Exploiting Multi-Agent Interactions for Identifying the Best-Payoff Information Source

Young-Woo Seo and Katia Sycara

School of Computer Science
Carnegie Mellon University
5000 Forbes Ave, Pittsburgh PA 15213, USA
{ywseo, katia}@cs.cmu.edu

Abstract

In many different applications on the Web, distributed agents would like to discover and access high quality information sources. This is a challenging problem since an agent does not know a priori which information source would provide high quality information for particular topics. In this paper, we utilize machine learning techniques to allow a set of distributed agents to use their past experience and collaborate with others to identify information sources with the best payoff. The proposed method allows an individual agent to estimate the next payoff based on its own history of interactions with the information source and also on collaboration with other agents whose individual analysis of the next payoff the agent trusts. Q-learning is applied for stochastic updates to the payoff. Experimental results show that the proposed method provides the best results when an individual agent collaborates with a moderate number of neighbors.

1. Introduction

The advent of ubiquitous computing and the network environment has facilitated access to information anytime and anywhere. At the same time, however, distributed agents find it increasingly difficult to locate an information source that provides information of high quality over time. In other words, since it is not always guaranteed that the best quality information source at the latest interaction provides the same (or even higher) quality of information at the next time step, it is difficult for an agent to find the information source that provides the highest quality over time. For example, a distributed agent is asked to find online bookstore which sells "Alice In Wonderland" at the lowest price. The agent might ask for the price from five different online bookstores that it has interacted with. The agent will pick the bookstore that offers the cheapest price as the best payoff information source because it would then receive a higher reward from the human user, assuming that the goal of a software agent is to maximize reward from the user. However this would not be a solution for the agent that is looking for the information source that produces the best payoff "over time," meaning that the long term payoff compensates for the short term loss. In particular, a buyer agent might be deceived by a malicious seller that intentionally provides good information for a short period, in order to get higher payments in the future while providing low quality information.

What if one of the book stores offers "Reinforcement Learning: An Introduction" at the lowest price due to noise in receiving the original query? This is another issue in finding the best-payoff information source. According to the the cheapest price criterion, this information might get the highest payoff, but the content of the information is not relevant at all to what the agent is looking for. Therefore, in order to obtain the highest payoff, the information should be qualified by both the information proximity and the information relevance.

Generally, it is highly unlikely for an agent to have complete knowledge of the environment because either an agent is deployed with an insufficient knowledge about the environment or it does not have much experience on a specific context. Therefore, it is unreasonable to assume that the information an individual agent collected satisfies both criteria. In particular, what if an online bookstore that the agent is not aware of (or has never interacted with) offers the cheapest price? It is quite risky to only rely on a distributed agent's own experience to find the best-payoff information source due to an insufficient amount of firsthand experience. Reliability and high quality information would be guaranteed if there is an authoritative third party that can collect all the necessary information so that it can easily evaluate the goodness of information sources. However the existence of such centralized entities is quite unrealistic and unmanageable in a distributed multi-agent communities. Therefore, in order to distinguish the information source consistently providing a better quality information from others temporarily supplying a high quality, it is highly desirable for the individual agent to evaluate its own experience intelligently

while collaborating with other agents to augment its partial observations. Such problems hold in finding a best-quality information over time from distributed resources such as sensor networks, P2P networking, daily search scenario, etc.

In this paper, we propose a new method of identifying the best payoff information source over time by exploiting a series of past interactions. A payoff is a compensation for an interaction between an agent and an information source (e.g., the book price offered by an online bookstore). We assume that the payoff is not generated randomly, but is drawn from an unknown function. However, this problem cannot be solved by polynomial approximation techniques nor probability density estimation because the function of payoff is time-variant and non-monotonic. As an individual agent is responsible for its estimation of the best payoff and its action selection, it maintains a series of payoffs as results of past interactions so that it can estimate the next payoff individually by using Q-learning. Since an individual agent does not have firsthand experience with all existing information sources, collaboration is required between heterogeneous agents, in order to augment a partially observed interaction history of information sources. The underlying idea is that an individual agent rates experience by discounting past experience and utilizing neighbors' opinions based on the agent's differential trust in its neighbor's opinions.

This paper is organized as follows. Section 2 investigates related work in terms of multi-agent interactions. Section 3 details the proposed method of estimating the next payoff by combining direct and indirect experience. Section 4 explains the experiments in detail and discusses experimental results. Section 5 provides our conclusions and future work.

2 Related Work

The issues of interaction among software agents have partially been tackled by the concept of trustworthiness. It is assumed that each individual agent is working towards a given goal under a complicated strategy on behalf of a human user. In order to achieve the goal, collaboration is important, but honesty or trust among heterogeneous agents is not guaranteed. Thus the ability to reason about trust is necessary.

There exits substantial research work in estimating the trustworthiness of other agents by combining direct and indirect experience [1], [7], [11]. These approaches are similar to ours in that direct experience is combined with others' statements to estimate trust of the target agent. They utilized different techniques (i.e., a set-theoretic approach, a probabilistic approach, and an ad-hoc approach) in order to implement the idea that the information from unreliable (or untrustworthy) agents must be ignored whereas the information from trustworthy agents must be highly regarded. However these approaches do not address the problem of collaboration between heterogeneous agents, to find the best-payoff information source over time.

The interaction between agents has also been studied from the perspective of social network analysis. Sabater and Sierra [5] utilize "clique" to collect information about the target agent from neighbors. Yu and Singh developed a distributed reputation management model which updates reputation with experience in a social network of trust based on referrals [11]. Sullivan and his colleagues [8] evaluate the effectiveness of socially conscious agents who care about their own reputation in the community for generating high payoffs in collaborative groups. These approaches seem to be natural for the formulation of a multi-agent community because social network analysis works on relational data which are obtainable from interactions between agents. However a problem might occur when we exploit social network analysis in that these techniques can only be applied to centralized information.

Reinforcement learning has served primarily as one of tools for modeling agents' interactions. Littman [4] proposed Min-Max Q-learning to model interactions between two agents using game theory. Santana and his colleagues [6] presented the multi-agent patrolling task for surveillance. In particular, Q-learning is applied to facilitate an effective coordination among individual agents. Each individual agent chooses its action in order to minimize the cost of travelling a terrain-graph. It is necessary for an individual agent to collaborate with others in order to overcome partial observability of the given environment. With regard to this, their approach is similar to ours in that they try to approximate global problems (i.e., optimization) by combining local ones. Kagan and his colleagues [3] investigated the usefulness of the "COllective INtelligence (COIN)" framework, which enables individual agents to pursue their goal in compliance with the community-level goal. Our work is different from COIN in that individual agents in our work only share their information to achieve their own goals without concern for a community.

3 Expectation of Payoff by Combining Direct and Indirect Experience

3.1 A Framework for Interactions in Multi-Agents Community

Figure 1 shows a framework for interactions in a distributed multi-agent community. There are n distributed agents and m information sources. Each distributed agent is deployed alone with very limited knowledge such as how to contact a limited number of neighbors and information sources. For instance, $Agent1$ is initially aware of three information sources and two peer agents. In this framework, there are only two types of interactions: interactions for Question-and-Answer (QnA) and interactions for Knowledge-Sharing (KS). A QnA interaction will happen between a distributed agent and an information source (e.g., an information source responds to a distributed agent's query) whereas a KS interaction will occur when two distributed agents share their experience about the information

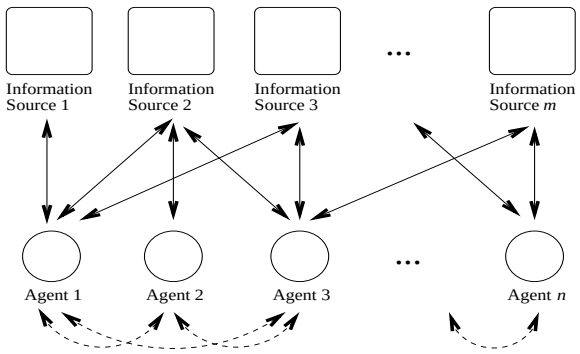

Figure 1. Interactions in distributed multi-agents community. A solid line represents an interaction for Question-and-Answer while a dashed line describes an interaction for Knowledge-Sharing.

source. [1] When a QnA occurs, the agent evaluates the quality of information according to its own criteria and it regards the quality of information as a payoff for the interaction. As described before, in order to obtain the highest payoff, the information should be qualified by both the information proximity and the information relevance. There are other aspects to be considered as factors of information goodness, such as a time constraint, however, we restrict our measure of information quality to proximity and relevance.

Our hypothesis is that the set of all individual agents' experience in a community is a subset of the actual interaction history of a particular information source. Accordingly, by interacting with other agents, an individual agent could have more comprehensive knowledge about an information source before actual interactions. To this end, individual agents record a series of past interactions both with information sources and with other distributed agents. While estimating the next payoff individually, an individual agent rates its own experience by utilizing an exponential smoothing of past experience (i.e., the highest importance is assigned to the latest experience). Such analysis of individual experience will be exchanged with the neighbors so that an individual agent could augment its estimate from partially observed history. However, when collecting the experience of others, the individual agents would not completely trust other agents because it is possible that they are competing for the same goal and thus there might be intentional deceptions. That is, they only trust on each other based on their previous experience – how good their past interactions were.

[1] For the simplicity, we do not concern about issues of the actual communication between distributed agents in detail, such as network topology and particular protocols.

3.2 Modeling of Interaction

3.2.1 Individual Expectation from Direct Experience

Reinforcement learning [9], particularly, Q-learning [10] is a good method to model an individual agent's experience with an information source over time for estimating the next payoff. In particular, firstly, Q-learning allows a sequential decision according to an agent's previous experience (i.e., Markov decision process). Secondly, there is a feedback (or reward) function that eventually influences an individual agent's decision – an agent will modify his policy of choosing an action in favor of a higher payoff QnA interaction. Furthermore, the goal of an agent is to identify the information source for which the payoff will be the maximum of accumulated payoffs.

The state space is a distributed agent's internal belief state of information sources with which the agent has interacted. Each state is a "mass" distribution because a payoff from a QnA interaction with a particular information source is independent of others. However a series of payoffs by a particular information source is continuous.

Given its internal state of the payoff history (s_t), the agent chooses (a_t) a particular information source as the next interaction partner. According to its selection of an information source, a feedback (i.e., a payoff for the interaction with the selected information source, r_t) will be given and the agent's internal state will change to reflect its action selection (i.e., to transit to another payoff mass distribution). This reward signal is delivered to an individual agent non-deterministically, meaning that a selection of the same action in the same state on two different trials may result in different next states [2]. As there is no information about the environment, such as a reward function and a state transition function, Q-learning, which is a model-free reinforcement learning algorithm [9], is suitable for our scenario.

The individual expectation for the next payoff with the kth information source is defined by:

$$\begin{aligned} Q_\delta(k, t+1, c) &= \sum_{t=0}^{\infty} \gamma^t Payoff(k, t, c) \\ &= Q_\delta(k, t, c) + E\{Payoff(k, t+1, c)\} \end{aligned} \quad (1)$$

where $Payoff(\cdot)$ is a payoff function that generates a real number for a question-and-answer (QnA) interaction. As described earlier, information given by a particular source is evaluated by information proximity and relevance. This is why the context, c, is a parameter of the payoff function because it is unreasonable to evaluate an information source providing a weather forecast by the price of a book. $E\{Payoff(k, t+1, c)\}$ is an expectation of the next payoff that averages discounted payoff over the previous h interactions under the context c with the kth information source:

$$E\{Payoff(k, t+1, c)\} = \{Payoff(k, t, c)$$

$$+\beta\left\{\frac{1}{h}\left(\sum_{l=1}^{h}\gamma^{l}Payoff(k,t-l,c)\right)-Q_{\delta}(k,t,c)\right\} \quad (2)$$

where $Payoff(k,t,c)$ is the latest payoff, β is a learning rate, and γ is a receding factor for discounting the payoff of previous interactions. Each of the previous interactions is discounted exponentially by the corresponding number of steps from the current one. This kind of exponential smoothing is reasonable in that experience close to the current time is more likely to happen again in the near future, accordingly receiving the highest importance. Technically, when γ is close to 1, dampening is quick and when γ is close to 0, dampening is slow. An individual agent keeps updating his estimation for the next time payoff by augmenting from the initial value as much β as the difference between the current payoff and the estimated payoff.

3.2.2 Expectation from Indirect Experience

It would be most desirable for an agent to have a complete interaction history for a particular information source when estimating the next payoff. However, it is not possible to collect all the interactions due mainly to the fact there is no way for an individual agent to know everything about an information source in the absence of a centralized entity. In order to augment its partially observed experience, an individual agent asks neighbors to share their experience of a particular information source. Each neighbor agent individually estimates the payoff for the next QnA interaction with all known information sources and then shares this estimation with others. In particular, a neighbor agent will use equation 1 if it is homogeneous. Otherwise the heterogeneous neighbors will use their own method for payoff estimation.

$$Q_{\Delta}(k,t+1,c) = \sum_{j=1}^{|neighbor|}\omega_j u_j(k,t,c) \quad (3)$$

where

$$\omega_j = \frac{\{\omega_j+\eta(1-|Payoff(k,t-1,c)-u_j(k,t-1,c)|)\}}{\sum_{h\neq j}^{|neighbor|}\omega_h}$$

where $\eta \in [0.0, 1.0]$ is a constant for the update rate, ω_j is a weight that affects the degree of an agent's reliance on the statements from neighbor agents, and $u_j(k,t,c)$ is a statement that is the $agent_j$'s individual estimation for the expected payoff from the next time interaction with the kth information provider under the context c. The statement from the jth agent is regarded important if the previous statements were trustworthy (i.e., the previous statements were close to the actual previous payoff for interaction with a particular information source).

With the combined estimates, an individual agent expects the best-payoff information source at the next interaction $(t+1)$ by using:

$$IS_{best}(t+1,c) = \arg\max_{k}\{(1-\alpha(t))Q_{\delta}(k,t+1,c) + \alpha(t)Q_{\Delta}(k,t+1,c)\} \quad (4)$$

where $IS_{best}(t+1,c) \in [0.0, 1.0]$, $Q_{\delta}(k,t+1,c)$ is an individual estimation for the expected payoff from the past interactions with the kth information source under the context c and $Q_{\Delta}(k,t+1,c)$ is an expected payoff from neighbors. $\alpha(t) \in [0.0, 1.0]$ is a factor that determines the degree to which an individual agent relies on its neighbors' statement. Since each individual agent does not initially have enough firsthand experience with a particular information source, it is desirable to keep α large at the beginning of interactions and to decrease α over the course of time. This reflects the fact that over time an agent becomes more confident of its own opinion based on direct experience and consequently its dependency upon other agents' opinion decreases. Thus α should ensure that the degree of dependence on others' experience decreases at every iteration. In particular, α is updated by $\alpha(t) = \alpha(t_0) \times \left(1.0 - \frac{t}{T}\right)$, where $\alpha(t_0)$ is the initial value of reliance and T is the total number of iterations.

4 Experiments

4.1 Experimental Setting

Many suitable transaction datasets exist (e.g., transaction records of eBay.com or of Amazon.com). However, we were unable to find a publicly available one and as a result constructed our own.

Let us suppose you have a list of concepts and their brief descriptions. A brief description represents what you currently understand from the concept. In order to comprehend a given concept clearly, you may want to consult several different web-search engines for further information. Each web-search engine provides a different quality of information (e.g., the degree of user's satisfaction about proximity and relevance of the returned URLs to a query).

To this end, we made use of the course list in the School of Compute Science, Carnegie Mellon University. Figure 2 shows an example of a course description that is usually comprised of the name of instructor, the date of class, the schedule, a brief description of the course, etc. A number of keywords [2] extracted from a course description were used as a query for each of the web-search engines. For example, the five words, "learning", "data", "class", "mining", and "machine," were chosen as keywords for the course in figure 2 for a web search. Each distributed agent is asked to seek the detailed information about a given course with a set of extracted keywords and an information source (i.e., a web-search engine) responds to the question by providing information (i.e., the 1 top-ranked url). The similarity [3] between the course description and the web-document pointed

[2] To this end, first we compute the weight of each word (unigram) according to its frequency in the course description and normalized it by the total number of words in that description. The n top ranked words were then returned as keywords of the course description.

[3] A course description and a downloaded web-document are represented in a multi-dimensional vector space. The similarity between two vectors is measured by the cosine angle between them.

Course Number: *xx-xxx*
Course Title: *Machine Learning*
Instructor: Prof. *XXX*
Units: *12.0*, Semester: *Fall 2004*
... The study of learning from data is commercially and scientifically important... This course is designed to give a graduate-level student a thorough grounding in the methodologies, technologies, mathematics and algorithms currently needed by people who do research in learning and data mining or who may need to apply learning or data mining techniques to a target problem. ...

Figure 2. An example of a course description.

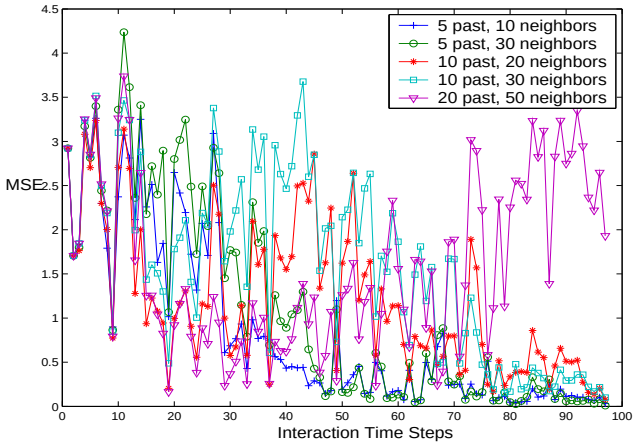

Figure 3. The size of memory is set from 5 to 20 and the number of neighbors is set from 10 to 50.

by the 1 top-ranked url is regarded as a payoff of this QnA interaction. [4]

There are 100 course descriptions collected from the homepage of the School of Computer Science, Carnegie Mellon University and there are 6 different web-search engines used as information sources [5]. As a result, in this simulated multi-agent community, there are 100 distributed agents in which each of them has 100 QnA interactions with one of six information sources under the same context (i.e., course information). In particular, at each interaction time step, each individual agent is asked to predict the payoff of a QnA interaction with an information source before the interaction actually happens. To this end, individual agents first estimate the payoff from their direct experience with information sources by using Equation 1. They then exchange their individual expectation with neighbors (Equation 3). And finally each of the individual agents predicts the next payoff by combining its own expectation and its neighbors' expectations with Equation 4. The error is then measured by calculating the difference between the predicted payoff and the actual one. Note each of the experimental results is an average of 5 different runs.

4.2 Experimental Results

A number of experiments were carried out to verify the usefulness of the proposed method. We would like to test our algorithm with respect to the following:

- *Cooperative or Working-Alone* Which distributed agent is better, one which is willing to cooperate or one which prefers to work alone. A "working-alone" agent is working alone to achieve a given goal from the beginning. It is different from an agent who is willing to collaborate with others at the beginning and then prefers to work alone when it has enough experience.

- *Optimal Number of Collaborators* If a collaboration is recommended, then how many collaborators does an individual agent need to achieve a given goal. This may shed light on the issue of the importance of how an agent's social network should be.

- *Usefulness of the Proposed Method* We would like to see how useful the proposed method is when it is used alone, even though our initial hypothesis includes the expectation that indirect experience compensates for lack of direct experience.

At each iteration, an individual agent chooses an information source according to its action selection strategy and receives a payoff in real numbers as a reward for its action selection. α decreases constantly over the course of interaction until an individual agent makes a decision all by itself. We heuristically assigned 0.9 to γ and 0.7 to α_0, respectively. We test three standard policies as candidates of an action selection strategy: *greedy*, ϵ*-greedy*, and *softmax* [9]. We found that a ϵ-greedy shows the best results. The value of ϵ is heuristically determined to 0.2, meaning that with 0.2 probability an individual agent chooses an information source greedily (i.e., exploit the current knowledge), but 0.8, $(1-\epsilon)$, selects an information provider at random (i.e., explore an unknown, but promising information source), independent of the estimated payoff. Greedy and a softmax action selection showed an unstable result because there are not much iterations for both of the methods to converge to a certain point.

Figure 3 shows the mean squared error (MSE) between the predicted payoffs and the actual ones. For this experiment, the number of neighbors is assigned from 10 to 50 and the size of memory for remembering payoffs of past interactions is set from 5 to 20, respectively. Over the course of interactions, MSE decreases, meaning that indi-

[4] While downloading the URLs, we eliminated the homepage of the course to be compared, duplicate pages between two different web-search engines, web-pages not in plain text (e.g., pdf, lecture slide in binary format, and others.), and advertisements.

[5] The web-search engines tested include Google, Altavista, Yahoo, Excite!, Vivisimo, and Alltheweb.

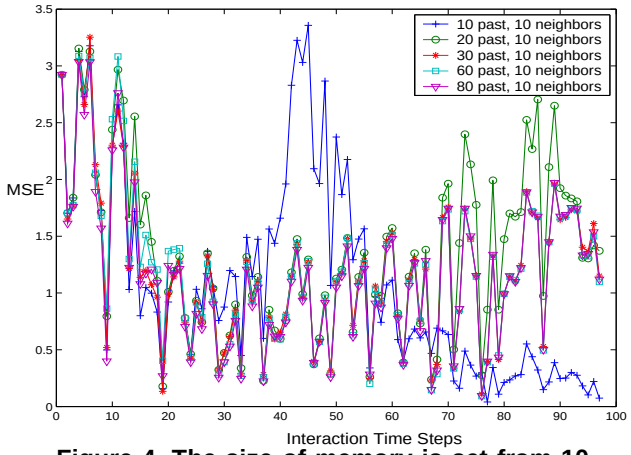

Figure 4. The size of memory is set from 10 to 80 and the number of neighbors is fixed to 10.

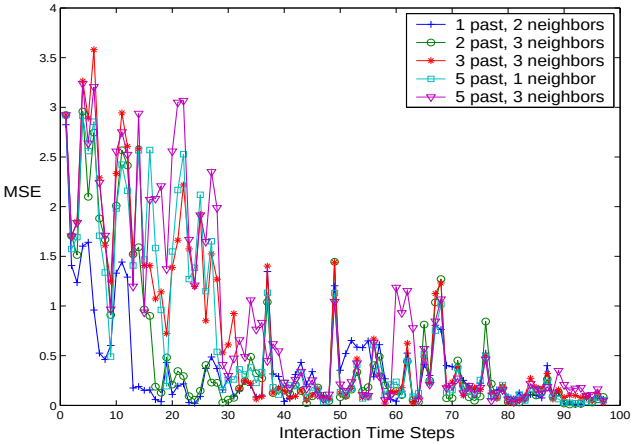

Figure 7. The size of memory is set from 1 to 5 and the number of neighbors is less than 5.

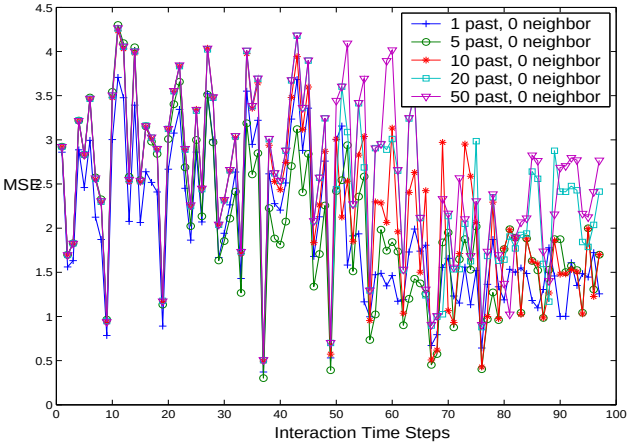

Figure 5. The size of memory is set from 10 to 50 and no neighbors is available.

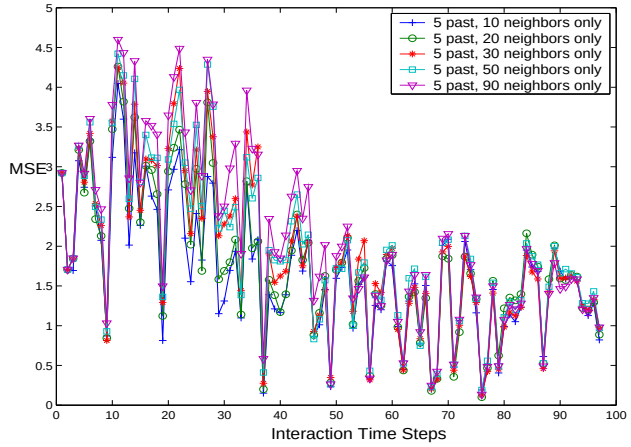

Figure 6. The size of memory is fixed to 5 and the number of neighbors is set from 10 to 90.

vidual agents became experienced with the given set of information sources and their expectations got closer to the actual payoffs. This trends holds good except the one with 50 neighbors. One interesting fact found is that the assignment of a more memory does not necessarily help an individual agent find the best payoff information source. In particular, there seems to be a confusing period where there are so many previous experiences to remember. For approximately the period from the 30th iteration to the 60th iteration, there is a large fluctuation of the MSE. After this, expectations by individual agents became unstable. Santana and his colleagues [6] reported a similar situation that a decrease of performance is observed when the growth of information by collaborators is accompanied with the noise in decision making. In our case those confusing periods eventually settle down and the rate of expectations converge. We believe the statements from neighbors compensated individual expectation well, even though their effects are diminished according to α. However the rate of expectation did not converge if an individual agent remembered more than 20 past interactions. Figure 4 confirmed this observation that individual agents got confused about the true payoff function if there are more than 20 past interactions to remember, even with the large number of neighbors.

In the experiment shown in figure 5 we assumed that all the statements from neighbors are false and accordingly no collaboration is recommended. The result shows that the proposed method is useful for individual agents to estimate the next payoff without exchanging experience. However the rate of convergence is quite slow and the rate of fluctuation is high. For the case where there are no collaborators available, it is still possible for an individual agent to work alone with the proposed method, but it is desirable to work with others, in order to rapidly figure out which is the best payoff information source.

Let us assume that an individual agent has a sufficient amount of direct experience, but it notices that all of them are incorrect. The experiment in the figure 6 testified such a situation in which an individual agent tried to approximate

the unknown payoff function of information sources by only using neighbors' statements. The result confirms one of our initial hypothesis that the statements from neighbors could be used as compensation for a partially observed history.

From the last two experiments we conclude that our method can be used alone, however, collaboration is required for a reasonable convergence rate. Thus in the final experiment, the results of which are shown in figure 7, we assigned a relatively smaller number to each parameter; the number of neighbors is assigned from 1 to 3 and the size of memory is set from 1 to 5. Although they look quite similar, the results are best for the one with 1 past/2 neighbors, then 2/3, 5/1, 3/3, and finally 5/3, in terms of average error. Let us suppose that an unrealistic centralized entity exists and it could make zero-error prediction for the next-payoff. The obtained results are promising in that they are very close to the optimum (i.e., zero-MSE) by utilizing only the small number of available resources with the size of memory as 1 to 5 and the number of neighbors as 1 to 3. In particular, the average coefficient of determination of the best one over six information sources is 78%.

5 Conclusion and Future Work

In this paper, we proposed a new method for identifying the best payoff information source over time by exploiting a series of past interactions. The underlying idea is that a distributed agent in a multi-agent community utilizes its own experience by recency. To this end, Q-learning is applied to estimate individual expectation for the next payoff by using exponential smoothing. Since a set of direct experience is a partially observed history of interactions, it is desirable for an individual agent to collaborate with others based on trust. The statements of other agents are assigned importance based on the number of past interactions where the other agents provided reliable information.

Combining the experimental results and analysis leads to the following conclusions:

- With the proposed method, an individual agent is able to figure out the best payoff information source by direct observations only.
- However, collaboration based on trust is desirable for an individual agent to augment its direct observations.
- The total number of observations is an important factor for reasonable convergence rate. That is, the number of neighbors for collaboration should be moderate (in our case, less than 3) and the size of memory for remembering past experience is also moderate (in our case, less than 5).

Since it is unclear that our own data set has actually been generated by a continuous function, an immediate future work is to test the proposed method to real-world data sets that are collected from an actual time period, such as transaction history by online stores, sensor reading data from a sensor network, or P2P file-sharing/search results. Furthermore, our future research aims to extend the proposed method for the interaction scenario under multiple contexts. Our experiments are carried out under the condition that there is one context available and accordingly the goodness of information is evaluated by proximity, assuming that all information is relevant. However it is unrealistic setting because an information source could provide service more than one context (e.g., Amazon.com) and responses to a particular query are mostly irrelevant. We also plan to incorporate the previous work on trust estimation between peer agents into our method.

6 Acknowledgement

This work was supported by ARDA under the CTA subcontract and by DARPA grant F30602-03-C-0009. We would like to thank Michael Stilman, Robin Glinton and Sean Owens for their friendly and responsive comments.

References

[1] A. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. In *Proceedings of 33rd Hawaii International Conference on System Sciences*, 2000.

[2] L. Kaelbling, M. Littman, and A. Moore. Reinforcement learning: A survey. *Journal of Artifical Intelligence Research*, 4:237–285, 1996.

[3] A. Kagan, T. adn Agogino and D. Wolpert. Learning sequences of actions in collectives of autonomous agents. In *Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS-2002)*, pages 378–385, 2002.

[4] M. Littman. Markov games as a framework for multiagent reinforcement learning. In *Proceedings of the International Conference on Machine Learning (ICML-1994)*, pages 157–163, 1994.

[5] J. Sabater and C. Sierra. Reputation and social network analysis in multi-agent systems. In *Proceedings of International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02)*, pages 475–482, 2002.

[6] H. Santana, V. Corruble, and B. Ratitch. Multi-agent patrolling with reinforcement learning. In *Proceedings of Autonomous Agents and Multi Agent Systems (AAMAS-2004)*, pages 1120–1127, 2004.

[7] S. Sen and N. Sajja. Robustness of reputation-based trust: Boolean case. In *Proceedings of Internatinoal Conference on Autonomous Agents and Multi-Agent System (AAMAS-2002)*, pages 288–293, 2002.

[8] D. Sullivan, B. Grosz, and S. Kraus. Intention reconciliation by collaborative agents. In *Proceedings of the International Conference on Multiagent Systems*, pages 293–300, 2000.

[9] R. Sutton and A. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[10] C. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8:279–292, 1992.

[11] B. Yu and M. Singh. A social mechanism of reputation management in electronic communities. In *Proceedings of International Workshop on Cooperative Information Agents*, pages 154–165, 2000.