Self-Reputation Awareness Agents in Spatial IPD Game

Chung-Yuan Huang, Chun-Liang Lee*

Department of Computer Science and Information Engineering, Chang Gung University, Taoyuan 333, Taiwan

*Corresponding Author: cllee@mail.cgu.edu.tw

Abstract

Iterated prisoner’s dilemma (IPD) researchers have shown that strong positive reputations plus an efficient reputation evaluation system encourages both sides to pursue long-term collaboration and to avoid falling into mutual defection cycles. In agent-based environments with reliable reputation rating systems, agents interested in maximizing their private interests must show concern for other agents as well as their own self-reputations—an important capability that standard IPD game agents lack. Here we present a novel learning agent model possessing self-reputation awareness. Agents in our proposed model are capable of evaluating self-behaviors based on a mix of public and private interest considerations, and of testing various solutions aimed at meeting social standards. Simulation results indicate multiple outcomes from the addition of a small percentage of self-reputation awareness agents: faster cooperation, faster movement toward stability in an agent society, a higher level of public interest in the agent society, the resolution of common conflicts between public and private interests, and a lower potential for rational individual behavior to transform into irrational group behavior.

Keywords: self-reputation awareness, reputation evaluation system, intelligent agents, evolutionary computation.

1. Introduction

Reputation provides a foundation for game theorists to analyze ways that past behaviors of social participants affect the behaviors and strategies of iterated prisoner’s dilemma (IPD) game opponents. Using the game as a social interaction model, participants who always choose to cooperate with opponents can be described as having good reputations, but participants who always defect are viewed as having damaged reputations. Participants who establish good reputations tend to receive trust, praise, and other positive feedback from their partners; those with poor reputations do not. We believe that the combination of a positive reputation and accurate reputation evaluation system can encourage two parties to pursue long-term collaboration and to avoid falling into mutual defection cycles, even when faced with short-term sacrifices.

Reputation-related behaviors and strategies have meaning for online commerce. Web 3.0 is supporting a growing number of Internet platforms and commercial applications that use intelligent agent architectures in support of complex online tasks such as auto-bidding on auction websites, placing Internet stock transaction orders, and shopping for cheaper e-commerce products and services (Chen and Yeh 2001; Maes et al. 1999; Wellman 2007). Web 3.0 researchers are therefore experimenting with artificial intelligence (AI) techniques to help intelligent agents “live” in Internet communities in ways that resemble how humans live in real-world communities (Gutowska and Sloane 2010; Hendler 2009; Morris 2011; Tasner 2010). However, because of Internet agent properties such as anonymity, mobility, and multiple identities (Garfinkel and Spafford 2002; Hogg 2006), the use of intelligent agents raises serious game-based and theoretical issues involving cooperation and defection scenarios and conflicts between public and private interests (Axelrod 1997; McGillivray and Smith 2000; Hogg 2006), the use of intelligent agents raises serious game-based and theoretical issues involving cooperation and defection scenarios and conflicts between public and private interests (Axelrod 1997; McGillivray and Smith 2000; Ramchurn et al. 2004). In their current form, intelligent agents do not have to worry about maintaining positive self-images, saving face, or being victims of acts of vengeance associated with fraudulent and defective behaviors commonly found in Web 3.0 e-commerce activities. Since they only care about private interests, they are unlikely to cooperate with other intelligent agents in support of group interests, thus increasing the potential for falling into cycles of never-ending mutual defection (Mintz 2002).

Several agent-based computational simulation
researchers have shown that determining game strategies and behaviors based on an opponent’s reputation is an effective solution that may increase the desire to cooperate (Carter et al. 2002; Fu et al. 2008; Josang 1999; Liang et al. 2009; McKnight and Chervany 2002; Ramchurn et al. 2004; Resnick and Zeckhauser 2002; Wang et al. 2006; Wang and Lin 2008; Wang et al. 2012). When one intelligent agent is required to cooperate with an unfamiliar agent to complete a task, a reputation rating system can have great utility in determining the unfamiliar agent’s trustworthiness (Bromley 1993; Kreps and Wilson 1982). However, a clustering effect resulting in decreased public interest may occur if all agents in a system simultaneously search for other agents with good reputations. This seems inevitable, since agents with good reputations only want to work with other reputable agents. In contrast, agents lacking good reputations must spend a great deal of time performing partner searches because they are in the awkward position of being rejected by ideal partners, in many cases without self-knowledge of their poor or unspecified reputations (Adler 2009; Arnold 1987; Aronson et al. 2010; Axelrod 1997; Leimar and Hammerstein 2001; Maskin 2009; Sachs et al. 2004). Further, if a reputable agent only cares about other agents’ reputations but lacks self-knowledge of its own reputation, it may try to maximize its private interests using behaviors that end up harming other agents, thus causing damage to its existing reputation. Accordingly, any multi-agent system with a reputation-rating scheme must contain a method so that agents interested in maximizing their private interests can exhibit concern for other agents as well as their own reputations.

Our proposed agent model is equipped with a self-reputation awareness component (SRAC) that learns and evolves during spatial IPD games involving two-dimensional social interaction networks. The SRAC agents in our model are capable of evaluating their behaviors based on a mix of public and private interest considerations, and of testing various solutions aimed at meeting and maintaining social standards. Self-reputation awareness helps new agents quickly learn that private interest maximization is best achieved via long-term cooperation with partners, which also serves to enhance their own reputations and to support their wishes for ideal partners in the future. According to our IPD simulation experiment results, as long as an artificial society has a small percentage of agents with this capability for self-reputation awareness, there will be faster cooperation, faster movement toward stability in an agent society, greater public interest in the agent society, resolutions of common conflicts between public and private interests, and decreased potential for rational individual behavior to change into irrational group behavior.

2. Model

Our adaptive agent model contains a self-reputation awareness component (SRAC) based on a mix of social expectation strategies and a reputation evaluation procedure for resolving ongoing conflicts between public and individual private interests in an agent society. It is our belief that an awareness capability that allows agents to reflect on their self-reputations will result in more and faster collaborative behaviors and social benefits. The simulation flow consists of four steps:

1. Reset all parameters and evolutionary computation operators to their default or user-required settings.
2. Use experimental requirements to generate a two-dimensional social interaction $G(V,E)$ network in which total number of nodes $v = |V|$ and total number of edges $e = |E|$.
3. Set generation $g$ to 0.
4. When ($g < Max_G$) do the following loop:
   a. Based on network connection patterns, have nodes at the ends of edges $A_i$ and $A_j$ play a IPD game round during each generation $g$.
   b. Use the payoff matrix shown in Table 1, calculate scores for $A_i(\text{as})$ and $A_j(\text{as})$.
   c. Use these scores as fitness value for agents $A_i(\text{of})$ and $A_j(\text{of})$, with $\text{of}$ $\Rightarrow$ as, and $\text{of}$ $\Rightarrow$ as.
   d. Agents calculate their relative fitness (fitness) and self-reputation (reputation) levels, which represent all opponent evaluations.
   e. Each agent determines whether or not to make strategy adjustments.
      - The current strategy is considered inappropriate when fitness $< 10\%$ or reputation $< 10\%$.
      - Agents needing to adjust their strategies use evolutionary computation crossover operator to combine their original strategies with those used by opponents with high fitness or high reputation levels.
      - Agents use evolutionary computation mutation operator to change strategies according to probability $P_m$.
   f. Add 1 to $g$.

In Step 1, all parameters and evolutionary computation operators must be reset to default or user-required settings. The default settings allow first-time simulation users to quickly execute simple and understandable demonstrations. We categorized individual parameters as IPD game, social interaction network, or evolutionary computation. The first category includes the total number of interactions between an agent and its opponent ($q$), agent memory capacity ($c$), and agent strategy length ($l$). The social interaction network category includes the width ($W$) and height ($H$) of a two-dimensional social network, total number of nodes ($v$) and edges ($e$), neighborhood pattern ($p$), network type ($T$), and edge rewiring probability ($p_r$). The evolutionary computation category includes crossover rate ($P_c$), mutation
rate (Pm), and total number of generations for each experiment (MAX_G). We used the following default values: q = 100, c = 1, l = 4, W = 50, H = 50, v = 2500, e = 10,000, P = Moore neighborhood and periodic boundary condition, Pc = 0.7, Pm = 0.01, and MAX_G = 100. The default configuration of the P neighborhood pattern parameter ensures that all nodes have equal numbers of neighboring nodes, and that each node establishes connections with its eight surrounding nodes to form tightly clustered groups.

Following initialization and parameter setting according to experimental requirements, a specific two-dimensional W x H social interaction network consisting of v nodes and e edges can be established according to the T parameter value. Each social interaction network node represents an IPD agent that is assigned a randomly generated memory-c deterministic strategy. Each edge represents a single IPD interaction relationship between two agents that are labeled as neighbors. Each IPD agent has an average of 2e/v neighbor opponents.

The T parameter can be set as either a cellular automata with high degrees of local clustering and separation, or a small-world network with a high degree of local clustering and low degree of separation. Cellular automata are widely used computational social science investigations of the large-scale outcomes of millions of small-scale events, and for creating visually striking patterns. Small-world networks, which are considered similar to human social networks, serve as the underlying foundations of social simulation models that are said to have high levels of reliability. To compare simulation results for the two network types, we stipulated that the numbers of nodes and edges in each must be equal. To satisfy this condition, if the T parameter is designated as small-world, the simulation is programmed to initially generate a two-dimensional cellular automata according to the P (neighborhood pattern) parameter configuration, and then to use a predetermined edge rewiring probability ρ (default: 1%) to determine whether or not individual edges must be rewired. If rewiring is necessary, either one of the two original nodes (one on each side of an edge) is discarded and replaced with a new, randomly selected node.

In Step 4a, for the sake of simplicity but without loss of generality, we used three IPD agent assumptions: (a) Agent Ai has n opponents, meaning that opponents Oi = (oi,0, oi,1, ..., oi,n-1)g during generation g, with oi,j representing the jth opponent of Ai. (b) Agent Ai plays q IPD game rounds with each opponent during each generation. (c) The fitness value of agent Ai equals the average of all payoffs received by that agent during rounds played within one generation. This value serves as an indication of its performance compared to others in the same agent population.

The IPD game payoff matrix used in Step 4a is shown in Table 1. As indicated, R = 3 represents the reward for mutual cooperation, T = 5 one party’s temptation to defect, S = 0 the “sucker’s payoff,” and P = 1 the punishment for mutual defection. Two conditions for generating a prisoner’s dilemma are T > R > P > S and 2R > T + S. The first guarantees that two rational agents will simultaneously betray each other after understanding that T > R and P > S, and therefore follow the second best choice, which is mutual defection (P, P). According to the second condition, prisoners cannot escape the same predicament by taking turns betraying each other—in other words, benefits for mutual betrayal are not as good as for mutual cooperation. Accordingly, each agent must rely on past behaviors to formulate strategies that optimize long-term benefits.

Table 1: IPD game payoff matrix. Numbers in each cell are payoffs for both players, with player A’s payoff listed first in each pair.

<table>
<thead>
<tr>
<th>Player A</th>
<th>Cooperation (C)</th>
<th>Defection (D)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cooperation (C)</td>
<td>R = 3, R = 3</td>
<td>T = 5, S = 0</td>
</tr>
<tr>
<td>Defection (D)</td>
<td>S = 0, T = 5</td>
<td>P = 1, P = 1</td>
</tr>
</tbody>
</table>

The default strategy in our model is memory-1 deterministic, with agents remembering the behaviors of their opponents in preceding rounds. There are only four possible combinations: both cooperate (expressed as CC), one cooperates and the other defects (CD), one defects and the other cooperates (DC), and both defect (DD). Thus, the memory-1 deterministic strategy can be expressed as the four-value tuple (Scc,Scd,Sdc,Sdd): if an agent’s memory of the preceding round is CC, then it will choose Sc when responding to an opponent. Since responses are limited to either cooperation (C) or defection (D), a memory-1 deterministic strategy consists of 16 (24) possible combinations of moves. Among these, S0 = (C,C,C,C) is known as the “yes-man” (YM) strategy, S5 = (C,D,C,D) the “tit-for-tat” (TFT) strategy, S6 = (C,D,D,C) the “win-stay, lose-shift” (WS/LS) strategy, and S15 = (D,D,D,D) the “scoundrel” (S) strategy. These four strategies have attracted considerable research interest. The WS/LS strategy applies Pavlovian psychological theory in
proposing that an agent will adhere to one strategy until its income goes below a threshold, after which it switches to the opposite strategy (Liu et al. 2012). In the TFT strategy, an agent always chooses cooperation during the first round of a game, and then imitates its opponent’s strategy in subsequent rounds.

In Step 4b, each agent initially uses the evaluation algorithm described in Appendix S2 to give its opponent a relative reputation score at the end of each generation, based on the mean and standard deviation of the number of cooperative moves made by its opponents during one generation. By applying this relative reputation evaluation algorithm, the two algorithms proposed in Appendix S3 can be used to respectively compute an agent’s relative fitness and self-reputation levels in the contexts of its opponents. As shown in Figure 1 and Step 4c of the pseudo-code of our IPD simulation, fitness and self-reputation levels are categorized as high, medium, or low, resulting in nine possible interaction types between a SRAC agent and its opponent. As an example, a SRAC agent with a high degree of fitness and low degree of self-reputation usually adheres to an always-betray or similar “villain” strategy that cannot produce a higher public good value, since it diminishes the ability of other agents to pursue their own interests. Therefore, SRAC agents must be taught that an always-betray strategy will negatively affect their reputations. By referring to and learning from their opponents’ positive performance strategies that conform to social expectations, SRAC agents can achieve higher levels of fitness and self-reputation.

3. Results and Discussion

Our first task was to analyze the results of IPD game simulations using cellular automata and without adding any SRAC agents (Figs. 2, 3 and 4). The first 99 generations can be divided into five stages based on the evolutionary dynamics and spatial distributions of agent-adopted strategies. During the first stage (generations 0-3), our proposed model starts with a pool of randomly generated strategies adopted by individual agents being evenly distributed throughout the cellular automata (Fig. 4a). During the second stage (4-10), agents tend to give in to the temptations of maximizing their private interests and use the S strategy. As stated earlier, when a majority of agents adopt that strategy, the entire community eventually enters a cycle in which overall and individual private benefits rapidly decrease (Figs. 2 and 3b). In cellular automata, if the majority of an agent’s adjacent neighbors adopt the S strategy, then the agent in the center is forced to adopt the same strategy in order to survive (Figs. 4b and 4c).

During the third stage (11-20), agents wanting to move away from the S strategy are likely to move toward a TFT strategy (Fig. 3a). In addition to confronting S strategy agents, this action also supports cooperation with agents who adopt either the YM or TFT strategies. The spatial clustering effect is also known as network reciprocity (Nowak 2006; Nowak and May 1993). Figures 4c and 4d illustrate a scenario in which TFT strategy agents gradually increase in number and cluster in a manner that surrounds and restricts agents who adopt the S strategy.

The number of TFT strategy agents declines during the fourth stage (21-40). Due to an asymmetry problem involving memories of previous encounters, these agents start to defect and stop trusting each other, resulting in less clustering over large areas. However, as shown in Figures 4d and 4e, some TFT strategy agents continue to surround S strategy agents to ensure that the latter do not expand to the point of overwhelming the former. Note also that as clusters of TFT strategy agents start to break up and decrease in size, the number of agents that adopt the WS/LS strategy increases (Fig. 3a) (Nowak 2006). Since WS/LS strategy agents do not have asymmetric memory problems regarding previous encounters (which increases the potential for breaking promises), and since those same agents generally

\[1\] Network reciprocity is receiving attention from physics researchers such as Perc and Szolnoki (2010), Santos and Pacheco (2005), and Wang et al. (2012a, 2012b, 2012c, 2013a, 2013b, 2014).
move toward mutual collaboration, their numbers and tendency to cooperate gradually increase.

During the fifth stage (41-100), strategy evolution enters a state of “dynamic stability”—a term we use to describe a long period of repetition. Within clusters of WS/LS strategy agents, the number of YM strategy agents gradually increases (Figs. 4e and 4f). Agents who adopt either the YM or WS/LS strategies interact in ways that benefit both sides. However, in reaction to this increase, some agents take advantage of the situation by reverting to the S strategy, which reduces (and in some cases eliminates) clusters of agents that adopt the all-cooperation strategy. This scenario, which is often found in human societies, increases the potential for damage from internal mutation and external invasions.

Average payoff curves from our IPD simulations using 0% (baseline), 10%, 30%, 50% and 100% SRAC agents are shown in Figures 5 (cellular automata) and 6 (small-world network). Initial parameter settings were identical. As indicated by the red average payoff curves in the two figures, the overall network community clearly benefited when all agents possessed the capacity for self-reputation awareness, with a state of dynamic stability achieved within very few generations. However, since such a situation is not possible in the real world, we focused on the effects of adding a small number of SRAC agents to an otherwise unaltered environment. According to the blue (10%) and green (30%) average payoff curves, adding a small number of SRAC agents exerted a significant influence, regardless of social interaction network type. Specifically, they suppressed growth in the number of agents who adopted the S strategy, prevented the initiation of a cycle in which all agents expressed betrayal and retaliatory behaviors, and helped resolve conflicts between society-wide benefits and individual private interests so that cooperation gained acceptance as mainstream behavior.

The average payoff curves in Figures 5 and 6 are similar because small-world networks contain many random long-distance shortcuts that reduce network separation. There are at least two reasons for a lack of strategic clustering: these shortcuts produce very low degrees of separation, and they significantly increase complexity in terms of agent interactions and indirect influences. Note that the influence of a single game strategy can result in increased evolutionary diffusion and the increased containment of other agents. Combined, these factors accelerate the movement toward dynamic stability.

4. Conclusion

In this paper we described our proposal for a self-reputation awareness model in which agents are given the ability to calculate and interpret their self-reputation levels, and to adjust their IPD game strategies accordingly. Our primary conclusions are (a) the model successfully encourages strategy adjustments to achieve an optimum balance between self-reputation and private interests, thus
increasing the likelihood that an agent will suppress its betrayal behavior and defection strategy in order to increase cooperation with other agents; and (b) compared to other models, overall cooperative behavior in our proposed model is likely to emerge much faster.

Figure 4. Spatial distributions of 16 memory-1 deterministic IPD strategies in cellular automata without adding SRAC agents.

Acknowledgment

The work was supported in part by a Grant from the Republic of China National Science Council (MOST-103-2221-E-182-052). The work was supported in part by the High Speed Intelligent Communication (HSIC) Research Center, Chang Gung University, Taiwan.

References

(18) Josang A (1999) Trust-based decision making for


