Effects of Agent Representation on the Behavior of a Non-Reciprocal Cooperation Game

Nicole P. Leahy

Graduate Program in Bioinformatics and Computational Biology
Iowa State University
Ames, IA 50011

Abstract

Game theoretic models are often used to simulate phenomena observed in the natural world. It is generally assumed that the implementation (or representation) of the agents within a game has no significant effect on the outcome of simulations. To test this assumption the effect of changing the representation of agents in the non-reciprocal cooperation game studied by Riolo et al. (2001) was used. In addition to the implementation used by Riolo et al. (2001), agents were also represented as higher dimension real vectors, integer values, at bit-strings. It was found that the method of agent implementation used has a highly significant effect on the cooperation rate and general behavior of the simulation.

1 INTRODUCTION

Biologists and social scientists have long been interested in the evolution of cooperation (e.g. Roy, 2000; Sigmund and Nowak, 1999; Mouston et al., 2000; Hemesath, 1994). It is not understood how selfishly acting individuals could evolve cooperative behavior (Axelrod, 1984). Scientists studying this phenomenon often use theoretical games to investigate the origin of cooperation. It is often tacitly assumed that agent representation has only minor effects on the long-term behavior and results of the model. The purpose of this paper is to present the results of a study in which the method of representing the agent in the model published by Riolo, et al. (2001) was altered.

In Riolo et al. (2001)'s game individuals (agents) learned to cooperate despite the cost of assisting another agent and without reciprocity. In brief, this game was designed so that an agent assisted another agent if identification tags were sufficiently similar. The agents had no memory of previous encounters and were unlikely to play other agents more than once (a substantial departure from the usual iterated game used to study the evolution of cooperation). Riolo et al. (2001) reported agents to assist other agents at a rate of 73.6%, after evolution. A more detailed description of the Riolo et al. (2001) simulation is provided in the Experimental Design section.

Agents in Riolo et al. (2001) consisted of two real numbers: the first served as an identity tag and the other determining the altruistic behavior. This study used three alternate implementations of the agents: multi-dimensional real vectors, integer value, and bit-strings. Using a multi-dimensional vector representation expands the tag space (the number of possible tags). The real number representation of computers is a fine-grained approach since tags may vary by very small amounts. When using integer value tags with a maximum value, the difference between agents becomes more pronounced. As the size of the maximum allowable integer increases, the results should approach those of the single dimensional real vector case. Using bit-strings examines the effect of both higher dimensionality and the granularity of the environment.

2 EXPERIMENTAL DESIGN

The experiments in this paper were conducted in the same manner as described in Riolo et al. (2001). Specifically, each simulation used a population of 100 agents which were evolved for 30,000 generations. Each agent had a tag (τ), tolerance threshold (T), and a score. Every generation, each agent played three agents selected at random with replacement. For each place the first agent (agent A) determines if it will assist the other agent (agent B). If the distance between tags of agent A and agent B was lower than the threshold of agent A, the score of agent A was lowered by 0.1, and the score of agent B increased by 1.0. This represents a costly contribution by agent A to agent B.

The next generation was determined with tournament selection of size two, where the population was randomly shuffled and two agents were select without replacement until there were no remaining agents. if the agents in each pair had equal scores, each was copied to the next generation. If one agent had a higher score, that agent was copied twice to the next generation and the agent with the lower score was not copied. Each copy was then subjected to mutation. Mutation of the tag occured at a frequency of 0.1, and was performed by generating a new tag uniformly at random in the tag space. At the same frequency, the tolerance threshold mutated by a representation-dependent function with a distribution designed to leave the simulation as close as possible to the original study (Gaussian with mean zero and a standard deviation of 0.01). If the new $T<0$, then it was set to 0.

2.1 MULTI-DIMENSION REPRESENTATION

An agent tag in Riolo et al. (2001) was represented as a real number, $\tau \in [0,1]$, with tag distance calculated as the absolute value of the difference between two tags. Similarly, the agent tolerance was initialized as a real number, $T \in [0,1]$. To examine the effect of a larger tag space on the behavior of the game, tags were represented as real vectors, $\tau \in [0,1]_n$, where n is the dimension of the tag space. The tolerance was represented as a single real variable with $T \in [0,1]$. The distance between the tags of two agents, A and B, is the Euclidean distance,

$$D_{\tau_A \tau_B} = \sqrt{\sum_{i=1}^{n} \left(\tau_A(i) - \tau_B(i)\right)^2}$$

Player A assists player B if $D_{\tau_A \tau_B} \leq T_A$.

The mutation of T was the same as in Riolo et al. (2001) except that the standard deviation (σ) of the mutation distribution was adjusted so that the ratio of the hypervolume of the hypersphere of radius σ (V_σ) to the hypervolume of the tag space (V_t) was constant across dimensions (Table 1). This was done in order to insure that the tolerance mutation was equivalent to theat used in Riolo et al. (2001). Runs were performed for n = 1, 2, 3, 4, 5, where n is the number of dimensions of the tag space. 30 replicates were run for each tag space. Note that the case where n=1 is identical to the setup for Riolo et al. (2001).

Table 1: σ for Higher Dimensions

n	***σ***	***V_σ:V_t***
1	0.01	0.02
2	0.079788456	0.02
3	0.16838903	0.02
4	0.25231352	0.02
5	0.32805473	0.02

2.2 INTEGER REPRESENTATION

Using real values for τ and T is a relatively fine-grained environment since it is extremely unlikely that any two agents will have the same τ. In Riolo et al. (2001), T most frequently changed by small values relative to the tag space, creating a fine distinction between agents with an approximately continuous tag space. By employing integer representation of the tag space and tollerance, the simulation becomes more discrete. Both τ and T were initialized as τ, $T \in [0,1,2,...,I]$, where I is the maximum integer. Constraining the tag space and randomly selecting new tags may result in duplication of the tags of other agents, especially at low values for I. The mutation of T was performed by adding $\Delta = \sum_{i=1}^{I} \delta_i$, for $\delta_i \in [-1,0,+1]$, $\delta_i \in$ [-1, 0, +1], with $p(-1) = p(+1) = 0.1$, and $p(0) = 0.8$. Experiments were run for I=10, 100, 1000, and 10,000.

2.3 BIT-STRING REPRESENTATION

Implementing the tag and tolerance as bit strings both increases the tag space by using larger strings and constrains the tag space, reducing the values of each dimension to one of two states. The distance between two agents was measured as the Hamming distance. The value of T was implemented as the weight of the tolerance binary string. Mutation of the tolerance was performed as bit flips determined by a Poisson distribution with the expected probability of each bit-flip set equal to 0.01. Since it is not possible for $T < 0$, there were no boundary effects as there were in the other two representations. Experiments were performed for string lengths, L of 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100 bits. The bit-string for the tag and tolerance were of equal length in all simulations.

3 RESULTS

An interesting difference between the results of these experiments and those reported by Riolo et al. (2001) was the presence of failed states. A *failed state* is an occurrence of donation rates less that 10% for at least one generation. The value of 10% was selected from examination of the reproduction of the results of Riolo, et al. (2001). In only one replicate did a population fall to a donation rate of 10% or lower one without persisting in a failed state for multiple generations. The comparisons of a typical run from Riolo et al. (2001) compared to a failed state are shown in Figures 1 and 2. The *donation rate* is the proportion of plays per generation that result in one agent assisting another. A *cooperative state* is defined as an occurrence of donation rates greater than 10%. Significance levels were calculated using ANOVA and student-t tests.

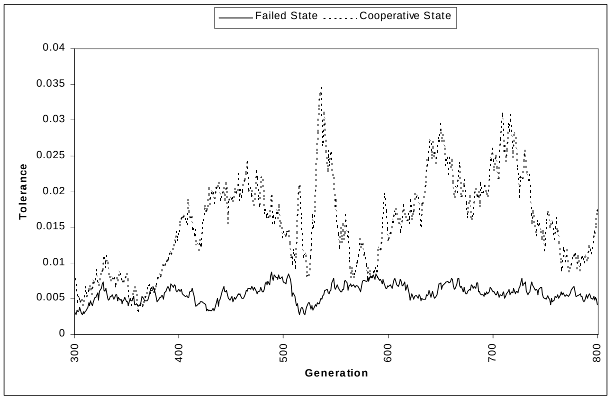

Figure 1: Tolerance Examples For Cooperative States (n=1, Replicate 6) And Failed States (n=1, Replicate 0)

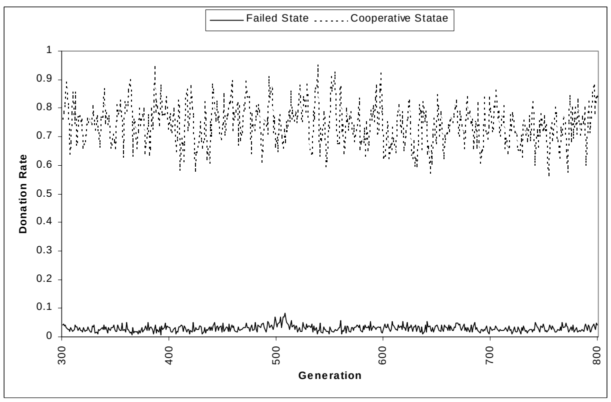

Figure 2: Example Donation Rate For Cooperative States (n=1, Replicate 6) And Failed States (n=1, Replicate 0)

3.1 MULTI-DIMENSION REPRESENTATION

Results of the one-dimensional case, a replicate of Riolo et al. (2001) were consistent with those they reported. In this study the mean donation rate was 73.4 ± 0.3%, whereas Riolo et al. (2001) foud 73.6%. The average toleranceswas identical at 0.019.

As the number of dimensions increased the average number of failed states increased significantly (Table 2) with $p < 0.0001$. The dimensions in which significantly different proportions of occured failed states are given in Table 3.

Table 2: Mean Values for Experiments

Experiment	Failed States	Tolerance	Donation Rate
n=1	18.6	0.0185	0.7344
n=2	1853.2	0.1071	0.6837
n=3	3898.3	0.1816	0.6340
n=4	4949.9	0.2394	0.6086
n=5	5630.2	0.2937	0.5913
I=10	0	0.0951	0.7217
I=100	127.7	0.0481	0.7096
I=1000	164.1	0.0229	0.7263
I=10000	23.2	0.0115	0.7420
L=10	0	0.0027	0.9898
L=20	0	0.0068	0.9807
L=30	0	0.0180	0.9730
L=40	0	0.0756	0.9702
L=50	0	0.2557	0.9817
L=60	0	0.4185	0.9953
L=70	0	0.4602	0.9983
L=80	0	0.4675	0.9987
L=90	0	0.4760	0.9988
L=100	0	0.4785	0.9989

The mean donation rate was found to decrease with increasing dimension ($p < 0.0001$) with the same significant groupings as for the average number of failed states (Table 3). The mean tolerances of the dimensions were significantly different ($p < 0.0001$).

Table 3: Significantly (*) and Insignificantly (·) Different Occurrences of Failed States Between Multi-Dimension Representations

n	**1**	**2**	**3**	**4**	**5**
1	-	*	*	*	*
2		-	*	*	*
3			-	.	*
4				-	.
5					-

3.2 INTEGER REPRESENTATION

The results of the integer representations were compared with the results of the one dimensional real vector representation, *R* (the replication of Riolo et al. (2001)). ANOVA results indicated that the average number of failed states was significantly different from *R* ($p < 0.0230$). Only the mean number of failed states for *I*=1000 was significantly higher than that for *R*. The significant comparisons are shown in Table 4.

Table 4: Significantly (*) and Insignificantly (·) Different Occurrences of Failed States for Integer Representations

	***I*=10**	**100**	**1000**	**10000**	***R***
***I*=10**	-	*	*	·	·
100		-	·	·	·
1000			-	*	*
10000				-	·
R					-

The mean tolerance of the integer representations decreased with increasing *I* ($p < 0.0001$), with each mean tolerance significantly lower than the previous (Table 2). At *I*=10,000, the tolerance was less than *R*. The mean donation rate decreased from *I*=10 and *I*=100 and then increased with *I*=10,000 having a greater donation rate than *R* ($p < 0.0001$).

3.3 BIT-STRING REPRESENTATION

There were no failed states in any of the bit-string experiments, so there was no significant difference across simulations with different values of *L*. The mean tolerance increased with increasing *L* ($p < 0.0001$) beginning at *L*=30 until *L*=70, with no significant difference between *R* and *L*=10, 20, and 30. The mean donation rate gradually decreased from *L*=10 to *L*=50 and then increased for *L*=50 to *L*=70 ($p < 0.0001$). The mean donation rate for bit-strings was never below 97.0%, significantly greater than *R* ($p < 0.0001$). See Table 2 for the data.

4 DISCUSSION

4.1 MULTI-DIMENSION REPRESENTATION

The results demonstrated that the frequency of failed states increased as the tag space increased in dimension. In the one-dimensional space when a tag becomes the most prevalent tag the selection for low tolerance is weakened and the population will move towards higher tolerance. Eventually, an agent with a different tag will fall within the tolerance of the dominant tag. This can occur through increasing the average tolerance of the agents with the dominant tag or the generation of agents with a new tag within the tolerance threshhold of the dominant tag type. Over time, agents with the rarer and less tolerant tag will become dominant through exploiting the previously dominant agents, as in Riolo et al. (2001).

In the one-dimensional case an exploitative agent will have a tag either less than or greater than the most common tag. When the number of dimensions was increased there were more axes along which tags may vary, allowing multiple agents with differing tags to exploit the dominant tag. Agents with the dominant tag will quickly be removed from the population with no new dominant type to take its place. This failed state will persist while numerous agents with low tolerances fail to assist one another. Eventually, another dominant type will emerge. From the duration of failed states, it is thought that the average amount of time until a new type takes over increases as the dimension of the tag space increases. It is conjectured that the increasing dimension makes it more difficult for a new tag to be within the tolerance threshold of an extant tag.

The decrease in donation rate observed as the number of dimensions in the tag space increased was due to a reduced donation rate in the failed states compared to the cooperating states. Adjusting for failed states by dividing the mean donation rate by the proportion of cooperative generations over all generations results in an increase in the average donation rate (Table 5). However, the one-dimensional case is still significantly higher than those of higher dimensions ($p < 0.0001$) with the higher dimensions having mean donations rates of 72.9% or 72.8%. This seems to indicate that the occurrence of failed states accounts for most of the decrease in donation rate.

Table 5: Adjusted Tolerance and Donation Rates for the Multi-Dimensional Representations

n	***Tolerance***	***Donation Rate***
1	0.0185	0.7349
2	0.1144	0.7289
3	0.2100	0.7290
4	0.2887	0.7284
5	0.36583	0.7288

The increase in mean tolerance is more puzzling. Since the standard deviation of the tolerance mutation function increased with increasing dimension, it was thought that there may be a consistent ratio between the two. Calculating the ration of the standard deviation of the mutation function to the mean tolerance demonstrate that this was not the case (Table 6). The ratios were calculated with both the mean tolderances from Table 2 and the adjusted tolerance from Table 5 While the mutation function likely has some effect, the selection of the game acts to decreasee the tolerance, weakening the correlation between the mutation function and the mean tolerance.

Table 6: Ratio of the Standard Deviation of the Tolderance Mutation Function to the Mean Tolerance

n	$E(T_n)$**:**σ_n ***unadjusted***	$E(T_n)$**:**σ_n ***adjusted***
1	0.0185	0.7349
2	0.1144	0.7289
3	0.2100	0.7290
4	0.2887	0.7284
5	0.36583	0.7288

4.2 INTEGER REPRESENTATION

The results of the integer representations were expected to approach those of the one-dimensional real vector simulations (*R*) with increasing *I*. It was found that the mean number of failed states and donation rate initially diverged from *R* and then approached that of *R* (Talbe 2). Examination of the data indicated that the failed states occurred early in the run and persisted for several generations before evolving to a cooperative state (Figure 3). For *I*=10,000, the initial failed states resemble the dip in donations rates observed by Riolo, et al. (2001). Given the decreased cooperation and increased failed states for *I* = 100 and 1000, it is unlikely that the cooperative behavior seen at *I*=10 and *I*=10,000 were due to the same dynamics. For *I*=10, a mutated tag may take on one of only ten values, so it was likely that the new tag would be the same as existing agents. Therefore, if the tag of an agent with low tolerance was mutated it would cooperate with agents already present. This coarse granularity makes it unlikely for an agent to be in a position to exploit others. For *I*=10,000, the dynamics are similar to those for *R*.

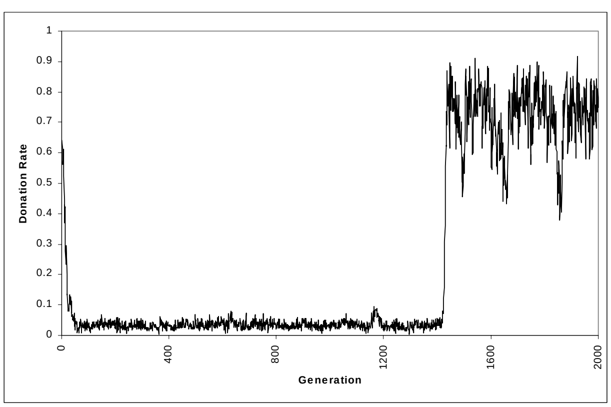

Figure 3: Example of the Initial Behavior for Integer Representations (I = 1000, Replicate 0)

4.3 BIT-STRING REPRESENTATION

Of the three types of representations, bit-strings produced the greatest average donation rate. All tested string lengths had mean donation rates exceeding 97%. In the case of runs with many agents of near zero tolerance, this must have been the result of identical or nearly identical agents. When the average tolerance is large ($T \geq 0.47$), it is not necessary for agents to be highly similar. The process of generating a new tag results in a tag that by chance will have 50% similarity to all possible tags. With an average $T \geq 0.47$, it is likely a tag will be in the tolerance threshold of the majority of the agents. Furthermore, since the agent with the new tag will have a *T* close to 0.5, most of the agents will be within its own threshold.

The pattern of populations of short strings evolving low tolerance while long strings evolve high tolerance remains to be explained. For intermediate values of *L* (lengths of 40, 50, and 60), the populations begin with a tolerance level similar to those of the longest strings, but changed to low tolerance levels (Figure 4). This suggests that a cooperative state may not be stable if the simulation is run for a large number of generations. The stability of high mean tolerance may be due to the fact that a single bit-flip in the tolerance string would have a greater impact on the population dynamics for low values of *L* than for high values.

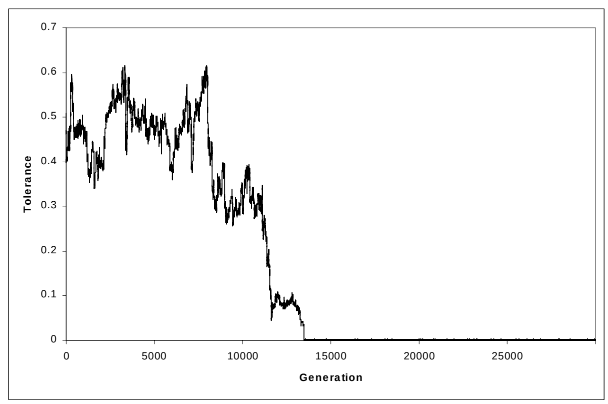

Figure 4: Behavior of Bit-Strings for Intermediate Values of *L*=50 (Replicate 0)

At the initialization of a population there will tend to be more exploitative ($T < 0.4$) and exploitable ($T > 0.6$) agents for shorter strings than longer strings (Table 7). The greater effect of a single bit flip at shorter lengths often results in a population in which agents with low tolerances take advantage of agents with high tolerances. In time, only the agents with low T values remain. When a tag is mutated, there is a greater likelihood of agents being more than 50% dissimilar at shorter lengths. As length increases, the number of exploitative and exploitable agents decreases, leading to the observed trend of more generations passing before *T* becomes small. Eventually, long strings (a length of 40 or greater) maintain high values for *T* for the duration of the run, though this may not be the case if run for more generations than was done in this study.

Table 7. Expected Proportion of Exploitable and Exploitative Agents at Initialization of Bit-String Populations

L	*Proportion*
10	0.363
20	0.263
30	0.200
40	0.154
50	0.119
60	0.092
70	0.072
80	0.057
90	0.045
100	0.035

5 CONCLUSIONS

The results of the experiments described in this paper demonstrate that altering the representation of the agents in the Riolo et al. (2001) simulation has significant effects on the outcome. In the literature, when game theoretic models of the evolution of cooperation are examined, only the rules of the game rather than the digital representation of the agents are modified. The results described above caution that generic statements of a game should not be made from a single implementation. Furthermore, since the outcome of this game is partially dependent upon the representation used, it would not be prudent to present data as a model for the natural world unless it is either confirmed by multiple representations or shown that the chosen representation is a reasonable approximation, such as fitting the model to data from the natural world. With the variety of models used to simulate aspects of the nature, it would be of interest to test the effect of agent representation on the behavior of other game theoretic models.

5 ACKNOWLEDGEMENTS

I would like to thank Dan Ashlock (Department of Mathematics, Iowa State University) for introducing me to game theoretic models of cooperation and his advice on this project.

6 REFERENCES

Axelrod, R. (1984). *The Evolution of Cooperation*. Basic Books, New York.

Hemesath, M. (1994). Cooperate or defect? Russian and American students in a prisoner's dilemma. *Comparative Economic Studies* **176**:83-93.

Mouston, J.M., J. Kinnie, B. Luop, C. Terry, and S.S. Ho (2000). Competitiveness and conflict behavior in simulation of social dilemma. *Psychological Reports* **86**:1219-1225.

Nowak, M.A., and K. Sigmund (1998). Evolution of indirect reciprocity by image scoring. *Nature* **393**:573-577.

Riolo, Rick L., Michael D. Cohen, and Robert Axelrod (2001). Evolution of cooperation without reciprocity. *Nature* **414**:441-443.

Roy, D. (2000). Learning and the theory of games. *Journal of Theoretical Biology* **204**:409-414.

Sigmund, K., and M.A. Nowak (1999). Evolutionary game theory. *Current Biology* **9**:R503-R505.