Collective Behavior of Reinforcement Learning Agents

Tatsuo Unemi
Department of Information Systems Science, Soka University
1-236 Tangi-cho, Hachioji, Tokyo 192, Japan
unemi@iss.soka.ac.jp and
Laboratory for International Fuzzy Engineering Research
Siber Hegner Bldg. 3rd Fl., 89-1 Yamashita-cho, Naka-ku, Yokohama 231, Japan
unemi@fuzzy.or.jp

Abstract: This paper describes a challenge of investigating what sorts of collective behavior could emerge by a group of learning agents. Reinforcement learning is a suitable framework to consider the design of an autonomous agent because it is on learning by delayed reward and blame. We propose taxonomy of types of interaction among learning agents, show some experimental results of computer simulation on a group of reinforcement learning agents in a task of foraging and collision avoidance, and then discuss about ethological, sociological, and engineering application of a variety of phenomena.

1 Introduction

Through many researchers' efforts of trying to understand the information processing mechanism on human intelligence and to apply that knowledge to designing intelligent machines, it has been obvious that learning is one of the essential functions required to realize an intelligent behavior. In the other hand, distributed autonomous systems have also been mentioned as one of the advanced technology to make an intelligent system more robust and capable of more complex and larger scale tasks. There are many research activities of these areas, such as [1] and [2] in Japan.

This paper describes a challenge to investigate what sorts of collective behavior could emerge by a group of learning agents. We are expecting this challenge to contribute to both engineering and sociological research. Intuitively saying, if we would execute a sophisticated simulation, we could observe some kinds of social phenomena such as chaotic patterns by interaction among many agents, acceleration of learning by competence, and hierarchical differentiation of behavioral styles. However, it is difficult to find any framework to catch this area clearly, so far.

The following sections describe current activity of two related fields, a taxonomy of types of interaction among learning agents, some experimental results of computer simulation on a group of reinforcement learning agents on a task of foraging and collision avoidance, and then discuss based on the observation from the experiments.

2 Collective Behavior of Autonomous Robots

In recent few years, a number of researchers are getting to agree that a very simple control mechanism for a robot, such as a set of reactive rules, can be superior to a traditional sophisticated control architecture in many application fields, for instance [3, 4, 5]. Studies on collective behavior by a number of autonomous mobile robots of simple control mechanism are also active, inspired by flock of birds, school of fish and social insects, such as [6, 7, 8, 9, 10].

However, almost all of these works do not mention any type of learning, but only use fixed control rules designed by human. There are studies on self-adjustment of control parameters[11] and adaptation through an evolutional process using genetic programming[12], but we can find no literature so far on emergence of collective behavior by learning agents, which may be interesting for sociologists and A-Life people.

3 Reinforcement Learning

We employ a framework of reinforcement learning[13] to design the learning algorithm of a single agent, because it is suitable to consider an autonomous agent which learns by delayed reward and blame. There are some studies on the mathematical foundation and the extension toward some fielded application specially in robotics, such as [14, 15, 16, 17, 18, 19]. Almost all of these researches are on a single learner, but recently a paper concerning learning by multi-agent was presented[20] which focused on comparison of learning performance between some kinds of information sharing among agents.

The rest part of this section describes a preliminary formalism of reinforcement learning and a brief introduction of an instance-based reinforcement learning method employed for our experiments.

3.1 Formalism

Difference against an ordinary definition is that reinforcement signal is not coming from environment but is calculated by internal function named *evaluator* which refers the sensory input data. This idea seems more suitable

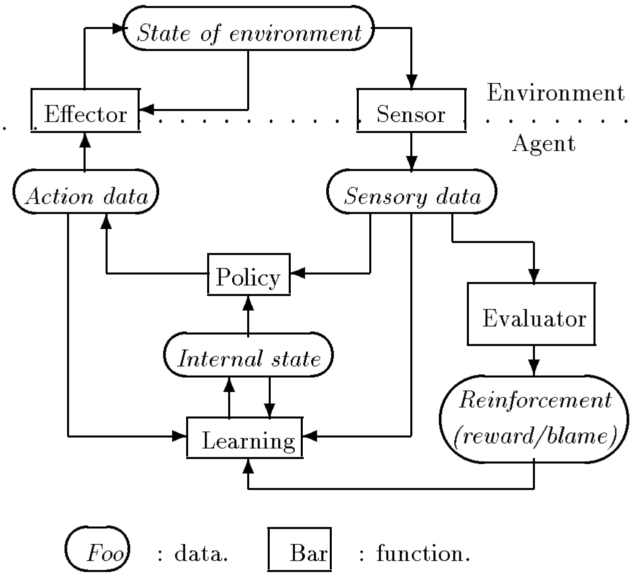


Figure 1: Block diagram of information flow of a learner.

when a learner is seen as an autonomous agent with ability of unsupervised learning.

Figure 1 shows the block diagram of information flow in a single agent. Table 1 indicates relations between functions and data.

The state of environment is determined from its previous state and the action of the agent. The sensor calculates the sensory data from the state of environment, and the function named *policy* computes the action data of the agent referring the sensory data and the internal state. The internal state is changed by the function named *learning* based on the sensory data, the action data, the reinforcement signal, and the previous internal state. These five functions and five kinds of data determine the specification of reinforcement learning.

3.2 Instance-based Reinforcement Learning Method

Many sorts of methods to realize reinforcement learning have been proposed, such as using artificial neural networks, look-up tables, and so on. The method employed in our experiments is an instance-based approach we proposed before[22], which is applicable to a domain where the input is a vector of real numbers and the output is a symbol selected from relatively small finite set. In this method, a set of memory of agent's concrete experience is used as an internal state of an agent. The learning function stores a tuple of the sensory data, the action data and its estimated evaluation in memory verbatim. The policy function decides the action by retrieving a similar and valuable experience from memory. If it has no tuple similar and valuable enough, it takes a random action.

The evaluation of the experience at time t is a *discounted cumulative rewards*, which is defined as

$$v_t = \sum_{i=t}^{\infty} \gamma^{i-t} \cdot r_i$$

where γ is constant of $0 < \gamma < 1$, and r_i is a reinforcement signal at time i. Each experience is stored in the queue of fixed length N while the reinforcement signal is zero, and the estimated value $\hat{v}_t$ of v_t is computed when the reinforcement signal becomes non zero, where the difference value for its modification is defined as

$$\Delta\hat{v}_j = \gamma^{i-j} \cdot r_i$$

where $r_i \neq 0$.

The key issue on any instance-based approach is how to avoid infinite increase of memory elements[23]. In our method, when the agent suffered the new experience which is very similar to an old experience already stored in memory, it discards the old one if the estimated evaluation of the new one is greater than of the old one, and otherwise it modifies the estimated evaluation of the old one. Additionally, it manages the reliability of each experience to forget the least reliable one when the capacity of memory has been exceeded.

One modified version of this method for learning only by negative reinforcement has been applied to cart-pole (inverted pendulum) balancing problem, and it is certified to perform better enough to realize a real-time control[24].

4 Taxonomy of Learners' Group

We can use the following features when we consider a taxonomy of the types of learners' group.

- Which data are shared among learners?
- Which functions are shared among learners?
- Which data are of same type among learners?
- Which functions have same definition among learners?

That is, degree of information sharing and similarity of definition on both data and functions specify the type of group of learners. Let us consider about some typical cases from this point of view.

4.1 Sharing Environment

One of the simplest types of group is of sharing only their environment, which means they are in the same local space. In some domain, it is required to distinct among local, global and intermediate range of environment shared.

Table 1: Relations between functions and data.

		Function				
		Effector	Sensor	Evaluator	Policy	Learning
Data	Environment	D/R	D	–	–	–
	Action data	D	–	–	R	D
	Sensory data	–	R	D	D	D
	Reinforcement	–	–	R	–	D
	Internal state	–	–	–	D	D/R

D : domain, R : range, – : not related.

It may be also possible to consider a group in which the members are not sharing their environment, but it is hard to find any kind of collective behavior by them. Sharing environment is the least requirement for collective behavior of autonomous agents.

Of course, it is not necessary for the data of the environment shared by the agents to be quite equal each other, but any agent must share some portion of the data with another agent. We can draw a simple graph where a vertex is an agent and an edge presents the data sharing between agents. It is meaningful to mention a collection of agents as a single group only when this graph is connected.

4.2 Homogeneous vs. Heterogeneous

Variety of the agent types is also an important feature, which concerns the similarity of definition of data and functions. When we design a huge number of agents such as in swarm intelligence[10], it is very difficult to specify many different functions for each agent, so we need some degree of common specification among them. We can also design a group of a variety of specialists such as CEBOT[21], which provides efficient realization of various types of tasks.

In the natural animal world, we can observe both types of group behavior. Social insects, such as bees, ants and termites, have intrinsic morphological differentiation. Simultaneously, it is known that there are social hierarchy in a society of wild dogs, horses and monkeys.

One interesting issue on a group of homogeneous learners is the emergence of hierarchical differentiation of individual behavioral styles, which means a functional change from homogeneous to heterogeneous.

4.3 Sharing and Difference of Goals

Specially in the case of learning agent, it is important to consider the relation between the goals of agents. It is an issue about feature of the evaluator. If the agents share a common goal, some of them may become idle because the others achieve their common goal. In this case, help behavior may also emerge. When each agent has a same goal but goal achievement of an agent does not have an effect on the others, there arises conflict between agents, and we can expect that conflict avoidance behavior may emerge.

4.4 Communication

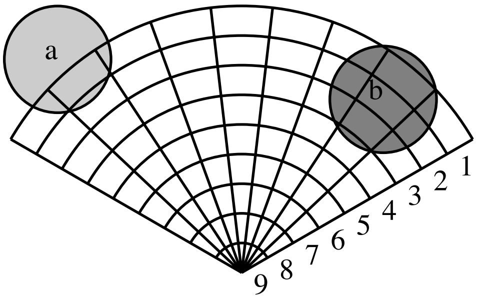
Communication between agents can be seen as a method of information sharing, but usually it means partial sharing.

There are many interesting issues concerning communication between learners, such as emergence of communication protocol and language, strategy of active query, and so on. Emergence of language or the origin of natural language is the big problem including philosophical issues on human beings.

5 Experiments

We design a group of reinforcement learning agents and their environment as a computer program based on our previous work which proposed an instance-based reinforcement learning method examined by a foraging task[22]. There is no communication between the agents, and they have quite same specification of data types and functions except the evaluator. We examined two cases where all of agents have same function of evaluator and where the half of them have the same but the others have another kind.

Each agent is a kind of artificial insect which roams around two-dimensional Euclidean space to seek its favorite foods avoiding obstacles and other insects. The *insect* has two sorts of sensors, vision and touch, and has one sort of action, argument of turn. A datum of vision is a vector of nine pairs of integers. One integer of the pair is a value from 0 to 8 which indicates the distance to the nearest object in the corresponding part of view. And the other integer indicates the sort of the object. Each element of the vector is corresponding to the distinct di-



$$\text{Vision data} = \begin{pmatrix} 1 & 2 & 1 & 0 & 0 & 0 & 3 & 4 & 3 \\ a & a & a & - & - & - & b & b & b \end{pmatrix}.$$

Figure 2: Sight of the insect.

rection of the view as shown in Figure 2. A datum of touch is the sort of object the insect touched at that step. The action of the insect is to change its position and direction in the world. In each of simulation steps, it walks by a constant length of step and turns left or right by some amount of argument according to the output which is an integer -1 and 1. The evaluator outputs the value 1 when the insect touches its favorite food, -1 when it touches another kind of food, an obstacle and another insect, and 0 if nothing, so that the objective of the insect is to feed its favorite foods as frequently as possible, and simultaneously to avoid collision.

Formally saying, the input on each learning cycle is a two-dimensional array of integers, the output is a symbol selected from *a priori* known finite set, and the reinforcement from the environment is $+1$, 0 or -1 usually being 0, that is, in the manner of delayed reinforcement.

Each food disappears when the insect touches it, and it appears again when the insect has walked away from it for some amount of distance.

The above specification is common among experimental settings shown in this paper, and we prepared the following two kinds of settings.

1. All of insects have same preference of foods and there is no food of another kind in the world.

2. Two kinds of insects and their corresponding favorite foods exist in the world. The number of insects of each kinds are equal.

We examined the cases of four insects and eight insects for each of the two kinds of setting shown above, totally four cases. Figure 3 shows an example of their traces.

In all cases in these experiments, we observed that each learner adapts its environment step by step even though fluctuation of the performance occurred sometimes. As we previously expected, the more the number of agents is, the more learning is difficult.

6 Discussion

In this section, we discuss about emergence of collective behavior by learning agents based on the results of experiments.

6.1 Effects of the Number of Agents

Learning becomes more difficult when the environment is not stational. In the situation where a number of learners are in the same space and they can observe each other, stationality of the environment is rapidly lost as the number of agents increases, because the environment of one agent includes the other agents observed and the behavior of learner changes through the learning process. It is obvious phenomena since learning is a change occurring in the internal state to alter its performance so that it achieve its own objective.

These effects can be said to come from the density of agents rather than the number of them, becase the probability of encountering of the agents becomes low when the agents goes more sparse.

6.2 Relation between Different Species

An agent which has a different kind of favorite foods can be seen as of a different species. It is expected that distinction of living place would emerge. Yes, we observed this phenomena in the experiments of the case of two kinds of foods. However, the reinforcement leaning agent always has possibility to take a random walk to explore the world to seek unknown paradise. So, sometimes the distinction is destroyed by invaders who are tending to explore the world. If all of agents take a conservative exploration strategy, that is, they have never tend to take any risky action when they know any action not so wrong, this kind of fluctuation is harder to occure. Note the fact that conservatism make the learning rate low. This means that there is a dilemma between learning rate and stationality of collective behavior.

7 Conclusion

The above experiments are only in the first stage of our challenge to collective behavior of autonomous learning agents. To understand more about this issue, we need to try a lot of other settings such as goal sharing, more heterogeneous agents, large scale space, and so on. And, we need also a quantitative analysis through statistically enough times of experiments. These are our works in near future.

Both distributed autonomous system and learning system must be key technologies to build a intelligent machine in the next century. The author hopes that this research can make any contribution to both designing a

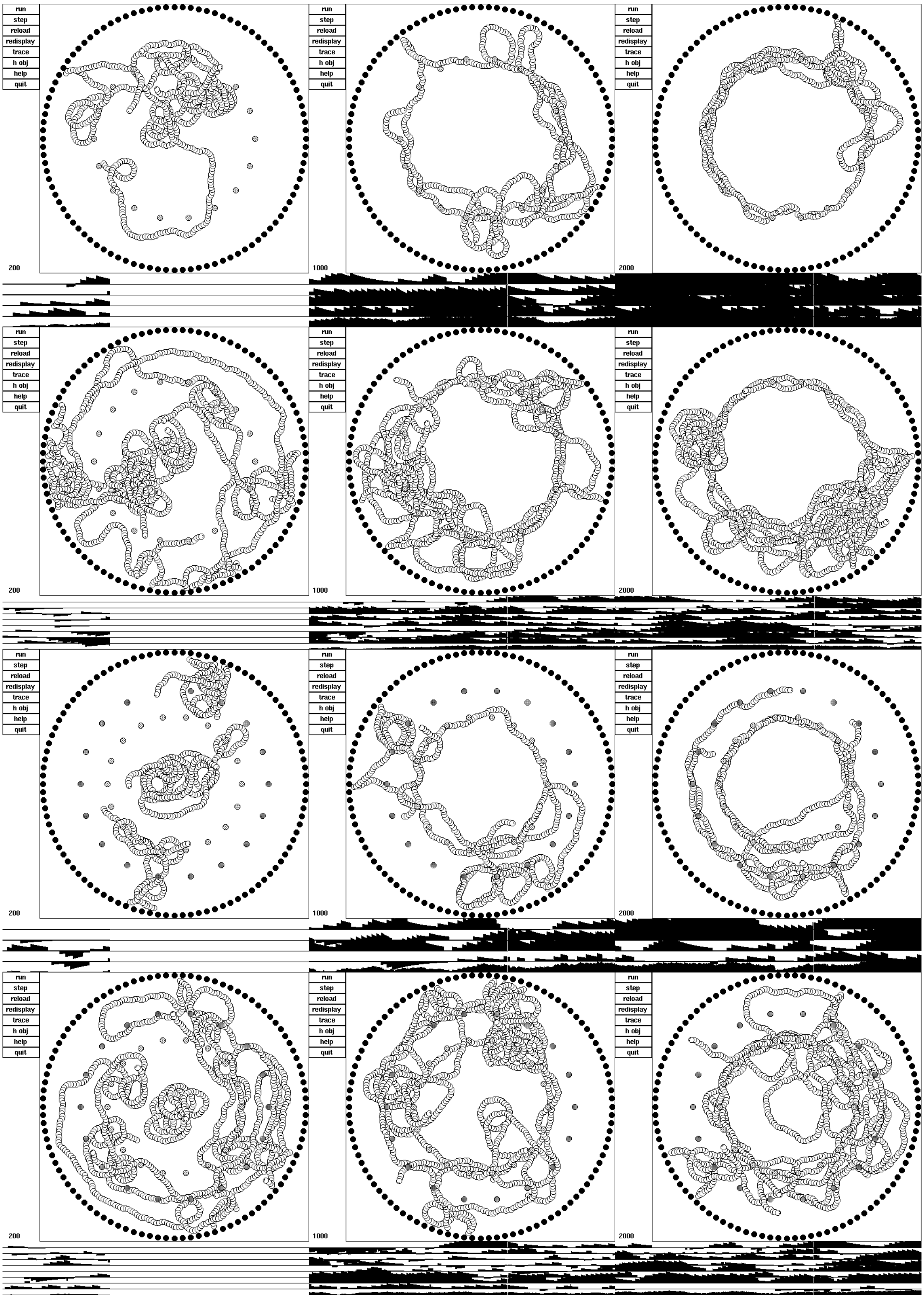


Figure 3: Experimental results of simulation. The cases are (1) one kind of foods and four agents, (2) one kind of foods and eight agents, (3) two kinds of foods and four agents, and (4) two kinds of foods and eight agents from top to bottom, and (a) trace of from the start to 200th step, (b) trace of from 800th to 1,000th step, and (c) trace of from 1,800th to 2,000th step from left to right respectively in each row.

more intelligent machine and analyzing complex social phenomena in human society.

Acknowledgement

The author thanks Toshio Fukuda and Fumihito Arai at Nagoya University for providing us an opportunity to present this research, and also Toshio Terano of LIFE to support this work.

References

[1] Asama, H. *et al* (eds.): Distributed Autonomouse Robot System, RIKEN (1992)

[2] Nakashima, H. (ed.): Multi Agent and Cooperative Computation I, Kindai Kagaku Sha, (1991) in Japanese.

[3] Brooks, R.: A Robust Layered Control System for Mobile Robot, *IEEE Journal of Robotics and Automation,* RA-2, pp. 14–23 (1986)

[4] Brooks, R.: Intelligence without Reason, *12th International Conference on Artificial Intelligence,* pp. 569–595 (1991)

[5] Maes, P.: Behavior-based Artificial Intelligence, *From Animals to Animats – Simulation of Adaptive Behavior II,* pp. 2–10 (1993)

[6] Mataric, M.: *From Animals to Animats – Simulation of Adaptive Behavior II,* (1993)

[7] Parker, L. E.: Adaptive Action Selection for Cooperative Agent Team, *From Animals to Animats – Simulation of Adaptive Behavior II,* pp. 442–450 (1993)

[8] Kube, C. R. and H. Zhang: Collective Robotic Intelligence, *From Animals to Animats – Simulation of Adaptive Behavior II,* pp. 460–468 (1993)

[9] Arkin, R. C., T. Balch and E. Nitz: Communication of Behavioral State in Mutil-agent Retrieval Tasks, *1993 IEEE International Conference on Robotics and Automation,* pp. 588–594 (1993)

[10] Beni, G. and J. Wang: Technical Problems for the Realization of Distributed Robotic System, *1991 IEEE International Conference on Robotics and Automation,* pp. 1914–1920 (1991)

[11] Maes, P. and R. A. Brooks: Learning to Coordinate Behaviors, *Eighth National Conference on Artificial Intelligence,* pp. 796–802 (1990)

[12] Koza, J. R.: Evolution of Emergent Behavior, in J. R. Koza, *Genetic Programming: on The Programming of Computers by Means of Natural Selection,* MIT Press, pp. 329–355 (1992)

[13] Sutton, R. (ed.): Reinforcement Learning, Kluwer Academic (1993) also in Special issues on Reinforcement Learning, *Machine Learning,* Vol. 8, No. (1992)

[14] Lin, Long-Ji: Scaling Up Reinforcement Learning for Robot Control, *Tenth International Conference on Machine Learning,* pp. 182–189 (1993)

[15] Clause, J. A. and P. E. Utgoff: A Teaching Method for Reinforcement Learning, *Ninth International Conference on Machine Learning,* pp. 92–101 (1992)

[16] Kaelbling, L. P.: Hierachical Learning in Stocastic Domains: Preliminary Results, *Tenth International Conference on Machine Learning,* pp. 167–173 (1993)

[17] Whitehead, S. D.: A Complexity Analysis of Cooperative Machanisms in Reinforcement Learning, *Ninth International Conference on Machine Learning,* pp. 607–613 (1991)

[18] Mahadevan, S.: Enhancing Transfer in Reinforcement Learning by Building Stocastic Models of Robot Actions, *Ninth International Conference on Machine Learning,* pp. 290-299 (1992)

[19] Moore, A. and C. G. Atkeson: Memory-Based Reinforcement Learning: Converging with Less Data and Less Read Time, in J. H. Connel and S. Mahadevan (eds.), *Robot Learning,* Kluwer Academic, pp. 79–103 (1993)

[20] Tang, Min: Multi-agent Reinforcement Learning: Independent vs. Cooperative Agents, *Tenth International Conference on Machine Learning,* pp. 330–337 (1993)

[21] Fukuda, T., T. Ueyama and F. Arai: Control Strategy for a Network of Cellular Robots, *1991 IEEE International Conference on Robotics and Automation,* pp. 1616–1621 (1991)

[22] Unemi, T.: Instance-based Reinforcement Learning Method, *Journal of Japanese Society of Artificial Intelligence,* Vol. 7, No. 4, pp. 697–707 (1992) in Japanese.

[23] Aha, D. W., D. Kibler and M. K. Albert: Instance-Based Learning Algorithm, *Machine Learning,* Vol. 6, pp. 37–66 (1991)

[24] Unemi, T.: Learning not to Fail by an Instance-based Reinforcement Learning Method, *Journal of Japanese Society of Artificial Intelligence,* Vol. 7, pp. 1001–1008 (1992) in Japanese.