ORIGINAL ARTICLE

Keiki Takadama · Koichiro Hajiri · Tatsuya Nomura
Michio Okada · Shinichi Nakasuka · Katsunori Shimohara

Learning model for adaptive behaviors as an organized group of swarm robots

Received: February 13, 1998 / Accepted: March 13, 1998

Abstract This paper describes a novel organizational learning model for multiple adaptive robots. In this model, robots acquire their own appropriate functions through local interactions among their neighbors, and get out of deadlock situations without explicit control mechanisms or communication methods. Robots also complete given tasks by forming an organizational structure, and improve their organizational performance. We focus on the emergent processes of collective behaviors in multiple robots, and discuss how to control these behaviors with only local evaluation functions, rather than with a centralized control system. Intensive simulations of truss construction by multiple robots gave the following experimental results: (1) robots in our model acquire their own appropriate functions and get out of deadlock situations without explicit control mechanisms or communication methods; (2) robots form an organizational structure which completes given tasks in fewer steps than are needed with a centralized control mechanism.

Kew words Organizational learning · Collective adaptive behavior · Reinforcement learning · Self-evaluation · Emergence · Multiple robots

K. Takadama (✉)
ATR Human Information Processing Research Laboratories
University of Tokyo, 2-2 Hikaridai, Seikacho, Soraku-gun, Kyoto 619-0288, Japan
Tel. +81-774-95-1088; Fax +81-774-95-1008
e-mail: takadama@hip.atr.co.jp/keiki@ai.rcast.u-tokyo.ac.jp

K. Hajiri · T. Nomura · K. Shimohara
ATR Human Information Processing Research Laboratories, Kyoto, Japan

M. Okada
ATR Media Integration and Communications Research Laboratories, Kyoto, Japan

S. Nakasuka
University of Tokyo, Tokyo, Japan

This work was presented, in part, at the Second International Symposium on Artificial Life and Robotics, Oita, Japan, February 18–20, 1997

Introduction

In space, robots are essential to support human work or to complete given tasks without human control, and many methods have been proposed to reduce the cost of work in space and to prevent danger to humans. One of these methods is the control of multiple robots. In this case, it is important for the robots to have some mechanisms for self-evaluation of their adaptive behaviors to improve their collective performance. The easiest way of evaluating behaviors is to employ a global evaluation function as an explicit control mechanism. In this situation, robots communicate with a centralized control system and perform according to orders from the system.[1] However, when a global evaluation function cannot be employed, robots have to communicate with other robots to evaluate their behavior[2,3] or broadcast information to get an evaluation of their behavior.

This global evaluation function guarantees the rational collective behavior of multiple robots, and the communication approach supports cooperation among robots,[4,5] but these both make communication costs high. In space in particular, since the amount of possible communication is limited by the weight and electrical power of the robots, it is very important to reduce communication costs. Furthermore, since communication traffic increases as the number of robots increases, it is difficult for robots to perform tasks which require cooperation with each other in real time. Moreover, it is generally difficult to define an appropriate global evaluation function which assigns an optimal and rational behavior to all robots. To overcome these issues, we propose a novel organizational learning model which enables robots to complete given tasks without explicit control mechanisms or communication methods, and which improves their collective performance by forming an effective organizational structure between them.

This paper is organized as follows. The next section explains the concept of organizational learning, and the following one gives details of the organizational learning model with learning classifier systems. We then proposed a novel reinforcement learning for multiple robots. An example of a truss construction task with multiple robots is given in the next section, followed by a discussion of the experimental results and our conclusions.

Organizational learning

Definition of organizational learning

Research on organizational learning has developed in the context of management organizational sciences, and much research focusses on economic market systems or human organizations.[6–9] In management organizational sciences, organizational learning is generally defined as follows: "*Organizational learning* is the creating, acquiring, and transferring of *distinctions* and *practices* in the organizations. It is effective if it increases the organization's fitness in its environment. Organizational learning implies behavior modification, including changes in relationships, in order to create the conditions for creating, acquiring, and transferring distinctions and practices.[8]

In organizational learning, there are two research streams: a top-down approach and a bottom-up approach. The former aims to analyze the phenomena in one organization, and seeks for a methodology to improve organizational performance. The latter, on the other hand, aims to analyze phenomena caused by the activities of other organizations, and studies the effects of organizational structures in organizational problem solving. Since the latter stream of organizational learning is expected to enable robots to acquire their own appropriate functions autonomously and form an organizational structure which completes given tasks effectively, we have applied it to the problems of multiple robots. Furthermore, we have also started to apply part of this concept of organizational learning to printed circuit board design,[10,11] a group dialogue model,[12] and multiple robots.[13]

Organizational structure and functions

Before we describe our organizational learning model, two technical terms are defined.

- *Organizational structure*: A structure in which robots acquire their own functions, as shown in Fig. 1.
- *Function*: A sequence of behaviors determined by rules.

Because organizational structure changes dynamically, robots must adapt to a new organization to maintain or improve organizational performance. For effective adaptation to a new organization, robots in our model learn adaptive strategies as an order of using rules, and thus acquire a long, effective sequence of behaviors. This long sequence of behaviors produces considerable variety of behaviors, and this variety provides a chance to improve the total organizational performance. Furthermore, a long sequence of behaviors is needed for robots to evaluate their own behaviors. This is because robots cannot evaluate their behaviors with only one action when the total recent situational state cannot be obtained by communicating with a centralized system or broadcasting information among one another. In such cases, robots store a sequence of behavior rules and self-evaluate their behaviors when they acquire the recent situational state by themselves.

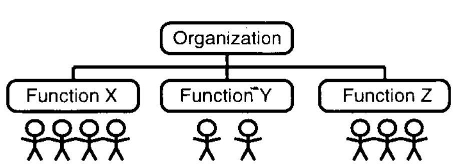

Fig. 1 Organizational structure

Organizational-learning-oriented classifier system

We now consider our organizational learning model. This is an organizational-learning-oriented classifier system (OCS) to address the problems of multiple robots for practical and engineering use. OCS is an extended multiagent version of the learning classifier system (LCS)[14,15] with the concept of organizational learning, and is based on an evolutionary approach. Unlike a conventional evolutionary approach, OCS does not evaluate populations with a global evaluation function such as a fitness function, but robots in OCS self-evaluate their own behaviors with their own local evaluation functions. Furthermore, by continuing this self-evaluation process, robots acquire their own appropriate functions for adaptive collective behaviors through local interactions among their neighbors, and form an organizational structure without explicit control mechanisms or communication methods. Although the Michigan approach and the Pittsburgh approach to a LCS are used, we modify both approaches by introducing the concept of organizational learning, as detailed below.

- As shown in Fig. 2, robots do not employ an explicit global evaluation function, but use their own local evaluation functions which are composed of four parts: (1) rule base composed of many rules (CFs: classifiers) in LCS; (2) working memory (WM) for storing information about the environmental state; (3) memory for storing the sequence of behavior rules; (4) evaluation mechanism to evaluate the sequence of behavior rules.
- Robots can obtain their own environmental state, but the total environmental state around the robots cannot be obtained.
- Robots do not have any type of communication function. Therefore, robots self-evaluate their behaviors with their own local evaluation functions, and modify or acquire their own functions to cooperate among themselves. When robots recognize that their functions contribute to completing a given task, the strength of the behavior rules is reinforced according to the method for task completion. On the other hand, when robots recognize that their functions lead to a deadlock situation, the strength is reinforced according to the method for deadlock situations.

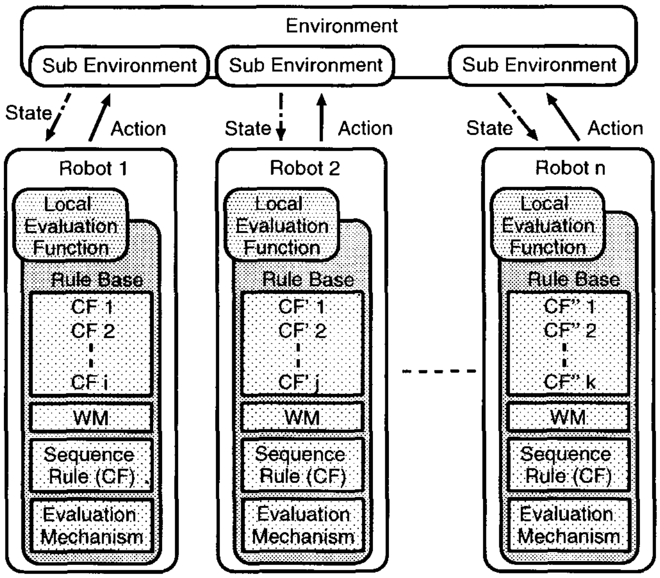

Fig. 2 Architecture of an OCS

In OCS, the following mechanisms and assumptions are also employed.

- A robot performs one behavior by selecting one rule, and the rule is selected according to the local environmental state using the roulette selection used in a standard LCS.
- At first, the strengths of all behavior rules are set to the same value.
- Since it is difficult to prepare effective and complex rules beforehand, OCS employs simple and rational rules and aims to generate effective and complex rules from these simple and rational ones.

Reinforcement learning for multiple robots

To make multiple robots cooperate with each other to complete given tasks without explicit control mechanisms or communication methods, the robots must have evaluation mechanisms to evaluate their own behaviors. To implement these mechanisms, we focus on reinforcement learning. Although many reinforcement learning methods for a single robot/agent have been proposed, we cannot apply these methods to multiple robot environments for the following reasons: (1) a discrete Markov process and the identification of the environment usually can no longer be assumed; (2) the landscape of the solution space usually becomes multimodal; (3) the tendency to get into a deadlock situation becomes higher as the number of robots increases.

To overcome these problems, we propose a novel reinforcement learning method for multiple robots, as shown in Fig. 3, and introduce this method to OCS as the evaluation mechanism shown in Fig. 2. In this reinforcement learning, the rewards are distributed to the behavior rules according to a profit sharing (PS) mechanism.[16] When the robots complete a given task, the method for task completion, as shown in Fig. 3a, distributes positive rewards for all correctly performed behavior rules in the same way as conventional PS. On the other hand, when the robots get into a deadlock situation, the method for deadlock situations, as shown in Fig. 3b, distributes negative rewards for the last selected behavior rule only, and distributes positive rewards for the remaining rules. This notion comes from our assumption that behaviors performed before getting into the deadlock situation contribute to the completion of tasks by cooperation between robots. In these cases, the strength of each behavior rule is calculated according to eqs. 1 and 2.

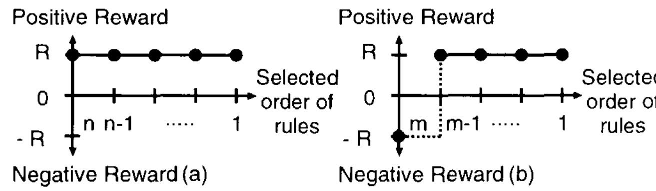

Fig. 3 Reinforcement learning for deadlock situations. (**a**) Task completion. (**b**) Deadlock situation. *R*, size of reward; *n*, *m*, selected order when robots acquire rewards

- *Task completion.*

$$ST(i) = ST(i) + R, \quad \text{where } i = n, n-1, \ldots, 1 \tag{1}$$

- *Deadlock situation.*

$$\begin{aligned} ST(i) &= ST(i) + R, \quad \text{where } i = m-1, \ldots, 1 \\ ST(i) &= ST(i) - R, \quad \text{where } i = m \end{aligned} \tag{2}$$

In Fig. 3, the vertical axis indicates the size of the rewards, and the horizontal axis indicates the selected order of rules. In eqs. 1 and 2, *ST* represents the strength of CF, *i* represents the selected order of CT, *n* and *m* represent the maximum number of selected CFs, and *R* represents the size of the rewards. In cases where *i* is small, CF(*i*) is selected within the first few selections.

Multiple robots in space

Example

To verify the effect of our organizational learning model OCS, we performed experiments on truss construction in space. These trusses are combined with beams and are the basis of other work such as the construction of a space station. To complete this task, robots must perform two main functions.

- *Cooperation in behavior order*: Robots must learn the order of selecting behaviors to cooperate with other robots. For example, two robots hold separate beams, and must set them in a desired location with a desired angle between them. Then a third robot welds the two beams.
- *Acquisition of different functions*: Robots must acquire different functions even if they are in the same situation. For example, robots acquire the following functions to

cooperate with other robots: holding a beam; welding beams; a combination of these two.

Robots

To reduce the weight of robots, each robot has one arm and holds either a beam or a welding tool. This limitation means that robots cannot weld beams and hold the beams at the same time, and also that cooperation among robots is required in truss construction tasks. However, robots are allowed to have their own welding tools for welding beams at any time.

In this truss construction task, some robots go to the beam construction location carrying their own beams, and wait at the beam construction location until other robots weld the beams. Alternatively, other robots go to a welding location holding welding tools, and weld beams at the welding location. In addition, robots at the station learn to decide whether they should hold their own beams or go to the welding location to weld beams. After beams are welded, robots with beams and robots with welding tools then learn to go to another welding location to weld the next beams or return to the station to get other beams. Continuing this cycle of learning processes, robots acquire appropriate functions. Furthermore, the following assumptions or limitations are taken into consideration in this task.

- Robots can move to set their own beams or to weld beams at a desired location, and can also rotate to set a beam at a desired angle. When robots move or rotate, they take the size of their body and the length of the beam into consideration so as not to bump into other robots.
- Robots which hold their own beams cannot release them until they are welded by the welding robots.
- Robots receive their own beams at the space station.
- Robots cannot pass their own beams to other robots.
- Robots cannot weld beams at the welding location until the beams are set at the desired location and at the desired angle.
- Robots do not know their functions beforehand.

Design of the centralized control system

To show the effectiveness of OCS, we designed a centralized control system to compare results. This centralized control system controls robots according to the values of WB and WW as follows. WB represents the number of robots which wait for beams, and WW represents the number of robots which wait to weld. Since robots in the centralized control system know the latest information by communicating with the system, they behave rationally and effectively.

- *Beam constructing location or welding location.*
  - $WB - WW < 1$: continue to perform the same behavior
  - $WB - WW \geq 1$: return to station
- *Station.*
  - $WB - WW < 1$: go to welding location
  - $WB - WW \geq 1$: hold a beam and go to beam construction location

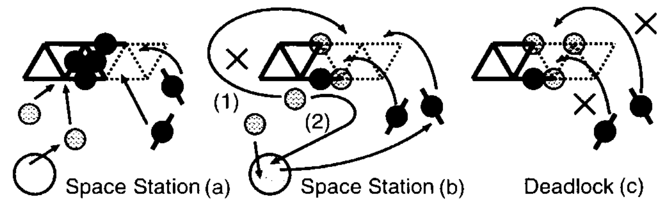

Fig. 4 Behaviors based on a centralized control system. (**a**) $WB - WW < 1$ ($WB = 0$, $WW = 4$). (**b**) $WB - WW = 1$ ($WB = 2$, $WW = 1$). (**c**) $WB - WW > 1$ ($WB = 3$, $WW = 1$). *Crile with bar*: robot which carries a beam; *shaded circle*: robot which welds beams or returns to station

In the case where $WB - WW < 1$ at the beam or welding location, as shown in Fig. 4a ($WB = 0$, $WW = 4$), robots continue to perform the same behaviors. When $WB - WW = 1$, as shown in Fig. 4b ($WB = 2$, $WW = 1$), robots which go to the welding location (Fig. 4b(1)) change their behavior to return to the station (Fig. 4b(2)) because beams alone are not enough to construct a truss. In the case where $WB - WW > 1$, since welding robots wait for beams at the beam construction location, as shown in Fig. 4c ($WB = 3$, $WW = 1$), robots with beams cannot go to the beam construction location. In this paper, we define this situation as deadlock. Furthermore, the situation where all robots hold their own beams or all hold welding tools is also defined as a deadlock situation. On the other hand, in the case where $WB - WW < 1$ at the station, robots go to the welding location because beams for welding are required. When $WB - WW \geq 1$, robots at the station hold the beams and go to the beam construction location because the beams are required to construct a truss.

Simulation

Experiments

We performed the following four experiments, and compared the steps needed to complete the given tasks. In these experiments, we count one step when all robots perform one behavior, and count one iteration when robots complete a task or get into a deadlock situation. In each case, robots construct a truss by combining 13 beams, and continue to repeat the same task to acquire more appropriate functions. R is defined as 1.

- *Case 1*: 5 robots with a centralized control system.
- *Case 2*: 5 robots with an OCS.
- *Case 3*: 10 robots with a centralized control system.
- *Case 4*: 10 robots with an OCS.

Results

Relationship between steps and iteration count

Figure 5 shows the relationship between the steps and the iteration counts with five or ten robots. In this figure, the horizontal axis indicates the iteration count and the vertical axis indicates the steps needed to complete the task. Using

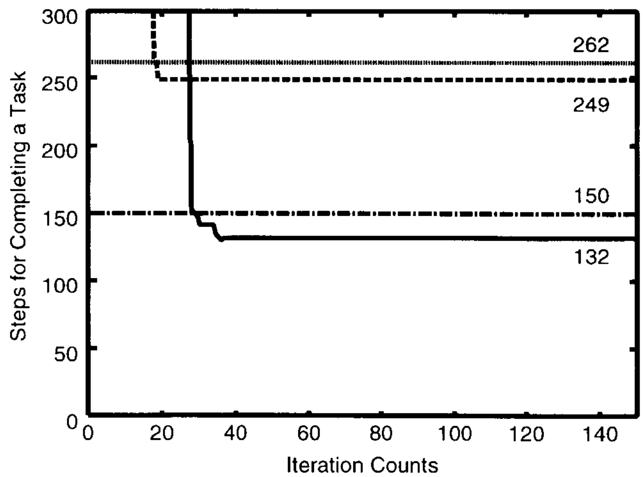

Fig. 5 Steps to complete a task. *Dotted line*, robot 5 with central control system; *dashed line*, robot 5 with OCS; *dashed and dotted line*, robot 10 with central control system; *solid line*, robot 10 with OCS

the centralized control system, which allows robots to get the latest information by communicating with the station, robots always complete the task rationally and effectively. However, since this system is designed in advance, the steps needed to complete the task do not decrease as the iteration count increases. As shown in Fig. 5, it takes 262 steps with five robots and 150 steps with ten robots to complete the task.

On the other hand, robots in an OCS cannot complete the given tasks in the first few iterations because they cannot get out of deadlock situations. However, robots can complete the task after more than 20 iterations with five robots, and after 35 iterations with ten robots. As shown in Fig. 5, it finally takes 249 steps with five robots and 132 steps with ten robots. From these results, the steps needed with either five or ten robots become fewer than those needed with a centralized control system.

Emergent function in an OCS

Using an OCS, many functions emerge according to the number of robots. As shown in Table 1, robots acquire two types of function with five robots and four types of function with ten robots. By comparing the functions of five and ten robots, it can be seen that robots acquire further complex functions as the number of robots increases. Since the number of functions with ten robots is a multiple of that with five robots, we might expect that ten robots form two organizations which are two groups of five robots. However, such kinds of organization are not formed. For example, the function composed of holding a beam and returning to the station for five robots is divided into the following types for ten robots: (1) a function which is the same as the function for five robots, and (2) a function which consists of hold a beam, return to the station, and weld the beam. Furthermore, although the number of the function which starts to weld beams at the beginning must be four with ten robots, this number decreases from four to one. This phenomenon indicates that three robots change their functions from welding beams to holding a beam to improve the total collective performance. These results indicate that the relationship between two organizations is not linear.

Table 1. Emergent functions

5 robots		10 robots	
Number	Function	Number	Function
3 robots	B → S	1 robot	W → S → W
2 robots	W → S	4 robots	B → S
		3 robots	B → W
		2 robots	B → S → W

B, go to beam constructing location; W, go to welding location; S, return to station

Findings and discussion

From the experimental results in these simulations of truss construction with multiple robots, the following findings were obtained.

— *Effectiveness of OCS*: In spite of not preparing appropriate behavior rules beforehand and not using the latest global information from the centralized control system, OCS enables robots to acquire their own appropriate functions through local interactions among their neighbors, and get out of deadlock situations without explicit control mechanisms or communication methods. Furthermore, robots with OCS complete given tasks by forming an organizational structure, and then improve their organizational performance. These results suggest that OCS is effective for the following problems: (1) any problem where it is difficult to design an appropriate global evaluation function; (2) any problem where it is difficult to prepare appropriate behavior rules in advance; (3) a domain where heuristic knowledge is not enough.

— *Effectiveness of reinforcement learning for multiple robots*: Since robots with OCS get out of a deadlock situation by forming a "rational" organizational structure, and complete given tasks in fewer steps than those with a centralized control system by forming an "effective" organizational structure, the proposed reinforcement learning method is effective for multiple-robot control. These results indicate that the combination of both positive and negative rewards explores the search space while maintaining an effective organizational structure. Therefore, the steps needed to complete given tasks are reduced without getting into a deadlock situation, even if the OCS reinforces the behavior rules from the viewpoint of whether the robots complete the given tasks or get into a deadlock situation. In general, the reinforcement learning method from this viewpoint contributes to reducing the iteration counts rather than the steps, but the proposed method enables robots to take fewer steps to complete tasks.

— *Emergent functions*: Because functions in an OCS emerge according to the number of robots, a large number of robots do not form the same organizations as are formed by a small number of robots, even if the number of robots in the two groups is linearly related. From these experimental results, an OCS enables robots to adapt to unexpected phenomena by acquiring appropriate functions.

— *Limitations of our reinforcement learning*: Since the proposed negative reinforcement learning only distributes negative rewards to the last selected behavior rule, this scheme is valid only for cases where deadlock situations can be prevented by only one action. This indicates that robots within this scheme cannot complete given tasks when the second last selected rule leads to a deadlock situation. Furthermore, this scheme only focuses on attempting to find fewer steps to complete given tasks, and ways of reducing the number of iterations are not considered. To overcome these issues, we have started to investigate the effect of other reinforcement learning methods which distribute the negative rewards to more than one rule, and which focus on reducing both steps and iterations. The effects of such cases are reported elsewhere.[17,18]

Conclusion

We have proposed an organizational learning model OCS, and have shown its effectiveness with task construction problems by comparing the number of steps taken with those needed with a centralized control system. An OCS enables robots not only to acquire their own appropriate functions through local interactions among their neighbors, but also to get out of deadlock situations without explicit control mechanisms or communication methods. Furthermore, robots with an OCS also complete given tasks by forming an organizational structure, and then improve their organizational performance. We have also found that reinforcement learning for multiple robots reduces the number of steps needed to complete given tasks compared to those needed with a centralized control mechanism. Future research includes:

— theoretical analysis of the proposed reinforcement leaning method for multiple robots;
— verification of the effectiveness of OCS in reconstruction not only when new robots are added, but also when a number of robots become inactive;
— investigation of the effectiveness of OCS with real robots.

References

1. Sugihara K, Suzuki I (1990) Distributed motion coordination of multiple mobile robots. IEEE International Symposium on Intelligent Control, pp 138–143
2. Mataric MJ (1994) Learning to behave socially. 3rd International Conference on Simulation of Adaptive Behavior (From Animals to Animats, 3), pp 453–462
3. Parker LE (1994) ALLIANCE: an architecture for fault tolerant, cooperative control of heterogeneous mobile robots. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94), pp 776–783
4. Asama K, Ozaki H, Itakura A, Matsumoto A, Ishida Y, Endo I (1991) Collision avoidance among multiple mobile robots based on rules and communication. IEEE International Workshop on Intelligent Robots and Systems '91, pp 1215–1220
5. Yuta S, Premvuti S (1992) Coordinating autonomous and centralized decision making to active cooperative behaviors between multiple mobile robots. IEEE International Workshop on Intelligent Robots and Systems '92, pp 1566–1575
6. Argyris C, Schon DA (1978) Organizational learning. Addison-Wesley, Reading
7. Duncan R, Weiss A (1979) Organizational learning: implications for organizational design. Res Organ Behav 1:75–123
8. Espejo R, Schuhmann W, Schwaninger M, Bilello U (1996) Organizational transformation and learning. Wiley, New York
9. March JG (1991) Exploration and exploitation in organizational learning. Organ Sci 2:71–87
10. Takadama K, Terano T (1997) Good solutions will emerge without a global objective function: applying organizational classifier system to printed circuit board design. IEEE International Conference on Systems, Man, and Cybernetics (SMC'97), pp 3355–3360
11. Takadama K, Nakasuka S, Terano T (1998) Printed circuit board design via organizational-learning agents. Applied intelligence: special issue on intelligent adaptive agents, in press
12. Takadama K, Hajiri K, Nomura T, Okada M, Shimohara K, Nakasuka S (1997) A computational group dialogue model with organizational learning. IEEE 1997 International Conference on Intelligent Processing Systems (ICIPS '97), pp 174–179
13. Takadama K, Hajiri K, Nomura T, Nakasuka S, Shimohara K (1998) Organizational knowledge on formation in multiple robots learning. 3nd International Symposium on Artificial Life and Robotics (AROB'98), pp 397–401
14. Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, Reading
15. Holland JH (1985) Properties of the bucket brigade algorithm. 1st International Conference on Genetic Algorithms (ICGA '85), pp 1–7
16. Grefenstette JJ (1988) Credit assignment in rule discovery systems based on genetic algorithms. Mach Learn 3:225–245
17. Takadama K, Hajiri K, Nomura T, Nakasuka S, Shimohara K (1998) Reinforcement learning for multiple robots with organizational learning. 3rd International Symposium on Artificial Life and Robotics (AROB'98), pp 392–396
18. Takadama K, Nakasuka S, Terano T (1998) Multiagent reinforcement learning with organizational-learning-oriented classifier system. IEEE 1998 International Conference on Evolutionary Computation (ICEC'98), pp 63–68