Interactive Evolutionary Computation for Real Robot from a Viewpoint of Observation

Daisuke Katagami CISS, IGSSE, Tokyo Institute of Technology
4259 Nagatsuta, Midori-ku Yokohama 226-8502, JAPAN
katagami@ymd.dis.titech.ac.jp
Seiji Yamada CISS, IGSSE, Tokyo Institute of Technology
4259 Nagatsuta, Midori-ku Yokohama 226-8502, JAPAN
yamada@ymd.dis.titech.ac.jp

Abstract. In this paper, we describe investigation on a viewpoint of observation in an interactive evolutionary robotics system. We propose a behavior learning system ICS (Interactive Classifier System) using interactive evolutionary computation and a mobile robot is able to quickly learn rules by direct teaching of an operator. Also ICS is a novel evolutionary robotics approach using an adaptive classifier system to environmental changes. We classify teaching methods as internal observation and external one, and investigate the relationship between the observation methods and the results. We have two experiments based on our teaching methods on a real world.

1 Introduction

In previous robot learning studies, optimization of control parameters has been applied to acquire suitable behaviors in an real environment. Also in most of such researches, a model of human evaluation has been used for validation of learned behaviors. However, since it is very difficult to build a human evaluation function and adjust control parameters, a system hardly learns behaviors intended by a human operator.

In contrast with modeling human evaluation analytically, we introduce another approach in which a system learns suitable behaviors using human direct evaluation without its modeling. Such an interactive method with *Evolutionary Computation* (EC) as a search algorithm is called Interactive EC (IEC) [1] , and a lot of researches on it have been done thus far[2] [3].

Additionally reinforcement learning has been applied to robot learning in a real environment[4]. Unfortunately the learning takes pretty much time to converge. Furthermore, when a robot hardly gets first reward because of no priori knowledge, the learning becomes far slower.

To solve these problems, we have been proposed the framework of Interactive Evolutionary Robotics (IER)[5][6]. It is an interactive EC learning method for the purpose of designing a robot using EC methods like *genetic algorithm*, *genetic programming* and *evolutionary strategy*. We can expect IER to perform high emergent property of ER and subjective adaptability of IEC. This method quickly learns effective rules by simple instructions of a human operator. The objective of IER is to make initial learning more efficient and learn behaviors that a human operator intended through interaction with him/her.

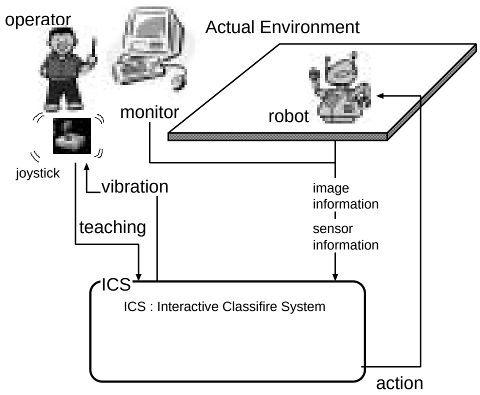

Figure 1: Teaching environment

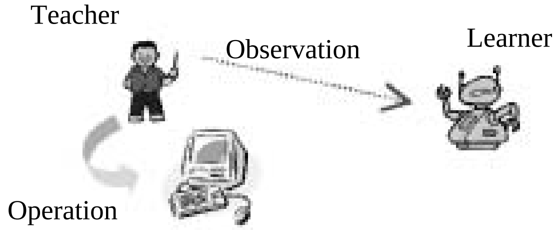

Figure 2: Teacher view

First of all, we developed a learning system based on *classifier system*[7] on IER framework, which is able to adapted to multiplicity of an environment and a variable dynamic state. we call it as *Interactive Classifier System* (ICS). Fig.1 give an environment of teaching using ICS.

The difference between ICS and an usual learning classifier system is to introduce an interactive method. Accordingly, we expect that the system performs efficient initial learning in an actual environment and can operate concentrative incremental learning. However there is few framework that an operator observes from robot's view. Therefore, the system can not make the best use of the learning. In this paper, we propose interactive method based on internal observation in order to solve the problem.

2 Teaching Method Based on the Viewpoint of Observation

A robot hardly knows how the robot modifies behaviors of itself for a task and recognizes that the task is in the process of achieving. Besides, a robot, which observes through an inside viewpoint of a system, may be unable to recognize that a task is realizing, or rather an operator, which observes through the outside viewpoint of a system, can recognize. We, then, had a significant difference, what is called a perceptual aliasing problem, between cognition of an operator and one of a robot. It hence become a subject of discussion when a operator teach skills for a task to a robot.

In this paper, we prepare the simple setting based on the observation to examine how the difference influence to acquired rules by teaching. We call the method which observe through an outside viewpoint of a system as *teacher view* (Fig.2), and the method which observe through the inside viewpoint of a system as *learner view* (Fig.3).

We examine the difference by IER based on teaching with this *teacher view* and *learner view*. To realize this IER, we applied these methods to developed ICS, which is a robot learning system based on interactive EC.

3 Interactive Classifier System

3.1 System overview

ICS applies XCS[8], which is a kind of Learning Classifier System (LCS), as evolutionary computation and equipped a interactive function. XCS equipped a function which preserve classifiers from overgeneralization makes system's performance worse. Moreover, XCS applies restricted mating which is a kind of strategy based on *genetic*

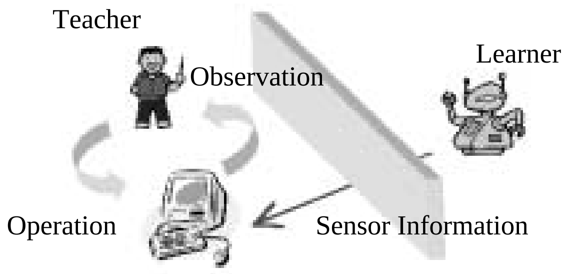

Figure 3: Learner view

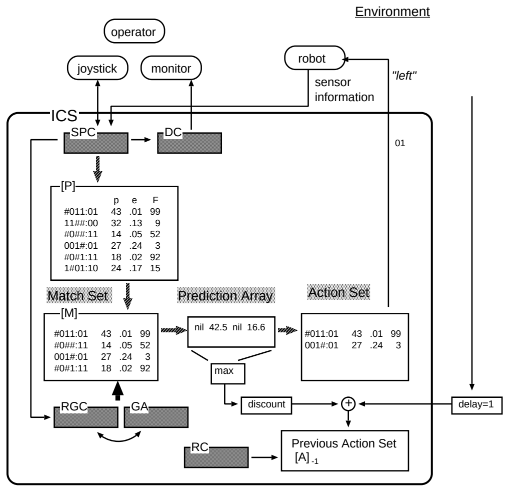

Figure 4: Overview of Interactive Classifier System

argorithms. For these reason, XCS improves learning performance of traditional CS. It is constructed as a robot learning model that can not only learn through teaching but also learn autonomously using XCS.

ICS mainly consist of a rule generation component (RGC), a sensor processing component (SPC), a display component (DC) and a reinforcement component (RC). It was developed with C language and GTK+ on Linux. It utilizes Video4Linux for image processing. The rule generation component makes a new classifier from teaching by the operator. The SPC processes each information of some sensors and camera, and through it for the RGC. The DC displays by GUI interface and processes the input from a joystick. Finally the RC performs learning by updating parameters in ICS. Fig.5 shows the developed interface of the system.

The experiments are made with a standard miniature mobile robot Khepera (Fig.6). The mobile robot has a cylinder shape, a diameter of 6 cm and a height of 5 cm. It possesses two motors and on-board power supply. The motors can be independently controlled by a PID controller. The eight infrared sensors are distributed around the robot in a circular pattern. They emit infrared light, receive the reflected light and measure distances in a short range: 2-5 cm. The robot is also equipped with a Motorola 68331 micro-controller which can be connected to a computer via serial cable. Moreover, the system utilizes SONY analog controller DUALSHOCK as a joystick. Fig.6 shows them respectively.

We describe a learning procedure in ICS as follows.

1. At first, a human operates robot with a joystick by viewing sensor information displayed on GUI, and the DC processes it.

2. Next, the SPC gets operator's instruction and robot's sensor information.

3. The RGC makes new rules from them and adds them into a rule list. When nothing is input from the operator, a mobile robot executes autonomous behaviors from interaction.

4. Finally, the RC reinforces the classifiers by updating their parameters in the actions which were previously executed.

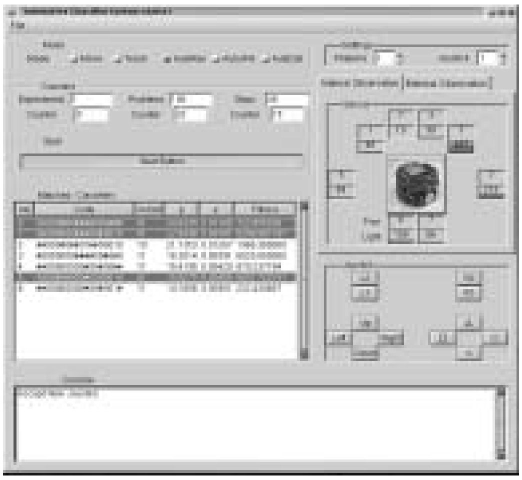

Figure 5: User interface

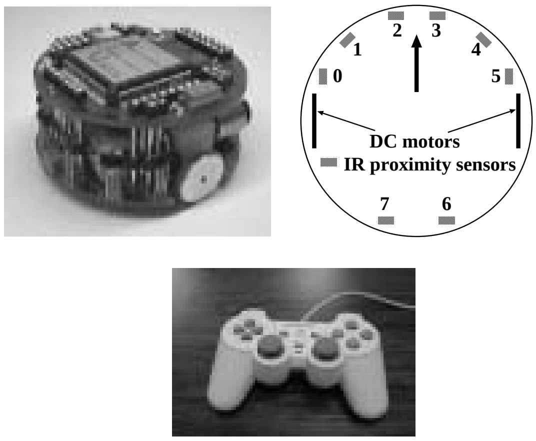

Figure 6: A mobile robot Khepera and a joystick

In traditional works of robot learning in a real environment, a learning takes pretty much time to converge because of learning by trial and error. In this work, we consider that this learning by trial and error is a problem in a real environment. However, it hardly prepare suitable apriori knowledge in an environment. For this reason, the ICS generate initial individuals by teaching from human-robot interaction. We can perform efficiently initial learning in this way.

4 Experiments

4.1 Experiment A with Cognitive Observation

We experimented in a real world to investigate difference in two teaching methods: *teacher view* and *learner view*. Fig.7 shows an experimental environment. As an experimental task ICS reduces the number of steps from any start points to a light source which set up as a goal in a field surrounded with white plastic plates. We compared two teaching methods with a traditional method in which a robot autonomously learns by simple EC.

An encoding of a classifier is as follows. A classifier is the twenty bit string "#000#10 0000###100#:01". The robot's condition is the left sixteen bit string, it represented eight infrared proximity (left eight bit) and light (right eight bit) sensors around the robot in a circular pattern respectively (Fig.6). The bit is "1" if a input sensor value larger than a threshold, or else "0". "#" is a "don't care" symbol which *classifier system* employed. The robot's previous action is the next two bit string, represented as forward "11", left-turn "01", right-turn "10 " and back "00". The robot's current action is the two bit string similarly.

A fitness function of ICS defined as follows. Reward F is computed by the sum of eight light sensors through a sigmoid function.

$$u = \sum_{i=0}^{7} light_i \frac{PaymentRange}{light_{MAX} \times 8} \qquad F = \frac{1}{1 + exp(-u)}$$

We consider fourty steps as a trial and begin teaching or autonomous exploration for five trials at randomly start points, and test a trial at each of five start points

Table 1: Experimental Parameters

Parameters	Value
number of problems in one experiment	30
number of expriments	1
maximum size of the population	300
probability to do crossover	0.8
probability of mutating one bit	0.04

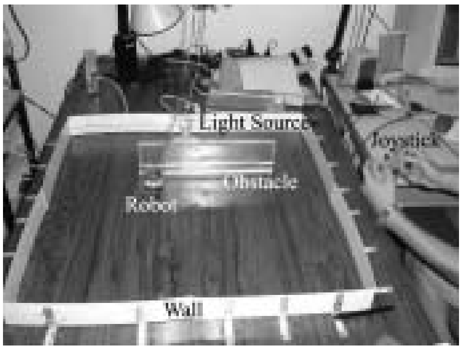

Figure 7: Experimental Environment

Figure 8: External Observation Settings

as examination of created rules by the teachings or the exploration. It performs this procedure six times, and consequently, we have thirty trials as examination. Table 1 shows experimental parameters.

In the case of *teacher view*, the operator can not recognize a small obstacles as the robot can perceive although he looks a whole environment. To represent *teacher view*, an operator teach to a robot using the camera which looks a whole environment. Fig.8 shows information through GUI by a camera which sets up the environment.

In the case of *learner view*, an operator can not look a whole environment although it can recognize directly a small obstacles and a recognition error which the robot can perceive. To implement *learner view*, ICS uses GUI which represent some sensor values. Since an operator hardly understands although one looks only numerical values of sensor data, we developed GUI interface which can represent sensor values by graphs. Fig.9 shows sensor information which represented by graphs through GUI interface.

4.2 Experimental Results of Examination A with Cognitive Observation

In this experiments, we examined the number of steps to a light source, system error, and fitness. Fig.10 shows the average of the number of steps at test trials from five start points. System error(Fig.11) is the absolute difference between the system prediction for the chosen action by a system and the actual external payoff. Fitness (Fig.12) which is used in *genetic algorithms* for evaluation.

Teacher view improves a robot's learning in the simple environment which cognitive difference is little by teaching is easy because an operator looks a whole environment. However, the difference is not so large. There was no difference in both observations regard for system error and fitness. We found out that the learning improves without they effect on system error or fitness by using two kind of teaching methods.

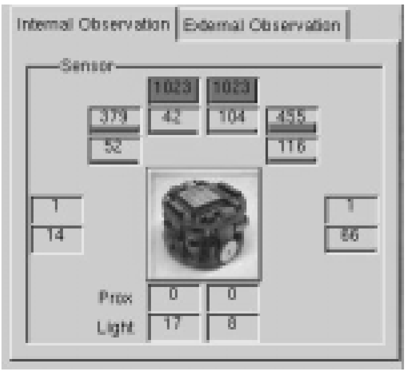

Figure 9: Internal Observation Settings

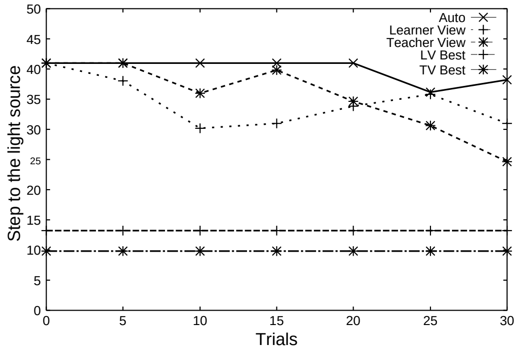

Figure 10: Step to Light Source

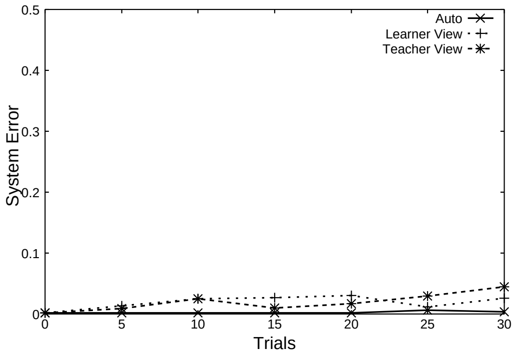

Figure 11: System Error

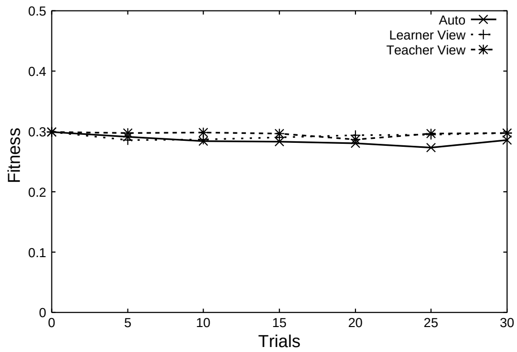

Figure 12: Fitness

4.3 Experiment B with Cognitive Observation

We introduced an obstacle in the environment of experiment A to investigate the effect by a more difficult task. Fig.7 shows the experimental environment. The obstacle was made of the transparent plastic board because of perception a direction of a light source. A robot must reach to the light source as avoiding the obstacle. We compared a teaching method by *teacher view* with *learner view* in the same way as experiment A.

We consider twenty steps as a trial and begin teaching for twenty trials at randomly start points. It is different from experiment A because of simplifying an experiment. We test a single trial at a start point every one trial as examination of created rules. The experimental parameters is as same as Table 1.

4.4 Experimental Results of Examination B with Cognitive Observation

In experiment B, we examined the number of steps to a light source. The number of steps to a light source shows Fig.13.

Seeing from a best value of teaching (TV Best and LV Best), we see that *teacher view* improves teaching likewise experiment A because of looking a whole environment. However, we can see that *learner view* outperformed *teacher view* about steps to a light source in contrast with experiment A. Since ICS hardly uses information of proximity sensors and can acquire effective rules by using only information of light sensors in the environment, the difference of cognition between an operator and a robot is a little. For this reason, the difference of two teaching methods was not so large.

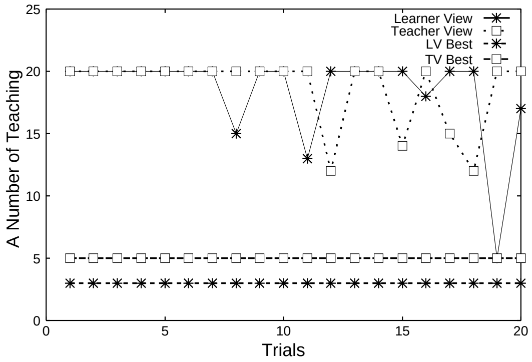

Figure 13: Step to Light Source

Table 2: Experimental Results of Exploit

	Step to Light Source	Reach to Goal
times	1 2 3 4 5	total
teacher view	- - - - -	0
learner view	6 9 - - -	2

Table 3: Created Rules by *learner view* Method

Condition	Action	Prediction	Teach
0#0#0#####0#10##1#	10	413.7	2
0010###0##000#1###	01	364.0	1
00###0#000#1#0#01#	11	292.0	4
###000000101001011	11	256.0	1
0#0#0#####0#10#01#	10	280.3	3
00#000001001##0##1	10	244.5	1
00###0#000#1#0##11	11	215.0	1
#000#100000###100#	01	101.5	3
#000##001000##0001	11	100.8	6
#000#0001000##0001	11	99.4	10

ICS however hardly learns from teaching of an operator in case of experiment B because there is the difference between the situation of the robot which an operator estimated by observation from the outside and the one of the robot in the real world. *Leaner view* actually creates effective rules, because an operator performed teaching as verifying robot's internal status.

After we have experiments *teacher view* and *learner view* each twenty trials, we test a trial at each of five start points as examination using each created rules. Table 2 shows the experimental results. The robot can not reach to a light source at any start points of five in the examination of *teacher view* because teaching does not improve. On the other hand, the robot reached to a light source by six steps and nine steps respectively at two of five start points in the examination of *learner view*. We can see that ICS can create rules which effective and do not depend on start points since *learner view* improves the robot's learning from teaching.

Table 3 shows ten rules which have best values of the system prediction in created ones after twenty trials with *learner view*. ICS created effective and common-sense rules like as the robot moves forward when the light source is in front of the robot, and it moves right when the light source is right. We can see its teaching improves very well because all the rules, which the system prediction is high, is created from teaching or its offspring.

Though an operator did not teach a robot to go back in the both experiments, the robot goes back for avoiding an obstacle when the robot colides with a wall and reaches to a light source. Because ICS created these rules by which a robot works in cooperation with human.

5 Conclusion

In this paper, we proposed a novel interactive method from the viewpoint of observation. We classified the teaching from the viewpoint of observation as external observation and internal one, and investigated its effects in a real world experiments. It was found that internal observation increase the effects according to arise the difference between the robot's recognition and operator's one in a complex environment.

This study showed the difference of a teaching effect due to the difference of observation methods and examined that it is possible to perform a fast learning of a robot by human-robot interaction in a real world. Besides, our system learns and automatically acquires a complex robot's program which human hardly describes by the way that an operator only teachs to a robot simply. Our future direction will be one that acquires the unconscious information of an operator like reactive behaviors of human and skills of an operator.

References

[1] R. Dawkins. *The Blind Watchmaker*. Longman, Essex, 1986.

[2] Y. Nakanishi. Capturing Preference into a Function Using Interactions with a Manual Evolutionary Design Aid System. Genetic Programming, pages 133–140, 1996.

[3] T. Unemi. A Design of Multi-Field User Interface for Sumulated Brdding. Asian Fuzzy System Symposium (AFSS'98), pages 489–494, 1998.

[4] E. Uchibe, M. Asada, and K. Hosoda. Behavior coordination for a mobile robot using modular reinforcement learning. In IEEE/RSJ International Conference on Intelligent Robots and Systems 1996 (IROS96), pages 1329–1336, 1996.

[5] D. Katagami and S. Yamada. Interactive classifier system for real robot learning. In *IEEE International Wortkshop on Robot and Human Interaction(ROMAN-2000)*, pages 258–263, 2000.

[6] D. Katagami and S. Yamada. Real robot learning with human teaching. In *The Fourth Japan-Australia Joint Workshop on Intelligent and Evolutionary Systems*, pages 263–270, 2000.

[7] J. H. Holland and J. S. Reitman. Cognitive systems based on adaptive algorithms. In Donald A. Waterman and Frederick Hayes-Roth, editors, *Pattern-Directed Inference Systems*, pages 313–329, Orlando, 1978. Academic Press.

[8] S.W. Wilson. Classifier fitness based on accuracy. Evolutionary Computation, 3(2):149–175, 1995.