expected probability the communities C_j and C_k will provide consistent information regarding a ‘Poor Contributor’, and W_e represents the degree the amount of information that is available based on past reports is sufficient to reason about the consistency of the reports of the communities C_j and C_k and can be determined based on the number of times the community C_i has requested information simultaneously from both the communities C_j and C_k. The latter probabilities are based on past experience and get updated each time an interaction, that involves the communities the vertices and/or the edges are associated with, takes place.

We consider that two or more communities provide *consistent information* regarding a 'Good Contributor' ('Poor Contributor') if they have characterized an agent as 'Good Contributor'('Poor Contributor') and the agent proved to be a 'Good Contributor' ('Poor Contributor') inside the advisor community. As we discuss in Section 3, we distinguish the case of 'Good Contributors' and 'Poor Contributors' and in each case we consider only the probabilities $(Pr^+(E), Pr^+(V))$ and $(Pr^-(E), Pr^-(V))$, respectively. Thus, for simplification reason, we provide the following definitions based on a 4-tuple graph.

Definition 1 *Given two graphs $G(V, E, Pr(V), Pr(E))$ and $G'(V', E', Pr(V'), Pr(E'))$, where $Pr(V)$, $Pr(V')$, $Pr(E)$ and $Pr(E')$ are probability distributions over V, V', E and E', respectively, the graph G (ϵ, μ)-*-dominates *the graph G' if there is at least one subgraph $G_{sb}(V_{sb}, E_{sb})$ of G isomorphic to G' such that:*

$$average(Pr(V_{sb})) + \epsilon \geq average(Pr(V')) \quad (1)$$

and

$$average(Pr(E_{sb})) + \mu \geq average(Pr(E')) \quad (2)$$

In case $(\epsilon, \mu) = (0, 0)$ we will say that G strongly*$-$dominates the graph G'.*

This definition determines whether the information a graph $G(V, E, Pr(V), Pr(E))$ depicts is equal to or richer than the information a graph $G'(V', E', Pr(V'), Pr(E'))$ depicts. For example, consider two basketball teams, A and B, which have 7 and 5 players, respectively. Assume now that you have to select the better of the two teams. If team A has a subset of 5 players that are 'equally' good as the 5 players of team B, then selecting the team A appears to be a better choice.

Definition 2 *The* E*xternal* M*aximum Clique* (EM-Clique(v)) *of a vertex v inside a graph $G(V, E, Pr(V), Pr(E))$ is the clique in G with the largest order in which the vertex v participates and which (ϵ, μ)*-dominates *all the other cliques of equal or smaller order that the vertex v participates. In case there is more than one clique that satisfies the above requirement, the* EM-clique(v) *will be the clique with the largest* set of external edges*.*

Definition 3 *Given a clique Q of a graph G, the* set of external edges *of the clique Q is the set of all edges $e, e = (v, w)$, in $E(G)$ such that $v \in V(Q)$ and $w \in V(G) - V(Q)$.*

Very briefly, the above definitions find the clique in a graph G with the biggest order that appears to be a better choice among the other cliques of G of equal or smaller order.[2] For example, in Fig. 2 the set of external edges of the clique $\{a, b, c\}$ of the graph G^* is $\{cf, cd\}$. The reason why we are interested in finding the largest clique that satisfies the domination condition and not simply finding

[2] The order of a graph is equal to the number of its vertices.

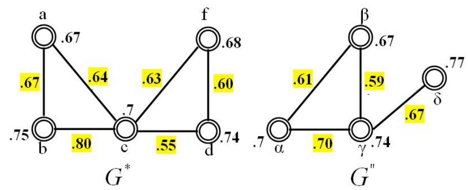


Fig. 2. (ϵ, μ)-domination example

the maximum clique can be well understood by finding the *EM - Clique(v)* of the vertex γ in Fig. 2. Clearly, the largest clique the vertex γ participates in clique G_1 which consists of the vertices (α, β, γ) but the quality of information of the clique G_2 which consists of the vertices (γ, δ) might be more valuable. For this reason, the selection of the *EM - Clique(v)* should also consider the probability distribution both on the vertices of the consistency graph and on the edges. For instance, the average probability distribution of the vertices in G_1 is 0.703 and of the vertices in G_2 is 0.755, while the average probability distribution of the edges in G_1 is 0.633 and of the edges in G_2 is 0.67. As we can see, although the graph G_2 has smaller order it appears to be a better choice with respect to both the probabilities over its edges and its vertices.

Proposition 1 *A graph* strongly dominates *all of its induced subgraphs.*

Proof: A graph G *strongly dominates* a graph g if it has a subgraph g' isomorphic to g such that:

$$average(Pr(V_{g'})) \geq average(Pr(V_g)) \quad (3)$$

and

$$average(Pr(E_{g'})) \geq average(Pr(E_g)) \quad (4)$$

Given that g is an induced subgraph of G, we can simply select for g' the graph g itself.

As we will explain in detail in Section 3.2 the reason we defined the *EM-Clique* of a node v_C is because we are interested in identifying the set of communities S that tend to provide consistent information with the community C, while at the same time provide consistent information with each other. Asking one community in S provides the same value of information as asking any community in S. For example, if Nick and George tend to agree with each other, George and Helen also tend to agree with each other, and in addition Helen and Nick tend to agree with each other, then instead of asking all of them we can simply randomly select one of them and ask him/her instead.

Now, assume that George and Mary were asked a set of questions and they appeared to agree in the majority of the cases. Now assume that we want to ask one question to George, Nick, Helen and Mary. Assume that for each person we ask we have to pay an amount of money and we can only afford to ask one. Clearly, based on the information we have, our best choice is George, since the probability he will agree with Nick, Mary and Helen is higher than anyone else. This information is what we aim in capturing with the following definition of the *Coverage Graph* of a community.

Definition 4 *A vertex v* covers *a vertex w, where $v, w \in V(G)$, if v participates in w's EM-clique.*

For example, in Fig. 2 the *EM-Clique* of the vertex a is $\{a, b, c\}$, thus the vertices b and c cover the vertex a.

Definition 5 *The* coverage graph $\hat{G}_v$ of a vertex v *in a graph G is the graph that occurs from the union of all the EM-cliques of the vertices in G the vertex v* covers.

For instance, consider the graph G^* in Fig. 2, for $(\epsilon, \mu) = (0.04, 0)$ c's *EM-Clique* is $\{b, d, f\}$. Given now that c covers a and b and that a and b have *EM-Cliques* the $\{a, b, c\}$, the graph G^* is the *coverage graph of a vertex c*.

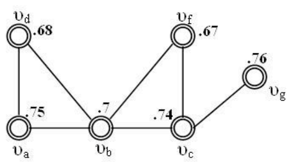
3 Selection Procedure

In this section we describe our procedure for finding the set of advisors L. Although our main aim is to identify the communities which provide accurate information regarding agents who are 'Good' contributors (i.e. agents that the recipient community is more likely to accept) we are also interested in asking a small number of communities that provide consistent information about agents who are 'Poor' contributors. This is due to the fact that communities might be interested in misreporting good agents in fear of losing them or misreporting poor contributors in an effort to get rid of them.

The list L^+ consists of the communities that tend to provided consistent information about agents who are 'Good' contributors while the list L^- consists of the communities that tend to provide accurate information about agents who are 'Poor' contributors. Obviously, $L = L^+ \cup L^-$. In the following sections we will provide the way that L^+ is decided. The list L^- can be decided in a similar way.

3.1 Filtering the Consistency Graph

The filtering step in the *Selection Procedure* refers to removing the nodes that represent communities that either the community C_i would like to ask anyways or represent communities that provide insufficient information. For example, if a community provides accurate information with probability 0.1 then this community would be a bad choice, thus it should be removed from the candidates list. We refer to the graph that is created as the *Consistency SubGraph* and we represent it by $G^+_{cg(i)}$. In particular, the *Consistency SubGraph* $G^+_{cg(i)}$ is created by removing:



v	$Pr^+(v)$	$Pr^-(v)$	$Expr(v)$
v_a	0.75	0.7	0.65
v_b	0.7	0.8	0.55
v_c	0.74	0.85	0.68
v_d	0.68	0.77	0.74
v_f	0.67	0.78	0.75
v_g	0.76	0.65	0.69

$e=(v,w)$	$Pr^+(e)$	$Pr^-(e)$	W_e
(v_a, v_b)	0.7	0.07	0.6
(v_a, v_d)	0.8	0.05	0.7
(v_b, v_c)	0.65	0.03	0.5
(v_b, v_d)	0.8	0.08	1
(v_b, v_f)	0.9	0.01	0.7
(v_c, v_f)	0.7	0.06	0.9
(v_c, v_g)	0.6	0.1	0.8

Fig. 3. The Consistency Subgraph CG_{C_i} of the graph G_{C_i}

- the **vertices** which represent communities that do not belong to $S(a_j)$ or represent communities whose probability of telling the truth regarding 'Good Contributors' is less than an acceptable threshold θ_v^+, or represent communities for which there is no sufficient experience and thus their $Expr(v)$ is less than an acceptable threshold θ_v^{expr}.
- the **edges** which connect communities whose probability of agreeing about 'Good Contributors' is lower than an acceptable threshold θ_e^+, or edges for which there is no sufficient experience and thus their w_e^{cf} is less than an acceptable threshold θ_e^{expr}.

An example of a *Consistency SubGraph* is depicted in Fig. 3. More specifically, the latter graph is created from graph Consistency Graph G_{C_i} in Fig. 1 if we consider that the candidate advisor list is $S(a_j)$=$\{a,b,c,d,f,g\}$, $\theta_v^+ = 0.65$, $\theta_v^{expr} = 0.6$, $\theta_e^{expr} = 0.5$, and $\theta_e^+ = 0.6$. As we can see the vertex e is removed since it does not belong in $S(a_j)$, while the edge $e_1 = (v_f, v_g)$ is being removed because there is no sufficient previous experience (i.e. $w_{e_1}^{cf} < 0.5$) and $e_2(v_b, v_g)$ is removed since the probability of C_b and C_g to agree is below the threshold θ_e^+ (i.e. $w_{e_2}^+ = 0.55 < \theta_e^+ = 0.6$). At this point we need to mention that in order to accumulate experience for the above nodes, each time a number of 'unexplored' nodes can be selected to be asked with some probability p each time.

3.2 Exploiting Consistency When Selecting Advisors

The next step is to exploit the consistency in advice that the communities in the *Consistency Subgraph* provide. In order to achieve this we need to identify the *Dominant Coverage Graph.* This is a graph that contains the set of communities that tend to provide the best quality of information while at the same time are consistent with each other. To identify the *Dominant Coverage Graph* we first need to find the *EM-Clique* of each community in the Consistency Subgraph. The *EM-Clique* of a community C consists of the set of communities that are consistent not only with the community C but with each other as well. Our goal is to identify the set of communities that C belongs and which has the following property: asking one of the communities in the latter set to be 'equivalent' to asking all of them, since if everybody tends to agree with everybody else in the set then simply asking one of them is sufficient.

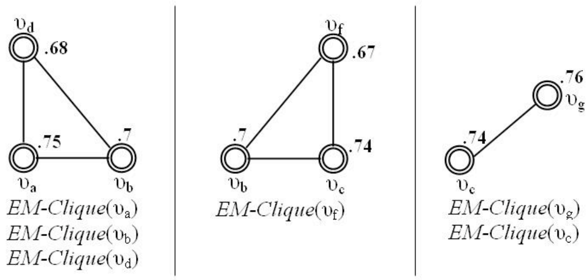


Fig. 4. The set of EM-Cliques of the Graph G

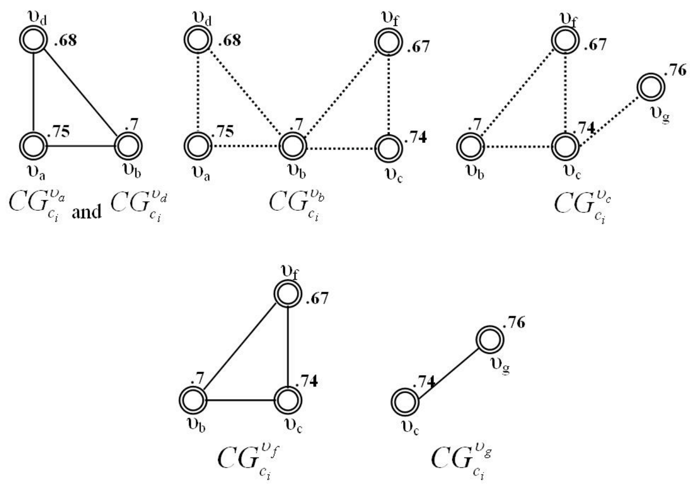


Fig. 5. The set of Convergence Graphs of Graph G_{C_i}

For example, the cliques the community C, which is represented by the vertex v_C in the consistency subgraph in Fig.3, participates are $\{\{v_b, v_c\}, \{v_c, v_f\}, \{v_c, v_g\}, \{v_b, v_c, v_f\}\}$. As we can clearly see, the cliques $\{\{v_b, v_c\}, \{v_c, v_f\}\}$ are induced subgraphs of the clique $\{v_b, v_c, v_f\}$, and thus the only candidate *EM-cliques* are: $\{v_b, v_c, v_f\}$ and $\{v_c, v_g\}$. In order to choose between the above two cliques we will use the (ϵ, μ)-domination condition. The *EM-cliques* of each of the nodes in Fig. 3 for $(\epsilon, \mu) = (0, 0)$ are depicted in Fig. 4. In particular, the node v_a participates in the following three cliques: $\{\{v_a, v_d\}, \{v_a, v_c\}, \{v_a, v_c, v_d\}\}$, and thus the *EM-clique* of the v_a is $\{v_a, v_c, v_d\}$, since the other two are its subcliques. Similarly, the cliques for v_d, v_f and v_g can be found. Regarding the node v_c, its *EM-clique* is $\{v_c, v_g\}$ since the clique $\{v_b, v_c, v_f\}$ does not have any subclique that can $(0, 0)$-*dominate* the clique $\{v_c, v_g\}$. In a similar way, the *EM-clique* of the node v_b is determined.

graph G	$avg(Pr^+(V_G))$	$avg(Pr^+(E_G))$
$G_1 = \{v_a, v_b, v_d, v_f\}$	0.7	0.825
$G_2 = \{v_a, v_c, v_b, v_g\}$	0.7175	0.7625
$G_3 = \{v_a, v_b, v_c, v_f\}$	0.715	0.7375
$G_4 = \{v_b, v_c, v_d, v_f\}$	0.6975	0.7125
$CG^{v_c}_{C_i}$	0.7175	0.7125

Table 1. The isomorphic to $CG^{v_c}_{C_i}$ (Fig. 5) subgraphs of $CG^{v_b}_{C_i}$ and the average Pr^+ probability of their edges and vertices.

The next step is to construct the *Coverage Graph* of each community C in the Consistency Subgraph. We will call the community C the 'owner' of the Coverage Graph. This can be done by merging all the *EM-Cliques* the community C participates in. If a community C participates in C_a's *EM-Clique* then asking C or C_a's is 'equivalent'. Thus, if the community C also participates in C_b's *EM-Clique* then asking C_b is equivalent to asking C. Consequently, in order to find how many communities each community 'covers' the *Coverage Graph* is created. For example, consider the *EM-cliques* in Fig. 4, the node v_b participates in v_f clique thus, its *Coverage Graph* is created by merging the its *EM-clique* with node's v_f *EM-clique*. The *Coverage Graphs* for the nodes in Fig. 3 are depicted in Fig. 5.

A first approach to identifying the first candidate advisor community could be to simply select the community whose *Coverage Graph* has the biggest order.[3] Assume now that the largest clique is G_1 and has order n and the second largest clique is G_2 and has order $n-1$. Furthermore, assume that the average vertex probability of G_1 and G_2 is 0.6 and 0.78, respectively, while the minimum probability of a vertex in G_1 is 0.5 and the minimum probability of a vertex in G_2 is 0.65. Obviously, although the order of graph G_1 is bigger than the order of G_2, the graph G_2 appears to be a better choice. For this reason, we choose to find the largest *Coverage Graph* that dominates all the *Coverage Graphs* of equal or smaller order, and thus we choose to apply the (ϵ, μ)-domination condition. We refer to the winner *Coverage Graph* as the *Dominant Coverage Graph*.

In order to determine the *Dominant Coverage Graph* between the *Coverage Graphs* of Fig. 5, we first examine the graph with the largest order which is $CG^{v_b}_{C_i}$. Clearly, since $CG^{v_a}_{C_i}$, $CG^{v_d}_{C_i}$ and $CG^{v_f}_{C_i}$ are induced subgraphs of $CG^{v_b}_{C_i}$, the graph $CG^{v_b}_{C_i}$ *strongly dominates* them. The first graph which we choose to check whether $CG^{v_b}_{C_i}$ (ϵ, μ)-dominates is $CG^{v_c}_{C_i}$. For simplicity reasons we consider $(\epsilon, \mu) = (0, 0)$. In order to decide whether $CG^{v_b}_{C_i}$ (ϵ, μ)-dominates $CG^{v_c}_{C_i}$ we have to identify the subgraphs of $CG^{v_b}_{C_i}$ that are isomorphic to $CG^{v_c}_{C_i}$. These subgraphs are $G_1 = \{v_a, v_b, v_d, v_f\}$, $G_2 = \{v_a, v_c, v_b, v_g\}$, $G_3 = \{v_a, v_b, v_c, v_f\}$ and $G_4 = \{v_d, v_b, v_c, v_f\}$. Table 1 summarizes the average probability of the edges and the vertices of each of the above subgraphs as well as the average probability on the edges and vertices of the graph $CG^{v_b}_{C_i}$. In the case of the graph G_2 we have $Pr^+_v(G_2) = Pr^+_v(CG^{v_c}_{C_i})$ while $Pr^+_e(G_2) > Pr^+_e(CG^{v_c}_{C_i})$, and thus $CG^{v_b}_{C_i}$ (0,0)-*dominates* $CG^{v_c}_{C_i}$. Now, we discard $CG^{v_c}_{C_i}$ and all the coverage

[3] Recall the order of a graph G is equal to the number of G's vertices.

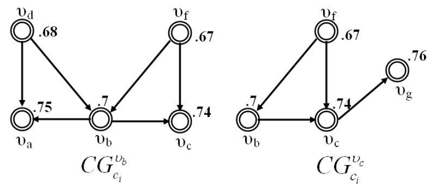


Fig. 6. ϵ-domination example

graphs that $CG^{v_c}_{C_i}$ (0,0)-*dominates*. In this particular case, given that the $CG^{v_g}_{C_i}$ is an induced subgraph of $CG^{v_c}_{C_i}$, $CG^{v_c}_{C_i}$ strongly dominates it and there is no other subgraph to be checked. This means that $CG^{v_b}_{C_i}$ is the *Dominant Coverage Graph*.

The algorithm for finding the *Dominant Coverage Graph* is as follows:

Algorithm 1 Finding the Dominant Convergence Graph

Input: Set S^* of coverage graphs,ϵ, μ
Output Dominant Convergence Graph if the set S^*

1: $S_{disc} \leftarrow empty$
2: FIND $g \in S^*$ with MAX Order[4]
3: **if** g $(\epsilon, \mu) - dominates$ all the graphs in S^* **then**
4:     **if** g is an induced subgraph of a graph $G' \in S_{disc}$ **then**
5:         RETURN G'
6:     **else**
7:         RETURN g
8:     **end if**
9: **else**
10:     REMOVE g from S^*
11:     ADD g in S_{disc}
12:     REMOVE all the graphs $g' \in S^*$ that g $(\epsilon, \mu) - dominates$ and add them in S_{disc}
13:     GOTO Step 2
14: **end if**

Considering the selection of which node in the *Dominant Coverage Graph* to ask, in an initial approach we could consider the selection of the community to which the *Dominant Coverage Graph* belongs. However, this is not always the best choice since the community that owns the *Dominant Coverage Graph* could be a community with one of the lowest probability among the other communities in this graph. For example, assume the *Dominant Coverage Graph* among the graphs in Fig. 5 is the graph $CG^{v_b}_{C_i}$. As we can see the vertex with the highest probability (0.75) is v_a while the owner of the graph is the vertex v_b.

Another approach we could consider is to select the node with the maximum probability. Assume now the *Dominant Coverage Graph* among the graphs in

Fig. 5 is the graph $CG^{v_c}_{C_i}$. As we can see the vertex with the highest probability (0.76) is v_g but the vertex v_c has lower probability but higher degree. Thus, given that these two probabilities are close enough, selecting the node v_c is a better choice. Thus, a more sophisticated approach is needed in order to select the 'strongest' node of a *Dominant Coverage Graph*.

We conclude that the approach we should use is the following: turn the *Dominant Coverage Graph* from undirected to directed by giving orientation to its edges. In particular, for each edge $e = (v, w)$ of the graph give a direction from w to v if $p(v) > p(w)$, from v to w if $p(v) < p(w)$ and a double direction if $p(v) = p(w)$, where p is the probability distribution over the vertices, and choose the node with the highest in-degree.[5] For example, assume that the *Dominant Coverage Graph* is the graph $CG^{v_c}_{C_i}$ by adding direction on its edges the graph in Fig. 6 is created. As we can see the node with the maximum in-degree is v_c. If we consider now $CG^{v_b}_{C_i}$ as the *Dominant Coverage Graph* we have three vertices with in-degree equal to 2 which are v_a, v_b and v_c. From these we select the one with the highest probability which in this case is the vertex v_a.

The algorithm for finding a list of advisors ordered from the most valuable to the least valuable is the following:

Algorithm 2 Finding the Ordered list L with the Advisors

Input: $G(V, E, Pr^*)$,ϵ
Output L

1: **for all** $v \in V$ **do**
2: FIND the *EM-Clique(v)* in $G(V, E, Pr^*)$
3: **end for**
4: **for all** $v \in V$ **do**
5: FIND the Coverage Graph $\hat{G}_v$
6: ADD $\hat{G}_v$ in list S
7: **end for**
8: Find the *Dominant Coverage Graph* $\hat{G}_{\tilde{w}}(Pr^*) \in S$
9: **for all** $e = (v, w) \in E(\hat{G}_{\tilde{w}})$ **do**
10: **if** $Pr^*(v) < Pr^*(w)$ **then**
11: GIVE edge e direction $v \rightarrow w$
12: **else if** $Pr^*(v) > Pr^*(w)$ **then**
13: GIVE edge e direction $v \leftarrow w$
14: **else**
15: GIVE edge e directions $v \rightarrow w$ and $v \leftarrow w$
16: **end if**
17: **end for**
18: FIND the node $k \in V(\hat{G}_{\tilde{w}})$ with the maximum in-degree.
19: ADD k at the end of the list L
20: REMOVE k from V
21: IF L not full AND G not empty THEN Go to step 1.

[5] The tie breaking policy is to choose the node with the highest probability.

Our approach guarantees that if the node with the highest probability is the owner of the clique it will be selected, while also guarantees that, in the example we just provided, the node v_c will be selected.

By using our proposed approach, each community is inclined to provide truthful reports. This is due to the fact that if a community chooses to lie this will decrease the probability of being selected in the future in the advisor list of other communities. For example, assume that a community C_i is interested in acquiring information regarding an agent a_j from a set of communities $S(a_j)$. Assume now that all the communities in $S(a_j)$ besides C^* are always honest. If C^* lies then the $Pr(v_{C^*})$ which captures the probability of the community C^* tells the truth will decrease.

If the new $Pr(v_{C^*})$ drops below the threshold that the Consistency Graph considers when deciding which nodes to discard, then the community C^* will not be included in the future in the *Consistency Graph* and thus it will not be asked. This is not desirable since, as we mentioned before, each community is interested in receiving requests in order to receive a payment which it can later use for paying other communities to provide information.

If the new $Pr(v_{C^*})$ is still above a threshold then, even in the best possible scenario where the community C^* is still part of the *Dominant Coverage Graph*, the probability of getting accepted as the 'strongest' community in the *Dominant Coverage Graph*, due to the fact that when $Pr(v_{C^*})$ decreases the probability of C^* having the maximum in-degree in the *Dominant Coverage Graph*, decreases. Furthermore, given that all the rest of the communities are honest their Pr increases and thus the probability of C^* getting selected decrease even faster.

If a subset S' of the communities in $S(a_j)$ also lies then the probability of a community in S' to participate in the *Dominant Coverage Graph* decreases. This is due to the fact the cliques these communities belongs to become weaker and thus the chances of participating in the final *EM-Clique* of a node decrease. This results in the chances of the communities in S' participating in the *Dominant Coverage Graph* to decrease. Of course, even if they still participate then any truthful community in the same clique as a lying community will have a bigger in-degree.

4 Discussion

In this paper we propose a graph-based approach for enabling the exchange of information between communities in such a way that the participant communities have strong incentives to provide truthful reports. In particular, we have developed a novel method for selecting communities to ask, in environments where communities seek to share information about their agents. The aim is to strengthen each community by having it retain its most effective contributors and to encourage agents to be good contributors, wherever they reside, in order to gain the benefits of community membership that their shared reputation ratings will provide.

Our proposed approach is intended to be used together with a novel payment mechanism that we developed [6]. In particular, our payment mechanism rewards honesty and provides fair payments among communities when they provide information about their agents to other communities. Our next steps will involve the integration of these two procedures, followed by formal proofs that the overall mechanism is incentive compatible and fair.

A variety of graph-based approaches have been developed to assist in the modeling of social networks of agents, in an effort to propose which parties an agent should consult, when the agent's own knowledge may be limited [3, 10, 13, 15, 16]. This may arise, for example, in settings where a buying agent consults other buyers in order to learn about the reputability of a selling agent [9] or in collaborative filtering based approaches for the design of recommender systems [11, 12] where an agent needs to determine the other parties whose recommendations will be taken into consideration. One approach that may be particularly useful for us to examine for future work is that of Wang and Vassileva [14]. Here, a Bayesian model is used to allow the trustworthiness of an agent to be determined on the basis of different capabilities (the variable learned by the model). In environments such as P2P file sharing, trustworthiness may differ for different areas of expertise (e.g. music file sharing vs. movie recommendation). It is important to note, however, that our focus is on modeling the trustworthiness of communities, not only determining the most trustworthy communities to ask but also providing them with strong incentives to continue being truthful in the future.

The issue of using trust and reputation as regulating tools is our primary focus. We want to share reputation ratings, but also to restrict the extent to which this information is shared; we provide a framework for making this possible. Architectures for social agents is also relevant. Our selection mechanism provides a method for locating a set of other communities with which information will be shared and as such determines the overall social structure of the entire collection of communities. Since the setting for our research is a payment-based framework for exchanging information about agents, the topic of agent-based communities and electronic institutions is also quite relevant. Finally, our research fits well with the theme of practical applications of agent organization systems. As mentioned, our approach is particularly well suited for the application of P2P file sharing, for instance.

In examining current research on trust and reputation modeling, the distinction of Castelfranchi and Tan [2] is relevant for our work. An agent may be first of all inclined to be trustworthy simply if agent trustworthiness is being modeled in the society, with penalties exacted for failing to uphold the necessary trust. This is the perspective adopted in many so-called prescriptive trust models [7] such as that of BRS [4] or TRAVOS [13], which respectively discard advice from agents deemed to be untrustworthy or discount the evidence that is provided. Castelfranchi and Tan [2] then suggest that if there is insufficient trust, some kind of control mechanism may then be required (for example requiring letters of credit). Unpredictable behaviour may still occur. In particular Kerr and Cohen chronicle a series of vulnerabilities that various trust and reputation modeling systems may exhibit [7] and are even able to demonstrate how well these vulnerabilities may be exploited by untrustworthy agents [8].

In our approach, we have discussed how the proposed selection procedure makes communities inclined to be truthful. Lying will make these communities less likely to be selected; as a result, they will lose the ability to earn payment. This is an important liability, as payment in our framework is required by each community in order to acquire new information about agents, in the future. As such, our approach to ensuring truthfulness is less one of deterrence and penalties, but instead more one of social control and reward – communities cannot continue to participate in the exchange of information with other communities, if their behaviour is untrustworthy.

For future research, it would be valuable to explore the possible contributions from research that is more focused on articulating norms in order to control the behaviour of information-sharing parties. One possible direction would be to examine the operationalisation of norms as proposed in [1]. This research aims to ensure the safety and stability of systems and as such is in the same spirit as the motivation of our own research, that of providing incentives for honesty in community-based multiagent systems.

Another valuable direction for future research is to examine the concern raised by Castelfranchi and Tan [2] that virtual communities are faced with the challenge of coping with agents that leave and return under a new identity – the issue of anonymity in virtual environments. This issue will be important when we move forward to formulate a model for representing agent trustworthiness, as part of the reasoning performed by communities. As discussed in [5] we believe that the level of participation of an agent within a community will be an important element to consider, beyond a simple consideration of trust.

References

1. Huib Aldewereld, Frank Dignum, Andrés García-Camino, Pablo Noriega, Juan Antonio Rodríguez-Aguilar, and Carles Sierra. Operationalisation of norms for usage in electronic institutions. In *AAMAS '06: Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 223–225, New York, NY, USA, 2006. ACM.
2. C. Castelfranchi and Y. Tan. The role of trust and deception in virtual societies. In *HICSS '01: Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS-34)-Volume 7*, page 7011, Washington, DC, USA, 2001. IEEE Computer Society.
3. Philip Hendrix, Ya'akov Gal, and Avi Pfeffer. Learning whom to trust: Using graphical models for learning about information providers. In *AAMAS '09: Proceedings of the Eighth International Conference on Autonomous Agents and Multi-Agent Systems*, 2009.
4. Audun Jøsang and Roslan Ismail. The beta reputation system. In *15th Bled Electronic Commerce Conference e-Reality: Constructing the e-Economy*, Bled, Slovenia, 2002.

5. Georgia Kastidou and Robin Cohen. Trust-oriented utility-based community structure in multi-agent systems. In *EDMS 2008: Workshop on Economic Models for Distributed Systems held in conjunction with the SIGAPP Mardigras Conference*, 2008.
6. Georgia Kastidou, Kate Larson, and Robin Cohen. Exchanging reputation information between communities: A payment-function approach. In *Twenty-first International Joint Conference on Artificial Intelligence, (IJCAI 2009)*, 2009.
7. Reid Kerr and Robin Cohen. Trunits: A monetary approach to modeling trust in electronic marketplaces. In *In Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'06) Workshop on Trust in Agent Societies,*, 2006.
8. Reid Kerr and Robin Cohen. Smart cheaters do prosper: Defeating trust and reputation systems. In *AAMAS '09: Proceedings of the Eighth International Conference on Autonomous Agents and Multi-Agent Systems*, 2009.
9. Kevin Regan, Pascal Poupart, and Robin Cohen. Bayesian reputation modeling in emarketplaces sensitive to subjectivity, deception and change. In *In Proceedings of AAAI-06*, 2006.
10. Jordi Sabater and Carles Sierra. Reputation and social network analysis in multiagent systems. In *AAMAS '02: Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 475–482, New York, NY, USA, 2002. ACM.
11. Aaditeshwar Seth and Jie Zhang. A social network based approach to personalized recommendation of participatory media content. In *In Proceedings of the International Conference on Weblogs and Social Media (ICWSM)*, 2008.
12. Xiaodan Song, Belle L. Tseng, Ching-Yung Lin, and Ming-Ting Sun. Personalized recommendation driven by information flow. In *SIGIR '06: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 509–516, New York, NY, USA, 2006. ACM.
13. W. Teacy, Jigar Patel, Nicholas Jennings, and Michael Luck. Travos: Trust and reputation in the context of inaccurate information sources. *Autonomous Agents and Multi-Agent Systems*, 12(2):183–198, March 2006.
14. Yao Wang and Julita Vassileva. Trust and reputation model in peer-to-peer networks. In *P2P '03: Proceedings of the 3rd International Conference on Peer-to-Peer Computing*, page 150, Washington, DC, USA, 2003. IEEE Computer Society.
15. Bin Yu and Munindar P. Singh. A social mechanism of reputation management in electronic communities. In *CIA '00: Proceedings of the 4th International Workshop on Cooperative Information Agents IV, The Future of Information Agents in Cyberspace*, pages 154–165, London, UK, 2000. Springer-Verlag.
16. Bin Yu and Munindar P. Singh. Detecting deception in reputation management. In *AAMAS '03: Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 73–80, New York, NY, USA, 2003. ACM.