Evolving agent societies that avoid social dilemmas

Manisha Mundhe & Sandip Sen
Department of Mathematical & Computer Sciences,
University of Tulsa
e-mail:manisha@euler.mcs.utulsa.edu, sandip@kolkata.mcs.utulsa.edu

Abstract

The social sciences literature abound in problems of providing and maintaining a public good in a society composed of self-interested individuals [8]. Public goods are social benefits that can be accessed by individuals irrespective of their personal contributions. In our previous work we have demonstrated the use of genetic algorithms (GAs) for generating an optimized agent society that can circumvent a particularly problematic social dilemma. In that approach, each chromosome represented the entire agent society and the GA found the best co-adapted society. Though encouraging, this result is less exciting than the possibility of evolving a set of co-adapted chromosomes where each chromosome represent an agent, and hence the population represents the society. In this paper, we describe our approach to using such an adaptive systems approach to using GAs for evolving agent societies. We present experimental results from several domains including the classic problem of the Tragedy of the Commons [17].

1 Introduction

We are interested in using GAs for evolving agent behaviors. In particular, we have been experimenting with evolving agent societies or agent groups to solve particular problems. For a number of years, GAs have been successfully used primarily as function optimizers. Holland's work on GAs, however, was motivated by design and implementation of robust adaptive systems [13]. In recent years researchers have started to pay more attention to the initial motivation of GAs [3].

In a typical function optimization based approach to evolving agent societies, a single chromosome represents the entire society, and the optimization problem translates to finding the optimal society [1, 10]. The evaluation of one chromosome is independent of the other chromosomes in the population. While such an approach can be effective, it is inherently less interesting a model for evolving agent societies. Contrast this to the evolutionary approach where a single population is evolved where each chromosome or structure represents an agent, and the challenge is to evolve an effectively co-adapted population of chromosomes. The evaluation of each structure depends on all the other structures in the population. We will call this an *adaptive systems* approach to evolving agent societies[1].

There are at least two compelling reasons for investigating the adaptive systems framework:

1. It allows for the construction and evaluation of novel challenging scenarios for evolving populations, e.g., situations where one agent has to compete with some and cooperate with other agents.

2. Careful experimentations with these scenarios can help us develop a better understanding of the dynamics of evolutionary algorithms. The optimization framework of independent chromosome evaluation have been studied extensively and we have developed a reasonable understanding of how an evolutionary algorithm work in this mode. But when the evaluations of population structures are interdependent, as in the adaptive systems framework, much less is known and there is little understanding of the dynamics and convergence of

[1] Though this can also be called *co-evolution*, the latter term is often used to denote situations where two or more GA populations are co-evolved in a cooperating or competing scenario where there are as many GA populations as agents in the agent society being evolved [2, 19].

the evolutionary process. In the adaptive systems framework, a more complex interplay between agent strategy representation, evaluation modes (how many pairings, with whom, etc.), evaluation functions, selection and replacement schemes, incremental versus generational evolution schemes, etc. provides a richer framework to research the capabilities and limitations of evolutionary paradigms.

Interesting computational problems in agent societies include paradoxes that involve reduction of system throughput when more resources are added to an existing system. Social dilemmas appear frequently in natural and artificial societies. The viability of an individual in such societies depends on all the other members. In our previous work [1], we presented our results of applying GAs as function optimizers to solve social paradox problems. We used Pitt style genetic based machine learning system, where each structure in the population represents an entire group of agents. A critique of this work would be that since the GA selects between alternative societies it really avoids the social dilemma problem: rather than solving the social dilemma problem for a society, the GA selects the society which has solved this problem. This argument is analogous to the Michigan versus Pitt-style classifier system debate, where the latter avoids the individual rule credit assignment problem in the former by working with entire rule sets.

Social dilemma arises because individual rewards promote exploitative behavior which ultimately leads to doom for the entire society. Societies that avoid such tragedy would consist of sufficient number of non-greedy individuals that can foresee the consequences of their actions. If we were to allow, in contrast to our previous work, evaluation of individual agents rather than societies, can social dilemma be avoided? In general, the answer is no.

This paper investigates some individual evaluation schemes that is able to surmount the odds and avoid harmful social dilemmas. Although the adaptive system approach seems to be natural for social dilemma problems, difficulties in finding a good representation and appropriate GA framework and parameters have been recognized by multiagent systems researchers [4]. To the best of our knowledge no other successful attempt has been made to use the adaptive systems approach to solving social dilemma problems.

In the rest of the paper, we introduce the social dilemma problems, compare the function optimization and adaptive systems approaches, present results from our experiments, and identify future plan of work.

2 Evolving agent societies

In recent years, considerable interest and enthusiasm has been generated by the prospect of widespread use of intelligent agent based systems [15]. The design of successful multiagent systems is, however, a problem of significant magnitude. Often multiple, conflicting criteria have to be simultaneously optimized to come up with a cost-effective multiagent system design. Genetic algorithms provide us with another tool for designing both individual agent characteristics as well as social rules for multiagent systems.

Some GA researchers have used GAs to evolve or co-evolve agent strategies for multiagent environments [6]. This body of research includes work in both competitive and cooperative co-evolution. Co-evolution involves concurrently running several GAs to evolve populations such that the fitnesses of individuals in one population is dependent on individuals in other populations. Frequently a structure in each population corresponds to the design of an agent in some environment. In cooperative co-evolution, each such structure (or agent) in one GA population is evaluated by combining with agents from other GA populations and observing the performance of these set of agents on some common task in the environment [7, 18]. In competitive co-evolution, on the other hand, agents from different GA populations compete for fitness in a shared environment [7, 21]. This may appear counter-productive, but the assumption is that the rise in fitness of the individuals in one population will pose stiffer challenges for individuals in the other population. This form of selection pressure will lead to an improvement in the population fitness of the latter population, which in turn will lead to an improvement in fitness in the first population, and so on.

Other approaches to developing agent strategies include the genetic programming paradigm [10]. This body of work is also different from the above cited literature in that here a single GP population is used to evolve agent strategies for multiple agents. In [10], a GP approach is used to evolve an agent behavioral strategy, which is then executed by multiple agents in an environment, i.e., the same program is followed by all agents. In [9], however, a single GP is used to evolve several agent programs, i.e., one structure in the GP population consisted of multiple agent strategies.

Another novel attempt of using GAs to address multiagent system design problems also used a single GA population [4]. In this work, the researchers attempted to use a GA to evolve a society of co-adapted agents. Each member in the population corresponded to a unique agent. They applied the GA to a social dilemma problem, where local greedy choices lead to a worse payoff for each individual. Other paradoxes involve reduction of system throughput when more resources are added to an existing system. The authors claim that social dilemmas appear frequently in natural and artificial societies. As such, mechanisms that are able to circumvent these problems will contribute significantly to multiagent system designs. Unfortunately, the researchers come to the conclusion that their GA implementation is not effective in addressing the social dilemma problems that they investigate. In this paper, we evaluate the results reported in the above-mentioned paper [4] and present an alternate, albeit more successful GA based approach to solving such dilemmas.

3 Social dilemmas

A social dilemma arises when agents have to decide between contributing or not contributing towards a public good without the enforcement mechanism of a central authority [5]. Individual agents have to tradeoff local and global interests while choosing their actions. If a sufficient number of agents make the selfish choice, the public good may not survive, and then everybody suffers. In general, social laws, taxes, etc. are enforced to guarantee the preservation of necessary public goods. In the following, we present a couple of well-studied social dilemmas that we address with GAs in this paper:

Tragedy of the Commons: In his book, *The Wealth of Nations* (1776), Adam Smith conjectured that an individual for his own gain is prompted by an "invisible hand" to benefit the group [22]. As a rebuttal to this theory, William Forster Lloyd presented the *tragedy of the commons* scenario in 1833 [17]. Lloyd's scenario consisted of a pasture shared by a number of herdsmen for grazing cattles. Lloyd showed that when the utilization of the pasture gets close to its capacity, overgrazing is guaranteed to doom the pasture land. For each herdsman, the incentive is to add more cattles to his herd as he receives the full proceeds from the sale of additional cattle, but shares the cost of overgrazing with all herdsmen. Whereas the common resource could have been reasonably shared by the herdsman exhibiting some restraint, greedy local choices made by the herdsmen quickly leads to overgrazing and destruction of the pasture. Hardin [8] observes that "Freedom in a commons brings ruin to all." and convincingly argues that enforced laws, and not appeals to conscience, is necessary to avoid the *tragedy of the commons*. Recently, attention has been drawn to this problem in the context of autonomous agent systems [23].

Braess Paradox: Consider a resource sharing problem where the cost of utilizing a resource increases with the number of agents sharing it (for example, congestion on traffic lanes). Assume that initially the agents are randomly assigned to one of two identical resources. Now, if every agent opts for the resource with the least current usage, the overall system cost (cost incurred per person) increases [12]. So, the dilemma for each agent is whether or not to make the greedy choice. We will now briefly review some of the work by Glance and Hogg [4] on social dilemmas in groups of computational agents. Glance and Hogg [4] study a version of the social dilemma problem known as the Braess' Paradox [14] in the context of a traffic flow problem. Figure 1 shows agents entering the network from the bottom and choosing among several paths, moving between nodes along the indicated links. The cost of traversing a link is either constant or is directly proportional to the fraction of agents traversing it (these fractions, f_i, are used as link labels). When the link between B and C is absent, both paths ABD and ACD are equally attractive and half of the agents take the left route and the other half chose the right route. Consider the network as a highway system where each agent seeks to minimize its travel time across the network. It would seem that adding more highways can only reduce the travel time of agents. However, it turns out that the addition of an extra route can sometime decrease the overall throughput of the system i.e., increase the travel time of agents. This is an instance of the Braess' paradox. For example when a link between B and C with cost $\frac{1}{4} < x < \frac{1}{2}$ is added, a greedy decision to minimize individual costs leads each agent to choose the path ABCD. This results in the average cost per agent to be $1 + x$ which is greater than the average cost of 1.25 without the link.

4 Experimental Results

In this section, we report results from our initial experiments using GAs to solve the "Tragedy of the Commons" and the Braess' paradox problems.

4.1 GAs and the Tragedy of the Commons

We wanted to evaluate the capability of the GA to effectively evolve and maintain a population of agent strategies that are tightly coupled. This means that the performance of each individual will depend on all other individuals. The use of competition to iden-

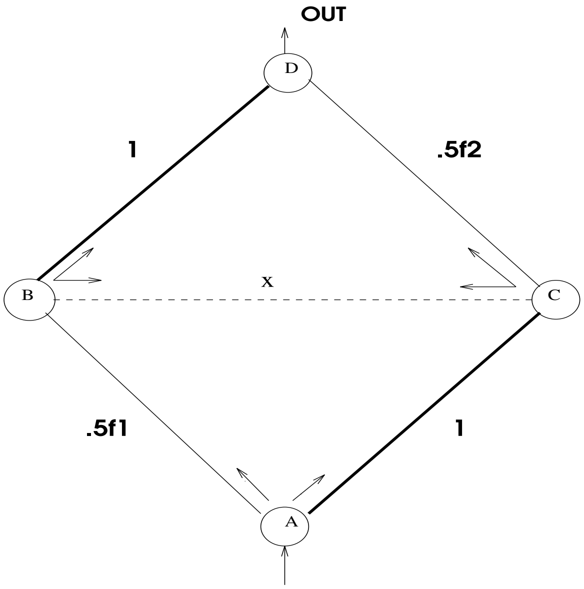

Figure 1: Example of a stand-alone Braess' Paradox.

tify and evolve robust individuals is not novel [16, 20]. However, the focus on evolution of the entire population where each and every individual's contribution is critical is new to the best of our knowledge.

We define an abstract version of the Tragedy of the Commons problem as follows: a shared resource can effectively support N units of load, but if the jointly applied load, M, exceeds N, the quality of the service received from the resource rapidly deteriorates. We call N the *critical load* of the resource. The above constraint is modeled by a utility per unit load function as follows:

$$\begin{aligned} U(x) &= C, \text{ when } x \leq N, \\ &= C * \frac{N}{x}, \text{ otherwise,} \end{aligned}$$

where C is a constant and $U(x)$ denotes the utility per unit load when a total of x units of load is applied on the system. The exponential decay of the utility function captures the rapid deterioration of quality of service from the resource when the applied load crosses a threshold.

In our GA based experiments, we assumed that each population member corresponds to a user of the resource and the chromosome represents the load applied to the resource by the corresponding user. The total load on the resource at any point in time is obtained by summing the loads from each chromosome. Each chromosome is then assigned a raw evaluation that is the product of the load it represents and the utility per unit load for the current population. Our goal was to come up with a fitness scaling function that transforms the raw evaluation to enable the evolution of a co-adapted group that uses the shared resource close to its critical load. Let $e(x, L)$ be the raw fitness of an individual who has put a load of x units on the resource when the total load imposed by all individuals is L, i.e., $e(x, L) = x * U(x)$. Let f be the corresponding scaled fitness of that individual. If f is made proportional to e, then individuals that impose higher loads will always receive higher fitness. This will invariably lead to overloading of the system and the population will be doomed to the tragedy of the commons.

Our approach was to build in individual restraint into the evaluation function without imposing centralized penalty mechanisms. When an individual receives an evaluation it can calculate the current per unit utility, u. Every individual also maintains the best per unit load utility it has seen so far, $\mathcal{U}$. The fitness function that we have used can now be specified as:

$$\begin{aligned} f(x, L, u, \mathcal{U}) &= e(x, L), \quad \text{if } u \geq \mathcal{U} \text{ or } x = 0, \\ &= \frac{1}{x} \quad \text{otherwise .} \end{aligned}$$

This *inverse scaling* fitness function encourages higher loads when the observed benefit per unit load decreases. Decreasing benefit per unit load indicates that total load on the resource has exceeded critical load. The difference between this and a centralized approach is that the fitness scaling function need not have exact or direct knowledge about the critical load of the resource.

We experimented with the above fitness function with a GA where population size is 25, crossover rate = 0.7, mutation rate = 0.01, fitness proportionate selection, replacement of worst individual, and incremental GA mode (one child produced per generation). Each chromosome was 8 bits long. So, the load that can be imposed on the common resource by one agent varied from 0 through 255. For the evaluation function, we used a critical load of 1525 and $C = 1$. If an optimal homogeneous society is designed for this environmental parameters, then every agent would put a load of 61 on the system. The total load presented on the resource as a function of generation number, and averaged over 100 runs, is presented in Figure 2. Results were similar for other critical loads where optimal homogeneous societies were feasible, that is the number of individuals (population size) exactly divided the critical load.

We found that the total load on the resource started high (due to random initializations), and then gradually settled close to the critical load. There were minor fluctuations around the critical load, due largely

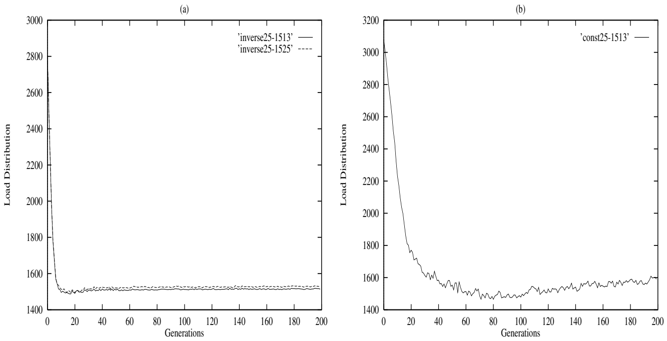

Figure 2: Experiments with the Tragedy of the Commons problem: (a) using Inverse Scaling, (b) using Constant Parabola Scaling.

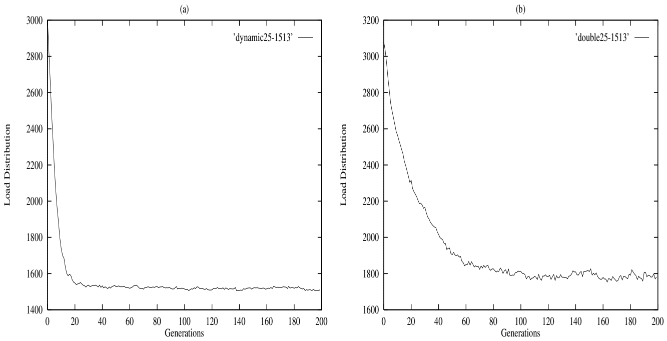

Figure 3: Experiments with the Tragedy of the Commons problem using: (a) Dynamic Parabola Scaling, and (b) Double Parabola Scaling.

to mutation. When we looked at the converged population we found that the load was equally imposed by all the population members. This is a clear demonstration of the effectiveness of the GA to evolve a coadapted population where the fitness of the population members are very closely coupled.

To further investigate the effectiveness of this mechanism, we ran experiments where the population size did not exactly divide the critical load. We used a population size of 25 and a critical load of 1513. This assured that a completely homogeneous population would not be the optimal configuration. The total load still converged close to the critical load and the distribution was more or less homogeneous, i.e., all agents applied almost the same load on the system.

In real world, with certain load constraints imposed on the system, we often find diverse population evolving rather than a homogeneous population. Some additional mechanism is required to promote diversity in the converged population. One mechanism that we investigate here is to use different parabolic functions to obtain the fitness values instead of the linear inversion scaling. These parabolic functions give higher fitness to individuals with very high or very low loads and suppresses individuals with load values in between when the observed utility per unit load decreases, thus promoting diversity.

We used following different forms of parabolic fitness functions:

Constant Parabola: As the name suggests, this fitness function is a parabola with constant focus and directrix. This function is of the form:

$$\begin{aligned} f(x, L, u, \mathcal{U}) &= e(x, L), \text{ if } u \geq \mathcal{U} \text{ or } x = 0, \\ &= a * (x-b)^2 + c \text{ otherwise,} \end{aligned}$$

where a, b, and c are appropriately chosen values. Figure 2(b) shows the load distribution in the population.

Dynamic Parabola: Constant parabolic function will always scale fitness in the same way, irrespective of the resource distribution in the current population. This requires careful choice of the parabola function that leads to optimal convergence for any arbitrary resource distribution. Hence, the constant parabola function becomes very much system dependent. One solution to this problem is a dynamically changing parabolic function which depends on the current population distribution:

$$\begin{aligned} f(x, L, u, \mathcal{U}) &= e(x, L), \text{ if } u \geq \mathcal{U} \text{ or } x = 0, \\ &= \frac{(x - (\mu + 2 * \sigma))^2)}{2 * \sigma} + c \text{ otherwise,} \end{aligned}$$

where c is a constant and μ and σ are the mean and the standard deviation of the current population respectively. Figure 3(a) shows the load distribution in the population using dynamic parabola fitness function. Dynamic parabola scaling not only brings diversity but also makes the system domain independent.

Double parabola: The single dynamic parabola scaling evolves a population comprising of individuals with very high and very low loads. To evolve diverse population in the middle range we came up with a double parabola fitness function. The fitness function is given by:

$$\begin{aligned} f(x, L, u, \mathcal{U}) &= e(x, L), \text{ if } u \geq \mathcal{U} \text{ or } x = 0, \\ &= f_1(x), \text{ if } x \leq c, \\ &= f_2(x), \text{ otherwise,} \end{aligned}$$

where,

$$
\begin{aligned}
f_1(x) &= a * \frac{(x-(\mu-\sigma))^2}{\sigma} + K, \\
f_2(x) &= \frac{(x-(\mu+\sigma))^2}{b*\sigma} + K,
\end{aligned}
$$

a, b, K are appropriately chosen constants such that $f_1(c) = f_2(c)$. Figure 3(b) shows the load distribution in the population using double parabola fitness function.

All scaling functions used above evolve a diverse population. Even though the population starts off with high initial total load, within a few generations the resources are redistributed to minimize the load on the system and maximize the overall profit. Our experiments suggest that the dynamic single parabola scaling works better than the other parabola scaling methods. It was particularly heartening that this adaptive scaling scheme generated close to optimal system performance.

The current systems always get close to the optimal resource distribution but does not necessarily reach it. We have not fine-tuned the fitness function parameters, and that can further improve performance. In particular, an appropriate choice of the c constant can improve the performance of the double parabola based scaling method. We plan to investigate an adaptive choice of c, e.g., based on μ.

We had hoped that as the parabola scaling methods are designed to introduce more diversity in the population, it would be able to produce better adapted societies compared to linear scaling techniques. This was not observed on the versions of the Tragedy of the Commons problem that we studied. This may be because the linear scaling approach was already producing very good solutions. We plan to design social dilemma problems where linear scaling is not as effective and then study the effectiveness of parabola or other non-linear scaling approaches in that context.

4.2 GAs and Braess' paradox

A genetic algorithm approach was used by Glance and Hogg [4] on a larger network (see Figure 4) which has a smaller network representing a Braess' paradox (see Figure 1) embedded in it. Each structure in the GA encoded a path taken by individual agents. Glance and Hogg found that the extra link always lowered the performance of the system and hence concluded that the GA was unable to solve the social dilemma.

In our previous work we used a optimization approach to evolve a set of agent strategies that "solves" the paradox [1]. Now, we present our experimental results using an adaptive systems approach for this problem. In our encoding we used 8 bits to represent each agent, one bit for each possible decision node in the network (the 8-bits correspond to the decision nodes 1,2,A,B,C,D,3,4). Thus for any two structures in the GA population, the same bit position now represent the same decision node in the network. For example consider the two strings S3 = `01001001` and S4 = `11000110`. The first bit forces the agent along the path between node 1 and node A (see Figure 4). Then we ignore the second bit as the decision at node need not be made. Thus, in both cases, when the agent is at node A , it proceeds to node B.

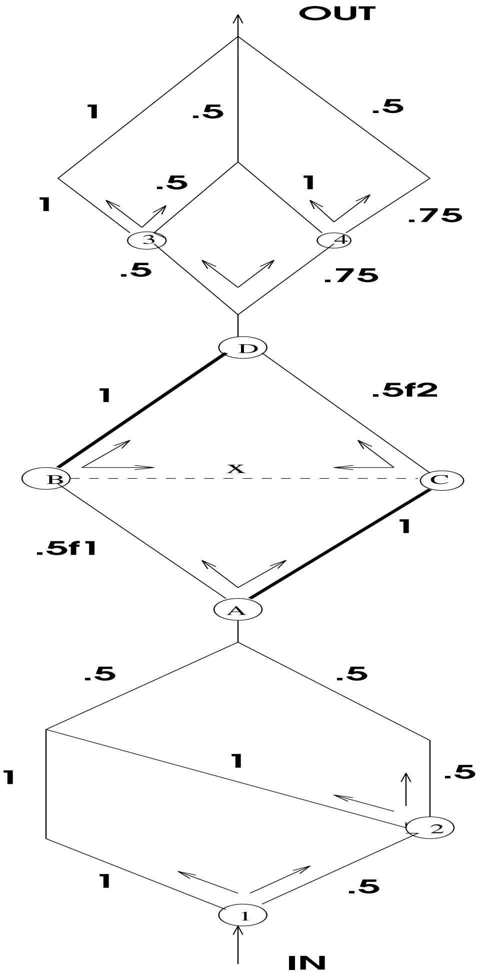

Figure 4: Example of an embedded Braess' Paradox.

Figure 5 shows the results of our experiments with 10 agents averaged over 10 runs with different costs of x link. The GA easily finds the optimal configuration if the x link is not present. If the cost of the link is too low, say, 0.2, all individuals converge to the rule that always takes the x link. Analogously, if the cost

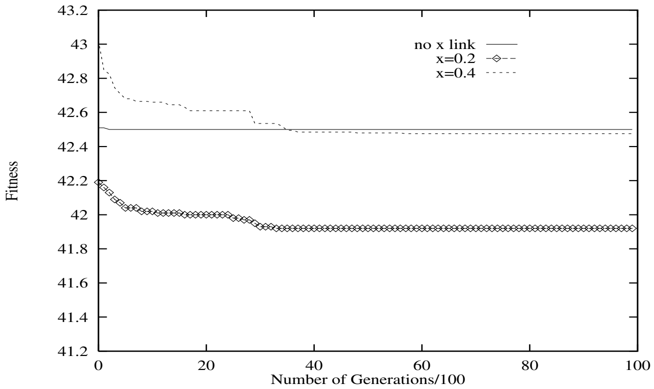

Figure 5: Experiments with the Braess's paradox problem.

is too high, the rules converge to the one that never takes the x link. The dilemma arises when the cost of x link is about 0.4 – nearly half of the individuals take the x link, while the rest avoid it. In this case, the adaptive GA takes longer to find the optimal pattern of individuals in the population than in the non-dilemma cases, but the population converges consistently to the optimal structure. Table 1 present populations at the end of typical runs for different x values. This results show that the GA is able to produce an effectively co-adapted set of structures that circumvents the Braess's paradox problem.

Performance comparison with our previous work on using a function optimization approach [1] shows that the adaptive systems approach performs a little better in the dilemma situation. But the claim in this paper is not necessarily that of superiority with respect to the function optimization approach. As mentioned before, our primary goal is to demonstrate the feasibility of the adaptive system approach for designing agent systems with effectively co-adapted individuals.

x=0.4	x=0.2	no x link
11010010	11110010	11010011
11110010	11010010	11100011
11010010	11010010	11100011
11000010	11010010	11100011
11000010	11000010	11000011
11110010	11000010	11000011
11010010	11010010	11000011
11000010	11010010	11100011
11011010	11110010	11100011
11110010	11110011	11100011

Table 1: Evolved population members for different network configurations.

5 Conclusions

We have been investigating the applicability of GAs for developing co-adapted agent societies. This is a novel application of GAs. In this paper we have shown that an adaptive system approach for using GAs for evolving agent societies faced with social dilemmas can be a viable alternative to a straightforward optimization approach that is normally used. This is not something that was easily foreseeable, and gives testimony to the robustness of the GA approach. Encouraged by our initial results from the Tragedy of the Commons and Braess's paradox problems, we plan to test the adaptive systems approach to larger, more complex social dilemma situations.

Social dilemmas are not restricted to human societies and are bound to plague artificial social systems [4, 11, 23], e.g., message congestion problems. A naive design and implementation of agent societies, therefore, will be likely to lead to ineffective utilization of resources. Given an environment, we can evolve agent societies that optimally utilize the resources available to it. The model proposed in this paper assumes an evolutionary setting, where agent societies that are better able to effectively utilize its resources are more likely to prosper over time. This model may not be appropriate in a number of domains where the agent designer is faced with designing one or more agents that will interact with other agents and little is known about the characteristics of these other agents. This model is appropriate, however, when an agent designer is designing an entire artificial agent society and wants these agents to effectively share all the resources. Since cooperative agent societies are not immune to social dilemmas [4], the mechanisms that we are studying will be useful for the design of effective cooperative societies.

Acknowledgments This work has been supported in part by an NSF CAREER award IIS-9702672.

References

[1] Neeraj Arora and Sandip Sen. Resolving social dilemmas using genetic algorithms: Initial results. In *Proceedings of the 7th International Conference on Genetic Algorithms*, pages 689–695, San Mateo, CA, 1997. Morgan Kaufman.

[2] Lawrence Bull. Evolutionary computing in multi-agent environments: Partners. In *Proceedings of the Seventh International Conference on Genetic Algorithms*, pages 370–377. Morgan Kaufmann Publishers, Inc., 1997.

[3] Kenneth A. DeJong. Genetic algorithms are NOT function optimizers. In L. Darrell Whitley, editor, *Foundations of Genetic Algorithms 2*, pages 5–18, 1993.

[4] Natalie S. Glance and Tad Hogg. Dilemmas in computational societies. In *First International Conference on Multiagent Systems*, pages 117–124, Menlo Park, CA, 1995. AAAI Press/MIT Press.

[5] Natalie S. Glance and Bernardo A. Huberman. The dynamics of social dilemmas. *Scientific American*, 270(3):76–81, March 1994.

[6] John Grefenstette. The evolution of strategies for multiagent environments. *Adaptive Behavior*, 1(1):65–89, 1992.

[7] John Grefenstette and Robert Daley. Methods for competitive and cooperative co-evolution. In Sandip Sen, editor, *Working Notes for the AAAI Symposium on Adaptation, Co-evolution and Learning in Multiagent Systems*, pages 45–50, Stanford University, CA, March 1996.

[8] Garrett Hardin. The tragedy of the commons. *Science*, 162:1243–1248, 1968.

[9] Thomas Haynes and Sandip Sen. Crossover operators for evolving a team. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Proceedings of Genetic Programming 97: the Second Annual Conference*, pages 162–167, San Francisco, CA, 1997. Morgan Kaufman.

[10] Thomas Haynes, Roger L. Wainwright, Sandip Sen, and Dale A. Schoenefeld. Strongly typed genetic programming in evolving cooperation strategies. In L. Eshelman, editor, *Genetic Algorithms: Proceedings of the 6th International Conference*, pages 271–278, San Francisco, CA, 1995. Morgan Kaufman.

[11] Tad Hogg. Social dilemmas in computational ecosystems. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 711–716, August 1995.

[12] Tad Hogg and Bernardo A. Huberman. Controlling chaos in distributed systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(6):1325–1332, December 1991. (Special Issue on Distributed AI).

[13] John H. Holland. *Adpatation in natural and artificial systems.* University of Michigan Press, Ann Arbor, MI, 1975.

[14] A.D. Irvine. How braess' paradox solves newcomb's problem. *International Studies in the Philosophy of Science*, 7(2):141–160, 1993.

[15] N.R. Jennings, K. Sycara, and M. Wooldridge. A roadmap of agent research and development. *International Journal of Autonomous Agents and Multi-Agent Systems*, 1(1):7–38, 1998.

[16] Bryant A. Julstrom. Effects of contest length and noise on reciprocal altruism, cooperation, and payoffs in the iterated prisoner's dilemma. In *Proceedings of the 7th International Conference on Genetic Algorithms*, pages 386–392, San Mateo, CA, 1997. Morgan Kaufman.

[17] W.F. Lloyd. *Two Lectures on the Checks to Population.* Oxford University Press, Oxford, England, 1833.

[18] Mitchell A. Potter, Kenneth A. De Jong, and John J. Grefenstette. A coevolutionary approach to learning sequential decision rules. In *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 373–381, San Mateo, CA, 1995. Morgan Kaufman.

[19] Narendra Puppala, Maria Gordin, and Sandip Sen. Shared memory based cooperative coevolution. In *Proc. of the International Conference on Evolutionary Computation'98.* IEEE Press, 1998.

[20] Rick Riolo. The effects and evolution of tag-mediated selection of partners in populations playing the Iterated Prisoner's Dilemma. In *Proceedings of the Seventh International Conference on Genetic Algorithms*, pages 378–385. Morgan Kaufmann Publishers, Inc., 1997.

[21] Christopher D. Rosin and Richard K. Belew. Methods for competitive co-evolution: Finding opponents worth beating. In *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 373–382. Morgan Kaufmann Publishers, Inc., 1995.

[22] Adam Smith. *The Wealth of Nations.* A. Strahan, Printer-stree; for T. Cadell Jun. and W. Davies, in the Strand, Boston, MA, 10 edition, 1802.

[23] Roy M. Turner. The tragedy of the commons and distributed ai systems. In *Working Papers of the 12th International Workshop on Distributed Artificial Intelligence*, pages 379–390, May 1993.