To help or not to help[1]

Mahendra Sekaran
Sandip Sen
Department of Mathematical & Computer Sciences
University of Tulsa
Telephone: (918) 631-2985
Fax: (918) 631-3077
Email: mahend@euler.mcs.utulsa.edu, sandip@kolkata.mcs.utulsa.edu

Abstract

Any designer of intelligent agents in a multiagent system is faced with the choice of encoding a strategy of interaction with other agents. If the nature of other agents are known in advance, a suitable strategy may be chosen from the continuum between completely selfish behavior on one extreme and a philanthropic behavior on the other. In an open and dynamic system, however, it is unrealistic to assume that the nature of all other agents, possibly designed and used by users with very different goals and motivations, are known precisely. In the presence of this uncertainty, is it possible to build agents that adapt their behavior to interact appropriately with the particular group of agents in the current scenario? We address this question by borrowing on the simple yet powerful concept of reciprocal behavior. We propose a stochastic decision making scheme which promotes reciprocity among agents. Using a package delivery problem we show that reciprocal behavior can lead to system-wide cooperation, and hence close to optimal global performance can be achieved even though each individual agent chooses actions to benefit itself. More interestingly, we show that agents who do not help others perform worse in the long run when compared with reciprocal agents. Thus it is to the best interest of every individual agent to help other agents.

Introduction

The design of intelligent agents that will interact with other agents in an open, distributed system involve the modeling of other agents and their behavior (Gasser, 1991; Hewitt 1991). Assuming all agents will be cooperative in nature, efficient mechanisms can be developed to take advantage of mutual cooperation, which can produce improved global performance. But, in an open system, assumptions about cooperative agents or system-wide common goals are hard to justify. More often, we will find different agents have different goals and motivations and no real inclination to help another agent achieve its objectives.

[1]With due apology to William Shakespeare.

The above situation may appear to be hopeless. If an agent cannot assume other agents to be cooperative, it might as well solve its problems individually. But this leads to inefficient problem solving performance because agents miss out on mutually beneficial interactions. Even if two individual agents are self-motivated, they should cooperate if such an arrangement is beneficial for both. The question therefore is, when should an agent help another agent? We cannot rely on in-built inclination towards cooperation. The decision to cooperate should be made to serve the agent's own interests. In this paper, we provide a decision-making paradigm that enables autonomous agents to accept or decline requests for cooperation from other agents based on local rather than global considerations.

We assume agent actions to be self-motivated. This means that an agent will help another agent, only if such an action is beneficial to itself in the short or the long run. We use the concept of reciprocity to show that when agents help others who have helped them in the past or can help them in the future, cooperative behavior can evolve out of self-motivation. We propose a stochastic method for deciding whether one agent should help another agent or not in a particular situation. Agents who use this stochastic reciprocity mechanism are called *reciprocative agents* We analyze the effects of *selfish* agents (agents who receive help but do not reciprocate) on the behavior of other reciprocative agents. We also characterize the performance of *individual* agents (agents who never help each other) and *philanthropic* agents (agents who always help others if requested), and demonstrate that a society of reciprocative agents can approximate philanthropic behavior under proper environmental conditions. Our results show that close to optimal system performance can be obtained without sacrificing individual preferences or autonomy.

Coordination of multiple agents

Multiagent systems are a particular type of distributed artificial intelligence (DAI) system (Bond, 1988), in which autonomous intelligent agents inhabit a world with no global control or globally consistent knowledge. In contrast to cooperative problem solvers (Durfee,Lesser, Corkill, 1989), agents in multiagent systems are not pre-disposed to help each other out with the resources and capabilities that they possess. These agents may still need to coordinate their activities with others to achieve their own local goals. They could benefit from receiving information about what others are doing or plan to do, and from sending them information to influence what they do.

Coordination of problem solvers, both selfish and cooperative, is a key issue in the design of an effective DAI system. The search for domain-independent coordination mechanisms has yielded some very different, yet effective, classes of coordination schemes. Whereas some of these work uses architectures and protocols designed off-line (Fox, 1989; Smith,1980) as coordination structures, others acquire coordination knowledge on-line (Durfee, 1988; Sekaran Sen, 1994). In addition, some of these work assumes agents to be cooperative with common system-wide goals (Durfee, 1988; Fox, 1981), and others assume self-motivated agents with individual goals (Genesereth, 1986; Gmytrasiewicz, 1991). The other dimension to consider is if we are analyzing a single instance of agent interaction or if we are considering an ensemble of agent interactions (e.g., in the prisoner's dilemma problem (Rapoport, 1989), most of the formal analysis assume repeated interactions).

In this paper, we assume agents have individual goals or tasks to complete. These individual goals, however, can be achieved more expediently if an agent receives assistance from other agents. This suggests that both individual and overall system performance will improve if agents can intelligently share tasks. We will consider agents who repeatedly interact with each other, and hence past history of problem solving can be used to decide future course of action. The question here is the following: given that there are scopes for cooperation, how should self-motivated agents choose when to cooperate and when not to cooperate with another agent? In the following section we provide an on-line mechanism to answer this question.

Reciprocal decision making

In a companion paper (Sen, 1995) we have shown that reciprocal behavior can be used effectively by agents to balance their workloads. In that paper, each task could be carried out by any agent, and agents could exchange tasks to improve local performance. In this paper we find out if reciprocity is sufficient to promote cooperation when agents cannot transfer tasks, but can use help from others to reduce the cost of performing an assigned task.

We assume a multiagent system with N agents. Each agent is assigned to carry out T tasks. The jth task assigned to the ith agent is t_{ij}, and if agent i carried out this task on its own, the cost incurred is C^1_{ij}. However, if another agent k helped agent i to carry out this task, the cost incurred by each of them is C^2_{ij}. We assume that $2 * C^2_{ij} < C^1_{ij}$, which implies that if two agents together work on the same task, the combined effort required to process the task is less than what it would take one of them to process it. Since the savings, $C^1_{ij} - C^2_{ij}$, obtained by the agent being helped is greater than the cost incurred by the helping agent, C^2_{ij}, there is a net saving of effort for the entire system. This saved effort when combined with reciprocal behavior, can lead to a system with effective individual as well as group performance. So, the gain of an individual is not at the expense of the group.

The obvious question is why should an agent incur any extra cost for another agent? If we consider only one such decision, cooperation makes little sense. If, however, we look at a collection of such decisions, then reciprocal cooperation makes perfect sense. Simple reciprocity means that an agent will help another agent if the latter has helped the former in the past. But simple reciprocity by itself is not sufficient to evolve cooperative behavior. This is because, no one is motivated to take the first cooperative action, and hence nobody ever cooperates! In spite of all the potentials for cooperation and the benefits that it can provide them, agents carry out their own tasks without ever offering to help others.

In real life, in addition to past experience, reciprocity includes a predictive mechanism. An agent can help another agent, if it expects future benefits from the latter. In absence of a general domain-independent predictive mechanism, we propose a much simpler but equally effective stochastic choice mechanism to circumvent the problem of simple reciprocity. We will define S_{ik} and W_{ik} as respectively the savings obtained from and extra cost incurred by agent i from agent k over all of their previous exchanges. Also, let $B_{ik} = S_{ik} - W_{ik}$ be the balance of these exchanges (obviously, $B_{ik} = -B_{ki}$). We now present the probability that agent k will help agent i carry out task t_{ij}. This probability is calculated as:

$$Pr(i,k,j) = \frac{1}{1 + \exp^{\frac{C^2_{ij} - \beta * C^k_{avg} - B_{ki}}{\tau}}},$$

where C^i_{avg} is the average cost of tasks performed by agent i (this can be computed on-line or preset), and β and τ are constants. This gives a sigmoidal probability distribution in which the probability of helping increases as the balance increase and is more for less costly tasks. We include the C_{avg} term because the probability of helping should depend on relative and not absolute cost (if the average cost is

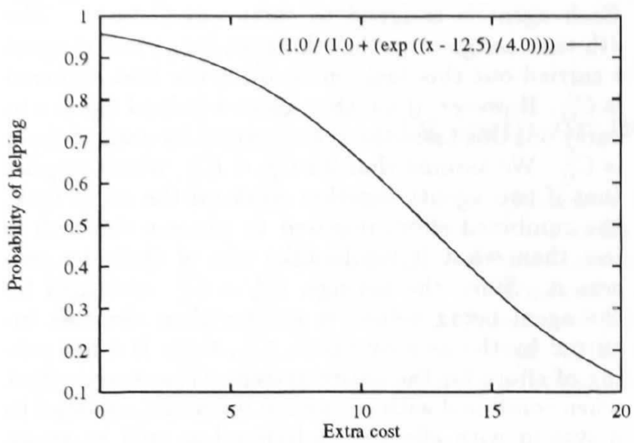


Figure 1: Probability distribution for accepting request for cooperation.

10, incurring an extra cost of 10 is less likely than incurring an extra cost of 1). Due to the stochastic nature of decision-making some initial requests for cooperation will be granted whereas others will be denied. This will break the deadlock that prevented simple reciprocity from providing the desired system behavior.

We present a sample probability distribution in Figure 1. The constant β can be used to move the probability curve left (more inclined to cooperate) or right (less inclined to cooperate). At the onset of the experiments B_{ki} is 0 for all i and k. At this point there is a 0.5 probability that an agent will help another agent by incurring an extra cost of $\beta * C^k_{avg}$. The factor τ can be used to control the steepness of the curve. For a very steep curve approximating a step function, an agent will almost always accept cooperation requests with extra cost less than $\beta * C^k_{avg}$, but will rarely accept cooperation requests with an extra cost greater than that value. Similar analyses of the effects of β and τ can be made for any cooperation decision after agents have experienced a number of exchanges. In essence, β and τ can be used to choose a cooperation level (Goldman, 1994) for the agents at the onset of the experiments. The level of cooperation or the inclination to help another agent, however, dynamically changes with problem solving experience.

A package delivery problem

In this section, we present a simple package delivery problem which we will use to demonstrate the effectiveness of our proposed mechanism to evolve cooperative behavior. Each of N agents is assigned to deliver T packets from a centralized depot to random destinations located at a distance between 1 and D from the depot. An agent can carry only one packet at a time by itself or with the help of another agent. On arriving at the depot, an agent is assigned the next packet it is to deliver. At this point, it checks for other agents currently located in the depot. If so, it asks the other agent for help to deliver this packet. This requests may or may not be honored.

The cost incurred by agents is the time taken to deliver packets. An agent takes 4 time units to cover unit distance if it is carrying a packet by itself. The speed of traveling increases to unit distance per time unit when another agent is helping it. When agents are returning to depot after delivery, they travel unit distance in unit time.

Experimental results

In this section, we present experimental results on the package delivery problem, with agents using the reciprocity mechanism described before to accept or deny a request for help from another agent. We vary the number of agents and packets to be delivered by each agent to examine the effects of different environmental conditions. Values of other parameters used are: $D = 10$, $\tau = 4$, and $\beta = 0.5$. Results are averaged over 10 different randomly generated data sets, where a data set consists of an ordered assignment of package deliveries to agents. All the agents are assigned the same number of deliveries. The evaluation metric is the total cost incurred by the agents to complete all their deliveries.

We used this domain to investigate the effects of agent characteristics on overall systems performance. In our experiments, **philanthropic agents** always agree to help another agent when requested; **selfish agents** request for help but never help others; **individual agents** neither ask for help nor provide help to other agents; **reciprocative agents** use the balance of cost and savings to stochastically decide whether to accept a given request for cooperation. In homogeneous environments (where all agents are of the same type), we expect the group of individual agents and the group of philanthropic agents to provide the two extremes for system performance. The individual agents should incur the highest cost to complete their deliveries (because no one is cooperating), whereas the philanthropic agents should should incur the least cost. We expect reciprocative agent behaviors to lie in between. The frequency of occurrence of cooperation possibilities should determine which of the two ends of the spectrum is occupied by the reciprocative agents. Whether selfish agents can benefit at the expense of reciprocative agents depends on the percentage of selfish agents in the group and the total number of interactions they are likely to encounter. It would seem that reciprocative agents should perform bet-

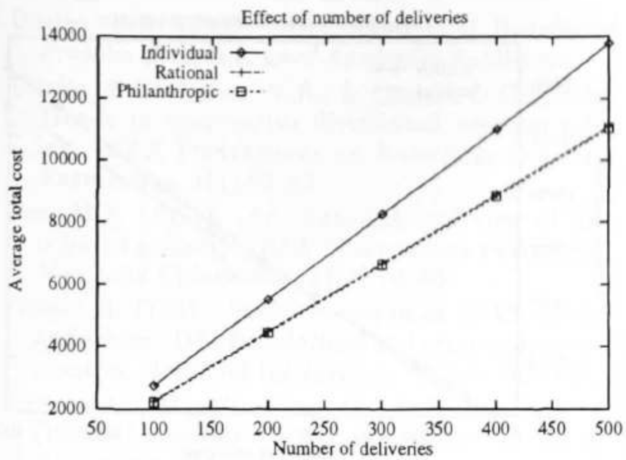


Figure 2: Average total cost incurred by an agent to complete all deliveries.

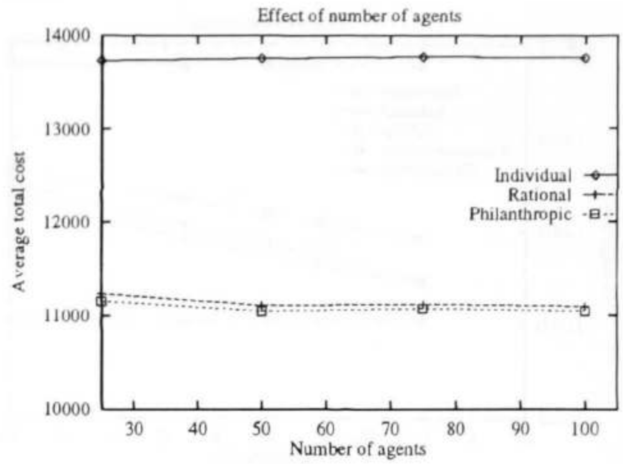


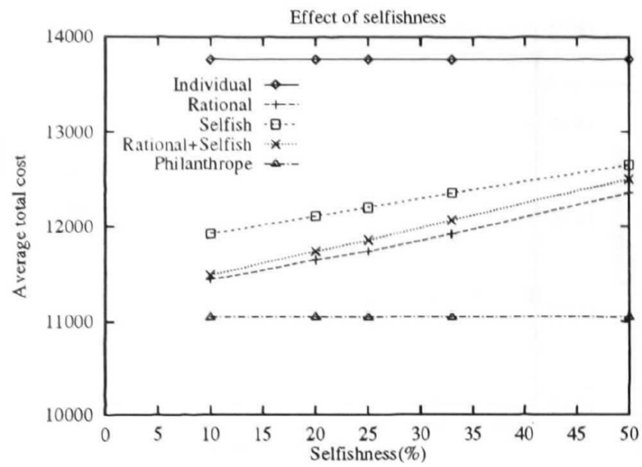
Figure 3: Average total cost incurred by an agent to complete all deliveries.

ter because with sufficient interactions they become philanthropic towards each other, a possibility denied of the selfish agents.

For the first set of experiments, we chose the number of agents, N, as 100 and varied the number of deliveries per agent from 100 to 500 in increments of 100. Experiments were performed on homogeneous groups of individual, reciprocative, and philanthropic agents. Results from these set of experiments are presented in Figure 2. As expected, the performance of the individual agents was the worst, and the philanthropic agents was the best. The interesting thing is that the performance of the reciprocative agent is almost identical to that of philanthropic agents. This is a significant result because it shows that under proper environmental conditions (frequent and prolonged interactions with possibilities of cooperation), self-motivated behavior based on reciprocity can produce mutually cooperative behavior that leads to near-optimal system performance. In addition, with more deliveries, the savings in cost incurred is more with reciprocative and philanthropic agents over individual agents. The ratio of corresponding points on the two curves should be the same, however, as it is determined by the probability of another agent being able to help one agent with its delivery. For the package delivery problem described in the previous section this probability is largely determined by the maximum distance traversed from the depot, D, and the number of agents, N.

We also performed a similar set of experiments by fixing the number of deliveries per agent at 500 and varying the number of agents from 25 to 50 to 75 to 100. Results from these set of experiments are presented in Figure 3. Since the average distance of a package destination from the depot is 5.5, the average cost incurred by an individual agent for delivering a packet is 22 on the way out and 5.5 on the way back, for a total of 27.5. To deliver 500 packets, therefore, the expected cost incurred by an individual agent is 13,750. This fact is verified in the figure. As in the previous experiment, the performance of the individual agents was the worst, and the philanthropic agents were the best. The performance of the reciprocative agents was very close to that of the philanthropic agents, and these improved with more agents. The reason for this improvement was that with more agents, there is more scope for cooperation. However, a level of saturation is reached when all cooperation oppurtunities have been exploited. At this point, an increase in the number of agents do not lead to further improvement in system performance.

The next set of experiments were designed to find out the effects of including selfish agents in a group containing reciprocative agents. We expected that selfish agents should be able to obtain some help from reciprocative agents, and their performance would be better than individual agents but not as good as that of reciprocative agents. For these set of experiments, we chose $N = 100$ and the number of deliveries to be 500. We varied the percentage of selfish agents in the group. Results are presented in Figure 4. The average performance of the group lies in between the performance of the selfish and reciprocative agents, and moves closer to the performance of the selfish agents as the percentage of the latter is increased. The selfish agents are able to exploit the reciprocative agents to improve their performance significantly over individual agents. This is because there are many reciprocative agents and they do not share their balance information with other reciprocative agents. If a reciprocative agent would broadcast the continuous denial of request for help by a selfish



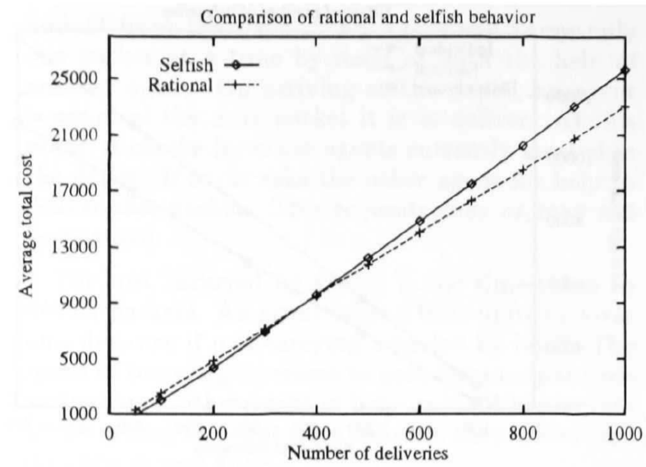


Figure 4: Average total cost incurred by each agent to complete all deliveries as the percentage of selfish agent in a group of reciprocative agents is varied. The individual and the philanthropic agent results do not contain selfish agents and are presented for comparison.

Figure 5: Average total cost incurred by an agent to complete all deliveries with different number of deliveries.

agent who has got a positive balance with the requesting agent, the selfish agent would not be able to exploit other reciprocative agents. Since reciprocative agents incur extra cost for selfish agents without being reciprocated, their performance is noticeably worse than the performance of philanthropic agents. So, the presence of selfish agents can lower the performance of the whole group.

To further analyze the relative performance of selfish and reciprocative agents, we ran a set of experiments varying the number of deliveries while keeping $N = 100$ of which 25 agents were selfish in nature. Results from these experiments are presented in Figure 5. An interesting result was that with few deliveries to make, selfish agents outperformed reciprocative agents. This can be explained by the fact that the number of reciprocative agents were large enough compared to the number of deliveries, which allowed selfish agents to exploit reciprocative agents for most of its deliveries. The performance of the reciprocative agents was affected, as they could not recover from the extra cost incurred to help these selfish agents. With sufficient deliveries to make, however, reciprocative agents emerged as the clear winners. This lends further credence to our claim that it is ultimately beneficial for an agent to be reciprocative rather than selfish in domains where cooperation is always beneficial to the group.

Conclusions

In this paper, we have shown that agents acting in their own self-interest may find it practical to help each other. Under appropriate environmental conditions, such a group of agents can also deliver near-optimal global performance. This is a significant result because in an open, distributed environment, the only defensible strategy an autonomous agent can follow in deciding its actions is that governed by self-interest. Our analysis and experiments show that reciprocal behavior can serve self-interest as well as global efficiency concerns. Since, reciprocating behavior produces better performance in the long run over selfish or exploitative behavior, it is to the best interest of all agents to be reciprocative. It is interesting to note that our proposed mechanism will automatically track behavior changes (e.g., if a reciprocative agent becomes selfish) and adjust agent responses accordingly. This is a powerful scheme that allows for dynamic behavior adjustment to suit the changes in the environment. Our results hold for domains where cooperation always leads to aggregate gains for the group. It would be instructive to study the effects of relaxing this constraint.

The current reciprocation scheme can be enhanced or modified to address other types of agent interactions. If an agent is unable to individually identify other agents, it can use its overall balance of interactions to decide whether or not to accept a request for cooperation. But this also creates new possibilities for exploitative agents. We also plan to investigate more complex domains such as distributed monitoring, distributed information gathering, etc. to further evaluate the strengths and limitations of our proposed mechanism.

References

Bond, A.H. & Gasser, L. (1988). *Readings in Distributed AI*, San Mateo, CA:Morgan Kaufman Publishers.

Durfee, E.H. (1988). *Coordination of Distributed Problem Solvers.*K;uwer Academic Publishers.

Durfee, E.H., Lesser, V.R., & Corkill, D.D. (1989). Trends in cooperative distributed problem solving. *IEEE Transactions on Knowledge and Data Engineering*, 1(1):63–83.

Fox, M.S. (1981). An organizational view of distributed systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(1):70–80.

Gasser, L (1991. Social conceptions of knowledge and action: DAI foundations and open systems semantics. *Artificial Intelligence*, 47(1-3):107–138.

Genesereth, M., Ginsberg, M., & Rosenschein, J. (1986). Cooperation without communications. In *Proceedings of the National Conference on Artificial Intelligence*, pages 51–57, Philadelphia, Pennsylvania, 1986.

Gmytrasiewicz91, P.J., Durfee, E.H., & Wehe, D.K. (1991). A decision-theoretic approach to coordinating multiagent interactions. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 62–68.

Goldman, C. & Rosenschein, J.S. (1994). Emergent coordination through the use of cooperative state-changing rules. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 408–413.

Hewitt, C. (1991). Open information systems semantics for distributed artificial intelligence. *Artificial Intelligence*, 47(1-3):79–106.

Rapoport, A. (1989). Prisoner's dilemma. In J. Eatwell, M. Milgate, and P. Newman, editors, *The New Palgrave: Game Theory*, pages 199–204. Macmillan, London.

Sekaran, M. & Sen, S. (1994). Learning with friends and foes.In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, pages 800–805.

Sen, S. & Sekaran, M. (1995). Using reciprocity to adapt to others. *International Joint conference on Artificial Intelligence workshop on Adaptation and Learning in Multiagent Sytems.*

Smith, R.G. (1980).The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C-29(12):1104–1113.