Evolutionary Design of a Rule-Changing Artificial Society Using Genetic Algorithms

Yun Wu
Graduate School of Systems and Information Engineering, University of Tsukuba,
Tsukuba Ibaraki 305-8573, Japan
wu@kslab.is.tsukuba.ac.jp

Hitoshi Kanoh
Institute of Information Sciences and Electronics, University of Tsukuba
Tsukuba Ibaraki 305-8573, Japan
kanoh@is.tsukuba.ac.jp

ABSTRACT

In this paper we address an artificial society in which action rules change with time. We propose a new method to design the action rules of agents in artificial society that can satisfy specified requests by using genetic algorithms (GAs). In the proposed method, each chromosome in the GA population represents a candidate set of action rules and the number of rule iterations. While the usual method applies rules in order of precedence, the present method applies a set of rules repeatedly for a certain period. Experimental results obtained using an artificial society show that the proposed method is more efficient than the usual method.

Keywords: Artificial Society, Agents, Action Rules, Genetic Algorithms, Design.

1. INTRODUCTION

Socioeconomic phenomena, cultural progress and political organization have recently been studied by creating artificial societies consisting of simulated agents [2]. Epstein and Axtell have used the sugarscape model to analyze the propagation of social systems and cultures by evaluating the evolution of agents with simple rules adapted to their environment [1].

Parisi and others proposed three sets of simulations that demonstrate how populations of artificial organisms (neural networks) tend to form social aggregations as they evolve. These simulations use genetic algorithms (GAs) to model the evolution of neural networks behaving in the same environment [3]. Kurahashi and Terano analyzed norm emergence in communal sharing by using GAs in an agent-based simulation in which the roles of agents were predetermined and acquired features were expressed as chromosomes [4]. And Billari and others studied the cultural evolution of age-marriage norms by using GAs in an agent-based model in which each individual is characterized by an age-at-marriage proscriptive norm [5].

In this paper, using a sugarscape model as a base model, we propose an efficient method for designing the action rules of the agents in an artificial society that can satisfy a demand by using a GA. This method uses “action rules” and “numbers of rule iterations” as the constituents of a chromosome in which action rules change with time. This method is expected to improve evolutionary design efficiency significantly.

We earlier proposed a method for obtaining the transition rules of one-dimensional two-state cellular automata (CAs) using a GA to solve the density classification problem [6]. Experiment results proved that applying two or more rules (that change with time) is more efficient than applying one rule. This paper shows how that method can be applied to agent simulations.

The rest of this paper is organized as follows: Section 2 outlines the research field, Section 3 introduces the coding and algorithm of the proposal method, and Section 4 describes and discusses experiments and their results.

2. OVERVIEW

Sugarscape Model

The platform we use is the sugarscape model developed by Epstein and Axtell. Sugarscape models an artificial society in which agents move over a 50-by-50 grid of cells, each of which has a gradually renewable quantity of ‘sugar’ and ‘spice’ which the agent located at that cell can eat. Moreover, each agent can find another agent with which to exchange sugar and spice. The sugarscape model is well suited to our purpose: designing an artificial society that meets a specified demand.

Environment: An environment is defined as a spatial distribution of the food (here, sugar and spice) required for an agent’s survival. If either the sugar level or spice level drops to zero, the agents will die. Initially, 400 agents are stationed in the environment at random. This is called initial configuration. We use $S(t)$ to denote a configuration at time t, so the initial configuration is expressed as $S(0)$.

Agents: People who live in the artificial society are called agents. Each agent has internal states and action rules, and each agent acts within the environment.

Internal States: The internal states of each agent are current location, vision, metabolic rate, wealth, sex, and age (Table 1).

Table 1. Internal states.

Current location	The pair of coordinates (x, y)
Vision	Agent can look to the north, south, east, and west of their current locations and can see for a distance that which varies randomly according to the agent’s ‘genetic endowment.’
Metabolic rate	The amount of food the agent consumes in every period
Wealth	The amount of food the agent has accumulated
Sex	Male or female
Age	It increases by 1 year every period.

Action Rules: Each agent has the four action rules listed in Table 2. The rules Move and Harvest describe the interactions between agents and environment. The rules Trade and Mate describe the interactions between agents.

Table 2. Action rules.

Move	Look (within the limits of one’s vision) for the unoccupied cell that has the most available food and move there.
Harvest	Store as wealth all the food available on that occasion.
Trade	Barter with neighbors for sugar and spice.
Mate	When mating conditions are fulfilled, mate with the neighbor and leave a child.

Previous Work

Agent simulation builds an artificial society autonomously from the bottom up and has been widely used to study social processes (*i.e.,* the functions and organizational structure of a social group). Epstein and Axtell have drawn several conclusions about social systems, economies, and cultural progress by using the sugarscape model to simulate social phenomena [1].

Parisi and others have examined the roles or three factors—the distribution of resources, social groups as “information centers,” and the advantages of learning from others—by simulating the evolution of populations of very simple organisms. Each organism is modeled by a neural network that receives as input some encoding of environmental information and generates as output the encoding of some movement of the organism. The simulations use GAs to model the evolution of neural networks behaving in various environments [3].

Kurahashi and Terano built an artificial social model called TRURL in order to analyze how the norm emergence in communal sharing of community functions in an information society. They used a GA to bring an artificial social model close to the actual world evolutionarily, without adjusting an intentional parameter. In TRURL an agent's predetermined features are described as a gene sequence on the chromosome expressing society. Moreover, an agent has acquired features that differ from predetermined ones by changing according to the interactions between agents [4].

Billari and others developed an agent-based model to simulate the dynamics of age-marriage norms by using a

GA. That model focused on the evolution of norms, and each individual was characterized by an age-at-marriage proscriptive norm specifying a lower acceptable limit of the age at marriage and an upper acceptable limit of the age at marriage [5].

In this paper we propose an efficient method for designing the action rules of agents that will constitute an artificial society that meets a specified demand by using a GA. In this method each chromosome in the GA population represents a candidate set of action rules and the number of iterations of each rule.

3. PROPOSED METHOD

Coding

Figure 1 shows an example of a chromosome of the usual method: action rules are arranged in applicable order. We improved the usual coding as shown in Fig. 2: each chromosome in the population represents a candidate set of R_i and M_i.

R_i indicates the i-th rule group and is a combination of the four action rules. A chromosome thus contains 4! = 24 rule groups. M_i (i=1, ···, n) indicates the number of rule iterations (n=24). That is, R_1 is applied to $S(0)$ M_1 times, then R_2 is applied to $S(M_1)$ M_2 times, and so on.

This method can produce stable growth and continuous action of agents by applying distinct rules every certain period. Our objective is to design an artificial society efficiently.

1	0	2	3	···	2	1	0	1

0 : Move　　1 : Harvest
2 : Trade　　3 : Mate

Fig. 1. Chromosome of the usual method.

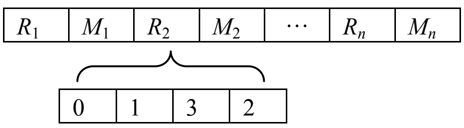

Fig. 2. Chromosome of the proposed method.

Fitness Calculation

The purpose of this paper is to obtain action rules that generate an affluent society efficiently. We define an affluent society as follows: ① The population of society exceeds a given number. ② Each agent has more wealth than the wealth taken from parents (initial wealth). ③ The minimum value of wealth in the society exceeds a given value. The corresponding fitness function F can be calculated by the following formula:

$$F = f_1(W_{min}) + f_2(\Delta W) + f_3(P) \quad (1)$$

where

W_{min}: minimum value of wealth in the society
ΔW: mean increase of the agent's wealth
$\Delta W = \Sigma\ (W - W_{ini})\ /\ P$
W: present wealth
W_{ini}: initial wealth
P: population

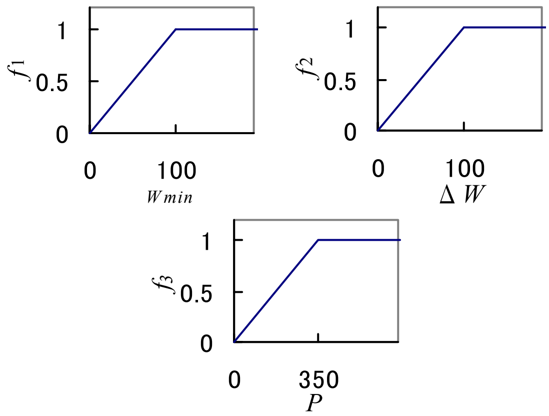

Fig. 3. Fitness function.

Figure 3 shows example of f_1, f_2, and f_3 when the maximum value of F is three.

Genetic Operation

After N_{pop} individuals are generated as an initial population, the following steps are repeated for N_{gen} generations.

Step 1: Generate a new initial configuration $S(0)$ and calculate the fitness on it for each individual in the population.

Step 2: Rank the individuals in order of fitness and copy the top N_{elit} elite individuals to the next generation without modification.

Step 3: Obtain the remaining ($N_{pop} - N_{elit}$) individuals for the next generation by producing single-point crossovers between individuals randomly selected from N_{elit} elites and individuals selected from the whole population by roulette wheel selection.

Step 4: Mutate the rule R_i in the offspring from crossover by substituting other action rules with probability 0.05.

4. EXPERIMENTS

Experimental Method

Experiments were conducted under the following conditions: the population size N_{pop}=50, the upper bound of generations N_{gen}=100, the number of elites N_{elit} =10 for the GA. The possible internal states of the agents are listed in Table 3.

Table 3. Internal states of agents.

Vision	1 to 10
Metabolic rate	1 to 4
Initial wealth	20 to 60
Sex	Male, Female
Age	Start from 0 years old
Lower acceptable limit of mating age	12 to 15
Upper acceptable limit of mating age	Man: 50 to 60 Women: 40 to 50
Lifespan	60 to 100

Experimental Results

Comparison of Fitness: The results of comparing the proposed method with the usual method at different desired values of W_{min}, ΔW, and P are listed in Table 4. Let the number of rule iterations be $1 \leq M_i \leq 10$. Each value in the rightmost two columns is the average fitness (standard deviation) of five trials. Each trial was evolved for 50 generations. We see from Table 4 that the fitness of the usual method at the last generation is 2.0 to 2.3, while that of the proposed method is 2.83 to 3.0. That is, the proposed method has succeeded in generating the artificial society that meets the user's demand.

Table 4. Comparison of fitness.

W_{min}	ΔW	P	Usual method	Proposed method
50	50	120	2.3 (3%)	3.0 (0%)
50	50	150	2.2 (12%)	3.0 (0%)
70	70	200	2.0 (0.2%)	2.96 (6%)
100	100	350	2.0 (0.3%)	2.83 (12%)

Time Series Behavior of Artificial Society: We are also interested in the change of the three social indicators W_{min}, ΔW, and P. The desired values of W_{min}, ΔW, and P were respectively set to 100,100 and 350. We pick out the elite individual obtained from this experiment, and apply it to initial configuration to examine the details of the changes in these indicators during the social evolution.

Figures 4 and 5 show the time series behavior of the wealth and the population of the society. We can see from Fig. 4 that the W_{min} and ΔW of artificial society increase gradually with the proposed method as well as with the usual method. And we can see in Fig. 5 the population decreases gradually with the usual method, while with the proposed method it first decreases gradually, then becomes stable, and later increases again. We thus can conclude that the proposed method can generate a rich society with a stable population.

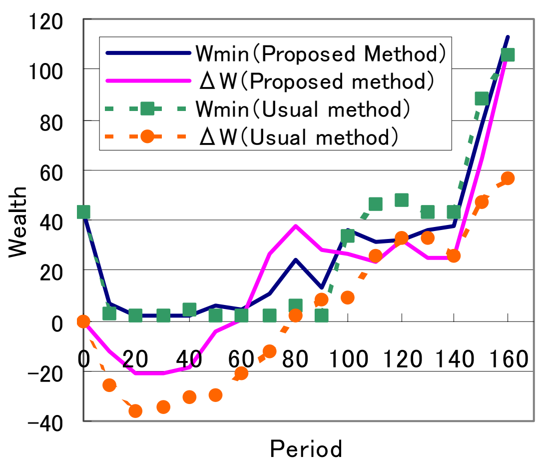

Fig. 4. Time series change of wealth.

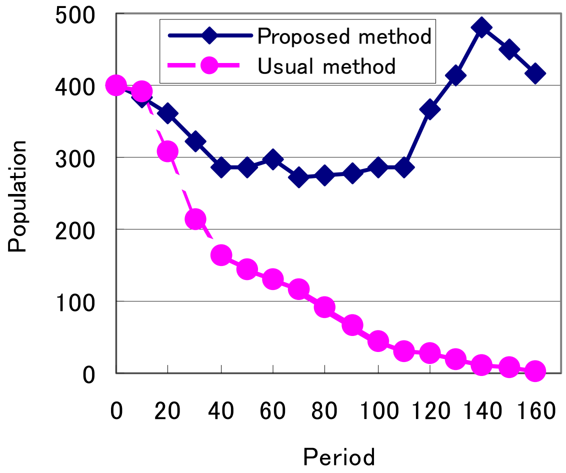

Fig. 5. Time series change of population.

Discussion

Evolutional in the Proposed Method: The data discussed below were collected in the experiment using the W_{min}, ΔW, and P values listed in the last line of Table 4. We pick out an individual that has a mean fitness value at the initial generation and an elite individual at the last generation.

The evolutional result of each indicator is listed in Table 5, where it can be seen that when the proposed method was used, all indicators had values equal to or greater than the desired value in the last generation.

The ratios of action rules included in the elite individual are listed in Table 6, where it can be seen that the Move rule and the Harvest rule increased while the Trade rule decreased. It thus seems that the Move and Harvest rules play important roles in the evolution of an artificial society that meets a user's demand.

Table 5. Evolutional result of each indicator.

Generation	W_{min}	ΔW	P
Initial generation	30	20	217
Last generation	100	113	369
Target value	100	100	350

Table 6. Ratios of action rules.

Generation	Move (0)	Harvest (1)	Trade (2)	Mate (3)
Initial generation	25%	25%	25%	25%
Last generation	29%	37%	10%	24%

Number of Agents that Actually Execute Action Rules: As described in section "Time series behavior of artificial society", an affluent artificial society can be generated by using the proposed method. To investigate the reason, we examined the number of agents that actually used the action rules.

Table 7 shows the comparison of the proposed method with the usual method. We pick out elite individual and apply it to the initial configuration. Based on the application order of the action rules of the chromosome, for those agents that survive in every period, we measured the number of agents which actually executed the given action rules. The experimental results are shown in Table 7.

We can see from Table 7 that when the usual method was used, more agents executed the Move rule than the Harvest rule. When the proposed method was used, in contrast, more agents executed the Harvest rule than the Move rule. Thus with regard to the collection of wealth, the proposed method is more effective than the usual method. And the number of agents executing the Mate rule when the proposed method was used is 14.4 times the number executing it when the usual method was used. This is effective in increasing of the population of a society.

Table 7. Number of agents actually executing action rules.

Action rule	Usual method	Proposed method
Move	17 082	65 753
Harvest	13 279	73 618
Trade	9 137	16 566
Mate	53	766

The changes of the number of agents actually executing the Trade and Mate rules are shown in Figs. 6 and 7. We can see from Fig. 6 that the number of agents executing the Trade rule decreases to 0 when the usual method is used but increases to 337 when the proposed method is used. And we can see from Fig. 7 that few agents actually execute a mate rule when the usual method is used but that agents mate actively when the proposed rule is used.

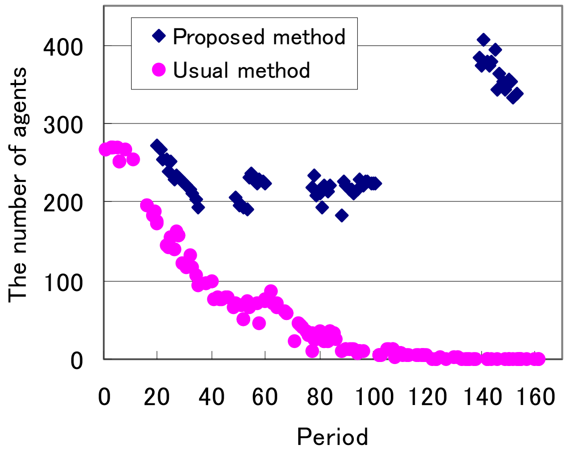

Fig.6. Time series change of the number of agents trading.

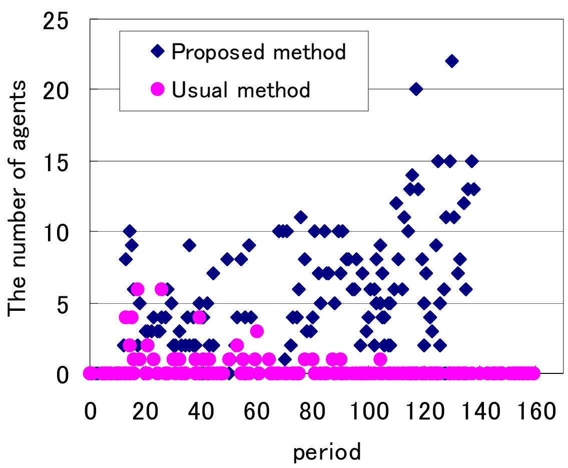

Fig.7. Time series change of the number of agents mating.

5. CONCLUSIONS

In this paper we address an artificial society in which action rules change with time. We examined a new method to obtain action rules of agents in artificial society that can satisfy requests by using genetic algorithms. As a result, we confirmed that an artificial society can be evolved efficiently by applying different sets of rules for a certain period.

To confirm the effectiveness of our proposed method, we compared the numbers of agents actually executing action rules when the proposed and usual methods were used. This comparison showed that an artificial society that satisfies a user's demand is generated more efficiently when the proposed method is used. In a forthcoming study, we are going to add more action rules—for example, a war rule and a cultural transformation rule—and further evaluate our method experimentally. We will also use other detailed analyses to evaluate our experiments.

REFERENCES

[1] J.M. Epstein, and R. Axtell, **Growing Artificial Societies:** Social Science from the Bottom Up, Brookings Institution Press, The MIT press, 1996.

[2] N. Gilbert, and K.G. Troitzsch, **Simulation for the Social Scientist,** Open University Press, Buckingham. Philadelphia, 1999.

[3] D. Parisi, U. Piazzalunga, F. Cecconi, and D. Denaro, "Social Aggregations in Evolving Neural Networks", Proc. Of **4th European Workshop on Modelling Autonomous Agents in a Multi-Agent World,** MAAMAW'92 (Lecture Note in Artificial Intelligence 830), pp. 41-54, 1992.

[4] S. Kurahashi, and T. Terano, "Analyzing Norm Emergence in Communal Sharing via Agent-Based Simulation", **LNAI 2253,** pp. 88-98, 2001.

[5] F.C. Brillari, A. Prskawetz, and J. Fürnkranz, "The cultural evolution of age-at-marriage norms", **MPIDR WORKING PAPER WP 2002-018,** 2002.

[6] H. Kanoh, and Y. Wu, "Evolutionary Design of Rule Changing Cellular Automata", Proc. of **Seventh International Conference on Knowledge Based Intelligent Information and Engineering Systems** (Lecture Note in Artificial Intelligence 2773), pp. 258-264, 2003.