Diagnosing a Team of Agents: Scaling-Up

Meir Kalech and Gal A. Kaminka
The MAVERICK Group
Computer Science Department
Bar Ilan University, Israel
Ramat Gan, Israel
{kalechm, galk}@cs.biu.ac.il
Tel-Fax: 972-8-9723471

Abstract. Agents in a team must be in agreement. Once a disagreement occurs we should detect the disagreement and diagnose it. Unfortunately, current diagnosis techniques do not scale well with the number of agents, as they have high communication and computation complexity. We suggest three techniques to reduce the complexity: (i) reducing the amount of diagnostic reasoning by sending targeted queries; (ii) using a light-weight behavior recognition to recognize which beliefs of the agents might be in conflict; and (iii) grouping the agents according to their role and behavior and then diagnosing the groups based on representative agents. We examine these techniques in large-scale teams, in two domains, and show that combining the techniques provides a diagnosis method which is highly scalable in both communication and computation.

1 Introduction

Agents in a team must be in agreement as to their goals, plans and at least some of their beliefs [1, 4, 9]. Unfortunately, they may come to disagree due to uncertainty in sensing, communication failures, etc. [7].

Once a disagreement occurs the agents should diagnose it and provide a solution. The diagnosis process identifies which agents are in disagreement and about what they disagree, so that they can negotiate and argue, to resolve the disagreements [4]. We refer to this kind of diagnosis as *social diagnosis*, since it focuses on finding causes for *inter-agent* failures, i.e., failures to maintain relationships between agents in a team. Social diagnosis stands in contrast to *intra-agent* diagnosis, which focuses on determining the causes for components within agents.

In this paper we focus on social diagnosis in teams of agents where the number of agents is scaled-up. Unfortunately, previous social diagnosis methods do not address large-scale teams, and have unacceptable communication and computation complexity [5, 7]. On the other hand, previous work on large-scale multi-agent systems did not address social diagnosis, instead focusing on fault detection [6], non-social, intra-agent diagnosis [3, 8], or general coordination [2].

We present two principles for reducing the communication and computation complexity in large-scale social diagnosis: Reduce the communicated information, and diagnose only a limited number of representative agents (instead of all the agents). We suggest three techniques which use these principles: (i) reducing the amount of diagnostic reasoning by sending targeted queries *(behavior querying)* (ii) using a light-weight behavior recognition method to recognize beliefs that may be in conflict *(shared beliefs)* (iii) grouping the agents by their role and behavior and then diagnosing the groups based on representative agents *(grouping)*.

We empirically examine these techniques in terms of communication and computation in two domains through hundreds of tests. The results show that using the shared beliefs technique is a key factor in reducing the runtime only in low number of agents (though with a large number of beliefs). In contrast, behavior querying significantly reduces both computation and communication time (compared to previous work). However, combining these techniques with the grouping technique provides a diagnosis method which is highly scalable in both communication and computation in large number of agents.

2 Related Work

A closely related area of work deals with failure detection, rather than diagnosis. Kaminka and Bowling [6] address social fault detection in large-scale teams. The detection capabilities they present can complement our methods, by triggering the diagnosis methods we present below once a failure has been detected. Other social failure detection methods have been previously proposed ([7]), but none address scalability issues.

Frohlich et al. [3] and Roos et al. [8] suggested diagnosis methods in distributed system. They considered the problem of communication and computation complexity, but their approach is different than ours. Frohlich et al. [3] suggested dividing a spatially distributed system into regions, each under the responsibility of a diagnosis agent. If the fault depends on two regions the agents that are responsible to those regions cooperate in making the diagnosis. This method is inappropriate for dynamic team settings, where agents cannot preselect their communication partners. Similarly, Roos et al. also [8] analyzed an agent-based approach for diagnosing distributed systems. But, their method assumes that there are no conflicts between the knowledge of the different agents. This assumption stands in contrast to a common fault of disagreement between agents in multi-agent systems, especially in teams.

In previous work [5] we focused on problems of disagreement between agents. We have shown that one can reduced the communication by centralizing the diagnosis, so all the agents may send their information to a single pre-defined agent who compares between these beliefs. However, in teams where the number of agents is scaled-up, the computation may be expensive. Moreover, we showed that further reductions in communications, based on using inference of other agents beliefs, is exponential in run time.

In this work we focus on tackling the complexity of communication and inference, to enable diagnosis of large-scale teams. We suggest new methods of social diagnosis, that reduce both the communication and computation. First we will describe the basics of teams and social diagnosis (Section 3), and then we will present our social diagnosis methods in large-scale teams (Section 4).

3 Social Diagnosis Basics

We focus on teams of behavior-based agents, since the control process of such agents is relatively simple to model, and we can therefore focus on the core communications and computational requirements of the diagnosis. A behavior is a software module that controls the actions of the agent. The behavior has preconditions and termination conditions. Once the preconditions are satisfied by the agent, the agent may select the behavior for execution. Once the termination conditions are satisfied execution stops.

Each agent has a decomposition hierarchy of behaviors, arranged from a general behavior at the top level to specific behaviors at the lower levels. The agent is controlled by a root-to-leaf path of behaviors (hereinafter *behavior path*).

In Figure 1 we show a hierarchy of a team. Each letter represents a behavior. An agent will select the behavior path (A,B,C) if their preconditions are satisfied.

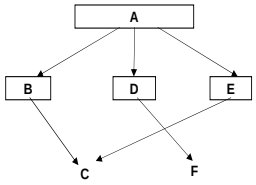

Figure 1. Team hierarchy

The designer of the agents declares in advance behaviors whose execution should be coordinated, i.e. all the agents should select these behaviors at the same time. We refer to these behaviors as "team behaviors" (the boxed behaviors in Figure 1). For instance, for a team of soccer-playing robots, the designer may declare the team behaviors to be *attack* and *defend*, where a precondition of attack is that one of the robots in the team gets the ball. Ideally, all the robots select the "attack" team behavior at the same time. This is typically achieved through a teamwork engine such as STEAM [9]. Each agent may select individually between non-team behaviors (unboxed in Figure 1).

Disagreement between team-members is manifested by selection of different team behaviors by different agents at the same time. This may happen due to sensor faults, communication failures, etc. Methods of detecting disagreements have been presented in [6]. In Figure 1 a disagreement will occur if agent X selects behavior path (A,B,C) while agent Y selects (A,E,C), since they differ in team behaviors (B and E).

The social diagnosis process identifies the disagreeing agents and the causes for their different selections (where the cause is differences in their beliefs). There are two phases: (i) selecting who will carry out the diagnosis; (ii) having the selected agents generate and disambiguate diagnosis hypotheses. It was previously shown in [5] that centralizing the diagnosis process is better than distributing it in terms of communication. So, in this paper a single diagnosing agent has been selected. To carry out the diagnosis it must identify the beliefs of the team members. Once the diagnosing agent knows the agents' beliefs it compares the beliefs and determines conflicting beliefs which account for the disagreement.

In our previous work, we have shown two algorithms: (i) reporting and (ii) querying. In the **reporting** algorithm all teammates communicate their beliefs to the diagnosing agent who compares them and finds the contradictions. In order to reduce communication, the diagnosing agent may use the **querying** algorithm to identify teammates' beliefs [5]. Querying proceeds in three stages (Figure 2). First, it observes its peers and uses a behavior recognition process (see below) to identify their possibly-selected behavior paths, based on their observed actions. Then, based on the hypothesized behavior paths it further hypothesizes the beliefs held by the teammates (which led them to select these behavior paths, by enabling sets of preconditions and termination conditions). Finally, it queries the diagnosed agents as needed to disambiguate between these belief hypotheses. Once it knows about the relevant beliefs of each agent, it compares these beliefs to detect contradictory beliefs which explain the disagreement in behavior selection.

The first phase of querying begins with **behavior recognition**. The diagnosing agent finds the behaviors that are associated with the observed actions of the diagnosed agents (a process with linear complexity in the number of behaviors, for each agent). This is done by maintaining behavior hierarchies for the other agents, and tagging all the behavior-paths that contain behaviors associated with observed actions. These tagged behavior-paths are used as hypotheses for the behavior-path actually selected by the observed agent.

For each one of the behavior-path hypotheses, the diagnosing agent then hypothesizes the beliefs that may account for it, a process known as **belief recognition**. These beliefs are those associated with the selection of the behavior over others (e.g., the behavior's preconditions and others' termination conditions). This process is exponential in the number of beliefs since we compute all the combinations of the possible belief values. For instance, if a pre-condition of an hypothesized behavior is $p \vee q$, it produces three belief hypotheses: (i)$p \vee q$ (ii)$p \wedge \neg q$ (iii) $\neg p \wedge q$.

Once the belief hypotheses are known, the agent can send targeted queries to specific agents in order to disambiguate the belief hypotheses. The same process is executed for each one of the observed agents.

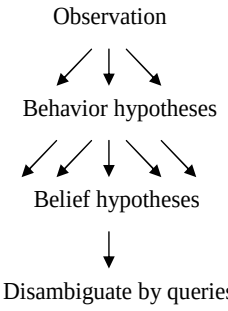

Figure 2. Querying process for a single agent

The querying algorithm is ill-suited for large-scale teams, mainly due to the exponential nature of its belief recognition component. In addition, the complexity of the belief comparison process (in both reporting and querying) is polynomial in the number of agents and beliefs, and is therefore problematic in large-scale teams.

4 Scaling Diagnosis Methods

We suggest three methods that tackle the runtime and communication complexities of querying. Each method tackles the complexity stemming for a particular factor in the complexity of querying: the number of behaviors, the number of beliefs, and the number of agents. **(i) behavior querying** eliminates the behavior recognition process by querying about the selected behavior path; **(ii) shared beliefs** limits the belief recognition process by inferring only the propositions of the beliefs, not their value; and **(iii) grouping** abstracts the diagnosed agents by grouping together agents along disagreement lines, and selecting representative agents for diagnosis.

4.1 Behavior Querying

Generally a behavior is associated with several beliefs through its preconditions and termination conditions. Therefore, we expect that the size of belief hypotheses will be greater than the size of behavior path hypotheses, since each behavior path hypothesis may generate several belief hypotheses as previously described.

We can eliminate the uncertainty in the behavior recognition process by disambiguating the observed agent's behavior path using communication, instead of inferring all its behavior path hypotheses. This goal is achieved by querying the observed agent about its behavior path. Once the diagnosing agent knows the behavior path of the monitored agent, it continues to build the belief hypotheses that are associated only with that behavior path. The advantage of this method is that by a single query about the behavior path of the observed agent, it eliminates all the queries about the belief hypotheses associated with other (incorrect) behavior path hypotheses.

We predict an improvement relative to the behavior path hypotheses process in term of communication, since we now expect to see only one message from each observed agent independently of the number of behavior path hypotheses. Also, we predict an improvement in terms of runtime since the behavior querying method eliminates the belief hypotheses computation of all the behavior path hypotheses except for the correct one. So instead of the linear complexity of behavior recognition (in the number of behaviors in the behavior hierarchy), the number of behaviors has no effect at all, and the resulting complexity is ($O(1)$).

4.2 Shared Beliefs

The main factor that causes a high runtime of the querying algorithm is the use of belief recognition process. This process grows **exponentially** in runtime with the number of beliefs associated with hypothesized recognized behavior paths. Even if the number of behavior path hypotheses is one, belief recognition will typically have multiple beliefs associated with it, and thus result in an exponential number of belief hypotheses.

We present a light-weight belief recognition technique whose complexity grows **linearly** with the number of beliefs. The key to this technique is to infer only the propositions associated with a belief, without hypothesizing about its value. In other words, the key is to infer that an agent has beliefs about p, without inferring what these beliefs are (p or $\neg p$). The diagnosing agent uses this technique to infer, for each agent, what propositions it holds. Then, for each pair of agents it queries for the values of propositions that are shared by the agent, and may thus be in conflict. For instance, if p is a proposition shared by agent A and agent B, a possible diagnosis is that agent A believes p while agent B believes $\neg p$. Thus the diagnosing agent should send a query to agent A and B about the value of p in order to determine if there is a contradiction.

Using this method, we expect that the communication will increase in the number of agents relative to the querying algorithm, since in teams we expect that most of the beliefs will be shared beliefs, so most of them are suspected. But, we expect to reduce the runtime complexity significantly, since instead of inferring all the exponential number of belief hypotheses, we use a process that is linear in the number of beliefs.

4.3 Grouping

Regardless of how knowledge of the beliefs of teammates is inferred, the diagnosing agent must compare between the beliefs of the teammates after inferring those beliefs. This comparison is polynomial in the number of agents and in the number of beliefs. However, in a large-scale team, runtime may be too high in practice.

The grouping method abstracts the observed agents, grouping together agents that are in a similar state. It then uses a single agent from each group as a representative for all agents in its group. To determine the diagnosis, it only compares the beliefs of these representative agents, thus significantly reducing the number of comparisons.

The process is based on the assumption that two or more agents that have both the same role in the team and the same behavior path will have the same beliefs, at least with respect to their selection of role and behavior path. Based on this assumption only representative agents of each role and behavior path must be diagnosed.

To determine the different role/behavior path combinations, the diagnosing agent first disambiguates the behavior path of each monitored agent using *behavior querying* process. It then divides the team to groups based in their roles and behavior paths. This essentially divides the team along disagreement lines. It continues to do the diagnosis process only against representative agents of each group (hereinafter: *representative agents*), either by querying algorithm or by shared belief methods. Finally, it uses the results of the diagnosis for the remaining members of the groups.

We predict that this process will reduce both the number of messages as well as the runtime, since the diagnosis process involves a significantly lower number of agents (only the representative agents of the groups), and likely this number is much smaller than the number of agents in the team (see the next section for an analysis of the maximum number of groups possible given a set of roles and behaviors). However, communications will still in the number of agents, though slowly, since the diagnosing agent has to disambiguate the behavior path of the agents by behavior querying in order to divide the team to groups.

The disadvantage of this method lies with its base assumption that agents in the same group will have the same beliefs, is not always correct. For instance, if the termination conditions of behavior Z are: $p \vee q$ then agent A may terminate this behavior because it believes that p is true (although q is false), while agent B which has the same role as A, may terminate the same behavior because it believes that q is true (although p is false). So, both of the agents terminate the same behavior and may select the same behavior although their beliefs are not the same. However, we believe that this case is very rare. In fact, it never occurred in our experiments.

5 Evaluation and Discussion

This section evaluates the scaling techniques we presented and draws lessons about their effects on computation and communication complexity. We compare several methods:

1. **behavior:** The diagnosing agent uses behavior querying. Once the behavior path of each monitored agent is known, the diagnosing agent continues to diagnose using the remaining phases of the querying algorithm.
2. **belief:** The diagnosing agent uses the shared beliefs method to generate belief hypotheses.

3. **behavior+belief:** This method combines the above methods. The diagnosing agent uses behavior querying to determine the behavior path of the observed agent, and then continues to diagnose the disagreements using the shared beliefs method.
4. **grouping:** The last method adds the grouping technique to the former one. Once the behavior path of each monitored agent is known using behavior querying, it divides the team to groups according to their role and behavior path, and continues using shared beliefs method against the representative agents of the groups.

We compare these methods to the original querying algorithm and to the reporting method, which relies on complete communication with no inference other than for the comparison step.

5.1 Real-world Domain

It would be useful to evaluate the techniques on a real-world large-scale multi-agent system, in order to determine their potential impact on realistic problems. We therefore took a real-world system, and determined its behavior in large-scale settings by simulating a greater number of agents than it originally had. The domain we chose was ModSAF, a virtual battlefield environment containing teams of synthetic helicopter pilots (described in [9]).

We performed experiments in which we varied the number of synthetic pilots from 2 to 150 (in jumps of 4). For each team size (n agents), we varied the selected behavior path of each agent, and the role of the agents (two roles, *scouts* and *attackers)*. We ran three sets tests: (1) one attacker and $n-1$ scouts; (2) $n-1$ attackers and one scout; (3) $n/2$ attackers and $n/2$ scouts. Overall, for every n agents, we tested close to 60 failure cases, varying the behavior paths (4 options) selected by the agents. For each single test we measured the number of messages sent and the runtime by each one of the diagnosis methods.

Figure 3 summarizes the results of these experiments. It compares the different diagnosis methods in terms of the average number of belief messages they utilize. The x axis shows the number of agents in the diagnosed team and the y axis presents the number of messages. Each data point is an average over approximately 60 trials.

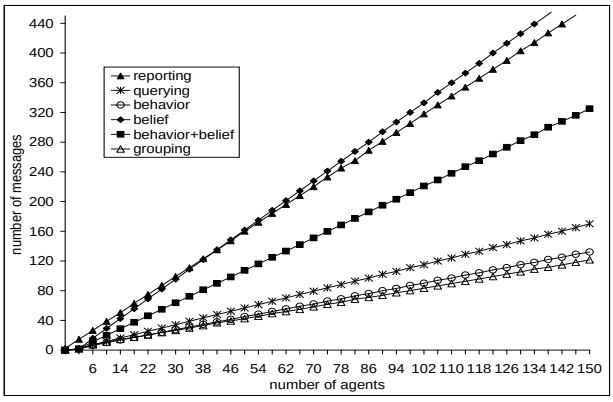

Figure 3. ModSAF: Number of messages

We can see that the growth of the shared beliefs method (*belief*) is very similar to that of the reporting algorithm (*reporting*). We believe that this is because in teams, the behavior paths selected by different agents refer to the same propositions, to a large degree. Thus the number of shared beliefs (that are then communicated) is in fact very close to the total number of beliefs (which are all communicated in the reporting method).

The behavior querying method (*behavior*) shows limited improvement relative to the querying algorithm (*querying*) graph. We believe this is because in the ModSAF domain there are only few possibilities of behavior path hypotheses and belief hypotheses, and as mentioned above (section 4.1) the benefit of this method is in the disambiguation of a high number of behavior path hypotheses and/or belief hypotheses by a single query.

The grouping method is better than the querying algorithm as shown in Figure 3, since the diagnosis communication is done only against the representative agents of the groups. Although the number of the representative agents is fixed through the tests, communication depends linearly on the number of agents since each added agent is queried about its behavior path. In an application with a high number of behavior path hypotheses and/or belief hypotheses we predict a significant growth in the querying graph in contrast to the grouping graph which will remain the same (since the communication growth is affected only by the queries that disambiguate the agents' behavior path).

Figure 4 presents the average runtime (in CPU milliseconds) of the different methods. The runtime of each test was taken as the maximum of any of the agents in the test. Surprisingly, the shared belief (*belief*) method grows much faster than querying. The reason for this is that the shared beliefs method compares all the beliefs that are associated with all of the behavior path hypotheses of all the agents, *before* disambiguating the beliefs' values. This is done to determine what propositions are possibly shared between agents, and may thus be in conflict. On the other hand, in the querying algorithm, the comparisons are done only between the beliefs of the agents *after* they already have been disambiguated, so only the actual beliefs of the agents are compared (although the inference preceding the querying is exponential in the number of beliefs).

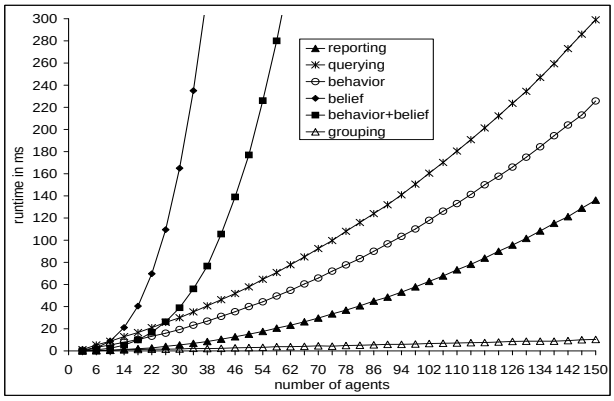

Figure 4. ModSAF: Run-time

The combination of shared belief and behavior querying methods (*behavior+belief*) shows a slight improvement with respect to shared belief alone (*belief*), since the comparisons are now done between the beliefs that are associated with only behavior path hypothesis of the agents (instead of all the behavior path hypotheses). However, the number of comparisons is still much greater than the number of comparisons in the querying algorithm, since all the hypothesized beliefs (of the single behavior path) are compared before disambiguating their value, while in querying algorithm the diagnosing agent compares the actual beliefs (after they have been disambiguated).

On the other hand, as expected, the behavior querying method (*behavior*) improves the runtime relative to the querying algorithm,

since it saves the belief recognition of all the beliefs that are associated with the behavior path hypotheses that have not been disambiguated as the correct one. However, it is still polynomial in the number of agents, since agents' beliefs are compared.

Undoubtedly, the significant runtime improvement is in the grouping method, since it reduces the complexity from polynomial to linear, as shown in Figure 4. The reason is that the number of representative agents is fixed (the product of the number of behavior hypotheses and the number of agents' roles), so the number of comparisons between their beliefs is fixed too.

The conclusion we draw from these figures is that while in general, runtime grows polynomially in the number of agents because of the comparisons, the grouping method reduces runtime to a slow linear growth due to the fixed number of comparisons. In addition, the reduced number of comparisons causes a reduction in the number of messages. On the other hand, it seems according to the figures that the other two methods, behavior querying and shared beliefs, have no contribution to reduce either the runtime or the number of messages.

5.2 Simulated Domain

The conclusions in the former section lead us to two questions: First, to what degree do the results of the grouping method depend on the characteristics of the ModSAF domain—low number of agent roles (two) and behavior paths (four)? And second, are there benefits to behavior querying and the shared beliefs methods?

In order to cope with these questions we examine the diagnosis methods while varying parameters such as roles and behaviors. To do this, we built a synthetic domain called TEST, in which we can control (1) the number of agents, (2) the number of roles, (3) the number of behavior path hypotheses and (4) the number of beliefs per behavior. Let us now tackle the above questions using the TEST domain.

Grouping Benefits. A key feature of the grouping method is that the number of representative agents is bounded from above, by the minimum of (i) the number of agents in the team, and (ii) the number of groups. Since groups are distinguished during diagnosis based on the combination of roles and selected behaviors, the number of groups, for any disagreement, cannot exceed the product of the the number of roles and number of behavior paths in the behavior hierarchy.

Figures 5 and 6 show the results from experiments testing this feature. In these experiments, we fixed the number of roles and the number of behavior-paths in the behavior hierarchy at six each. Since groups are distinguished based on role-behavior combination, the maximal number of groups is 36. We then ran the diagnosis methods in teams of 6–150 agents, where for each team size of n agents, we tested six disagreements. Each such disagreement was maximal (i.e., worst case), in the sense that every agent tried to select behaviors and roles different from its peers. For instance, for twelve agents, six roles are divided equally between the agents, and for each two agents that have the same role, they select different behavior paths. Overall, each data point in the figures is an average over these six trials.

Figure 5 shows the number of messages of the grouping method compared with querying and reporting. We can see that indeed in thirty six agents the linear graph of the grouping method changes its angle and grows much slower. The same phenomenon occurs in Figure 6, that shows average run-time in these experiments. The graph is polynomial as long as the number of agents is smaller than thirty six, then the graph becomes approximately linear since this number is fixed.

To answer the question above, we believe that the grouping method is well suited for large-scale teams. As teams get large, the number of groups (and therefore the number of diagnosed representative agents) is likely to be much smaller than the total number of agents in the teams, even if we assume that the complexity of the different agents (in terms of roles and behaviors) would also be higher than in the experiments above. The complexity of the grouping method is bounded from above by $\min$*(number of agents,number of behavior paths*number of roles)*.

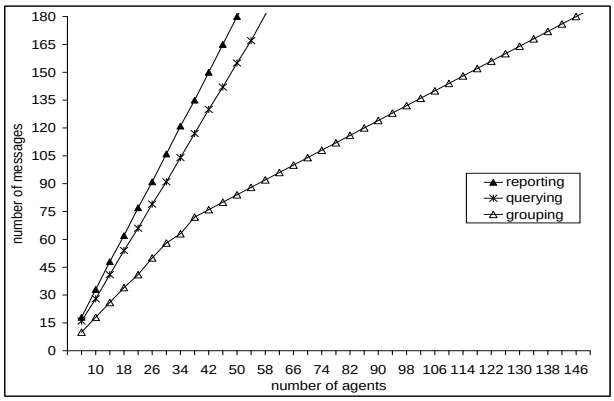

Figure 5. TEST: Number of messages in diagnosing of maximal disagreements

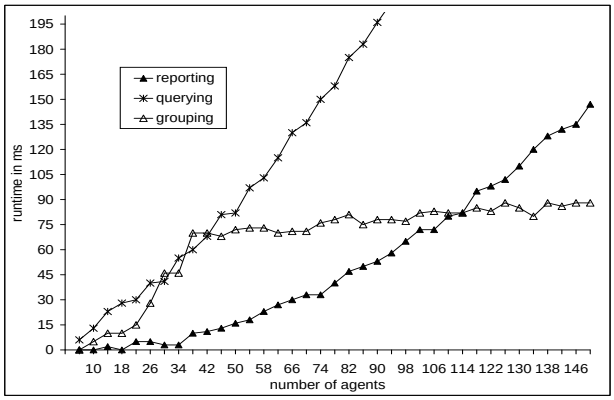

Figure 6. TEST: Run-time of diagnosing maximal disagreements

Let us turn to examining the benefits of the behavior querying and shared beliefs methods. We believe there are two ways in which these methods can be beneficial to the diagnosis process: First, by combining them with the grouping method; and second, in settings involving a large number of behavior path hypotheses and number of beliefs.

Combining the Three Methods. The grouping method is composed of two stages: Dividing the agents to groups according to their role and selected behavior path; and diagnosing the representative agents of the groups, where the results are assumed to hold for the other agents. In order to diagnose the representative agents in the second stage, we can use either the querying algorithm or the shared beliefs method. Since the number of diagnosed agents is relatively small (only representative agents are diagnosed), it is important to choose a method that works well in small teams.

In the experiments we ran in the ModSAF domain (previous section), we preferred the shared beliefs method. To evaluate this choice, Figures 7 and 8 show the communication and run-time results, respectively, of querying and shared beliefs, in diagnosing small teams (up to 20 agents, close to 60 trials per data point). We see that the two methods are close in terms of communications (Figure 7) while the shared beliefs (*belief*) is better than the querying in term of runtime (Figure 8). However, we remind the reader that with larger team sizes, querying runs faster than shared beliefs, and thus with a large number of groups generated by the grouping method, it may be preferable to diagnose representative agents using querying.

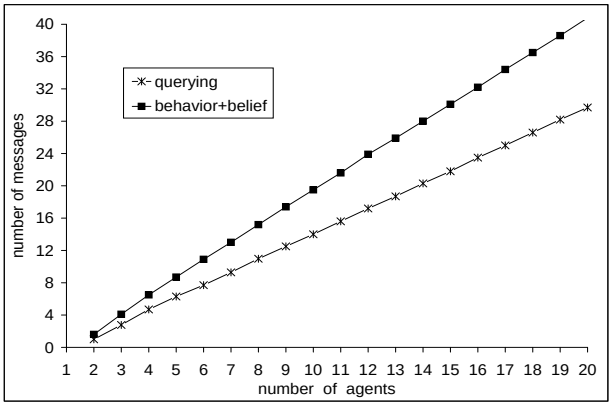

Figure 7. ModSAF: Average no. of messages

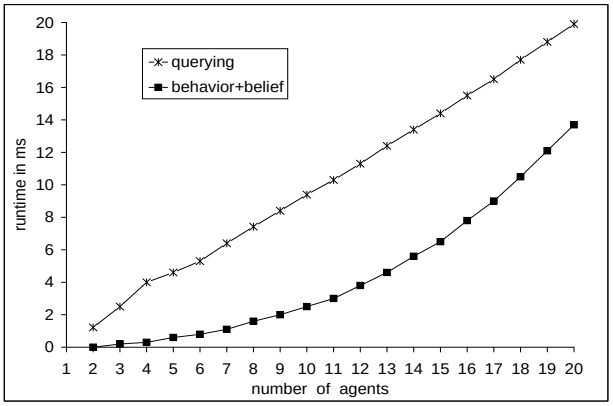

Figure 8. ModSAF: Average runtime

Shared beliefs Benefits. A second benefit of shared beliefs is in high number of beliefs and behavior path hypotheses. As mentioned in Section 4.2 the complexity of shared beliefs method is **linear** in the number of beliefs since the modelling agent infers only the proposition of the beliefs that are associated with the behavior path hypothesis, without reference to their value. This is in contrast to the querying algorithm that grows **exponentially** in the number of beliefs. However, this computational advantage did not manifest itself in the ModSAF domain, since in the ModSAF domain tests only the number of agents is varied where the number of beliefs is fixed.

To examine the effects of this difference between shared beliefs and querying, we compare them in settings involving a larger number of beliefs, in the TEST domain. In these experiments, the number of agents is fixed to fifteen, while we vary the number of beliefs from two to nine per behavior path. Figure 9 summarizes the results of these experiments (6 trials per data point). The x-axis shows the number of beliefs per behavior path and the y-axis shows the run-time in CPU milliseconds. Indeed we can see that while the querying graph grows exponentially, the shared beliefs graph grows very slowly. The implicit conclusion is that in a domain that involves a high number of beliefs, shared beliefs would be preferable to querying.

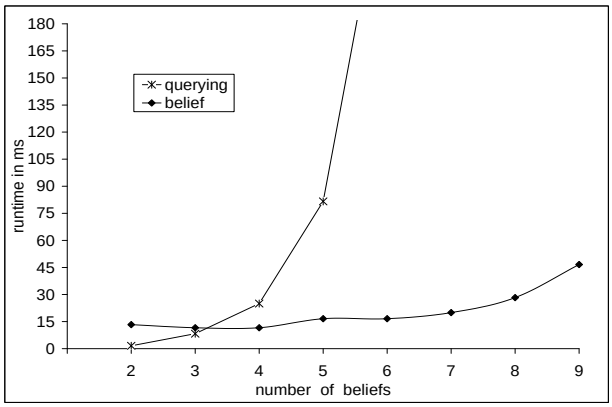

Figure 9. TEST: runtime in varying number of beliefs per behavior

Behavior Querying Benefit. The behavior querying method has a similar benefit, with respect to a high number of behavior path hypotheses. As we saw in Section 4.1, the number of messages in the querying method depends on the number of behavior path hypotheses. As the number of behavior path hypotheses grows, it typically multiplies the number of belief hypotheses, and this results in requiring many more queries to disambiguate the belief hypotheses. The intention behind behavior querying was to eliminate all behavior path hypotheses but one, by directly querying about the behavior path of the observed agent. In a domain where the potential number of behavior path hypotheses is small (e.g., only two in the ModSAF domain), the benefit of the behavior querying is not realized. Therefore, we examine it in the TEST domain. In this set of experiments, the number of agents is fixed at six, the number of beliefs per behavior is fixed at three, while the number of behavior path hypotheses is varied from two to ten.

Figure 10 summarizes the results of the experiments. The x axis shows the number of behavior path hypotheses, while the y axis shows the number of messages. Both the behavior querying method (*behavior*) as well as the grouping method (that relies on the behavior querying) are essentially constant in the number of sent messages, since once the behavior path of the observed agent is disambiguated the rest of the process depends on the number of agents and the number of beliefs, where these parameters are fixed here. On the other hand, the querying algorithm grows with the number of behavior path hypotheses. We conclude that behavior querying can be very beneficial in domains involving a large number of behavior path hypotheses.

6 Summary and Future Work

This paper presented novel techniques that enable scalability of social diagnosis in the number of agents: (i) Reducing the amount of diagnostic reasoning by sending targeted queries (*behavior querying*);

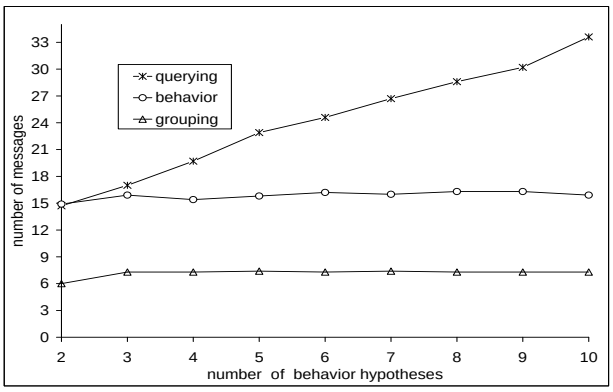

Figure 10. TEST: number of messages in varying number of behavior hypotheses

(ii) using a light-weight behavior recognition to recognize which beliefs of the diagnosed agents are relevant to the diagnosis (*shared beliefs*); and (iii) grouping the agents according to their role and behavior path and then diagnosing the groups based on representative agents (*grouping*). We empirically examine these techniques in two complex domains and concluded that the grouping method is very effective in large-scale teams since the diagnosis involves only a fixed number of representative agents. The behavior querying method is helpful in large number of behavior path hypotheses since it replaces multiple (incorrect) hypotheses using a single query, and the shared beliefs method is effective in large number of beliefs since it is linear in the number of beliefs. In this paper we focused on social diagnosis of inter-agent failures. In the future, we hope to examine ways of merging social diagnosis with intra-agent diagnosis methods.

REFERENCES

[1] P. R. Cohen and H. J. Levesque. Teamwork. *Nous*, 35, 1991.

[2] E. H. Durfee. Scaling up agent coordination strategies. *IEEE Computer*, 34(7):39–46, July 2001.

[3] P. Frohlich, I. de Almeida Mora, W. Nejdl, and M. Schroeder. Diagnostic agents for distributed systems. In *ModelAge Workshop*, pages 173–186, 1997.

[4] B. J. Grosz and S. Kraus. Collaborative plans for complex group actions. *Journal of Artificial Intelligence Research*, 86:269–358, 1996.

[5] M. Kalech and G. A. Kaminka. On the design of social diagnosis algorithms for multi-agent teams. *in International Joint Conference on Artificial Intelligence (IJCAI-03)*, pages 370–375, 2003.

[6] G. A. Kaminka and M. Bowling. Towards robust teams with many agents. *in Proceedings of Autonomous Agents and Multi Agent Systems (AAMAS-02)*.

[7] M. Klein and C. Dellarocas. Exception handling in agent systems. *Proceeding of the Third International Conference on Autonomous Agents*, May 1999.

[8] N. Roos, A. t. Teije, and C. Witteveen. A protocol for multi-agent diagnosis with spatially distributed knowledge. *in Proceedings of Autonomous Agents and Multi Agent Systems (AAMAS-03)*, pages 655–661, July 2003.

[9] M. Tambe. Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7:83–124, 1997.