2005 IEEE International Conference on Systems, Man and Cybernetics
Waikoloa, Hawaii October 10-12, 2005

Exploiting Interaction between Sensory Morphology and Learning

Komei Sugiura[*†], **Makoto Akahane**[*], **Takayuki Shiose**[*†], **Katsunori Shimohara**[*†] **and Osamu Katai**[*]
[*]Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan
{sugiura, akahane, shiose, katai}@sys.i.kyoto-u.ac.jp
[†]ATR Network Informatics Laboratories
2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288, Japan
katsu@atr.jp

Abstract - *This paper proposes a system that automatically designs the sensory morphology of an autonomous robot. This system uses two kinds of adaptation, ontogenetic adaptation and phylogenetic adaptation, to optimize the sensory morphology of the robot. In ontogenetic adaptation, individuals with many different sensory morphologies use reinforcement learning to adapt to a task. In phylogenetic adaptation, a Genetic Algorithm is used to select morphologies with which the robot can learn the task faster. We made the system design a line-following robot, and carried out experiments to compare the design solution with a hand-coded design. The results have shown that the designed robot outperforms the hand-coded design in terms of line-following accuracy and learning speed, although it has fewer sensors than hand-coded robots. The paper also shows the effective use of sensory morphology obtained by our system.*

Keywords: sensor evolution, ecological balance, embodiment, learning and evolution.

1 Introduction

About 100 years ago, Uexküll pointed out the close relationship of the perception and action of animals [1]. He argued that animal species, including humans, do not share an objective world, and that their behavior is based on their subjective perceptional world (Umwelten). Therefore, to comprehend animals' behavior, it is important to consider sensors that create a perceptional world. Nature has created various kind of sensors, which are closely related to the effectors, neural systems, and ecological niche of animals [2].

On the other hand, the sensors, actuators and controllers of robots are separately designed in most research [3]. For example, it has been important for the designers of sensory morphology to design the morphology with which they can easily compute physical quantities. Thus, the relationship between the sensory morphology and the controller is often ignored.

However, to construct an autonomous and adaptive robot, the designer needs to consider the interaction between the morphology, controller, and environment [4]. Pfeifer calls this design principle "ecological balance". Some studies are aimed at constructing adaptive robots by exploiting the aforementioned interaction. Sims and Lipson *et al.* developed systems that evolve the morphology and neural controller of robots [5], [6]. In recent years, work has began on evolution of sensors, which aims at studying the interaction between the sensory morphology and controller [7]–[9].

In these studies, the morphology and controller evolve along the same time scale. Therefore, it is not clear that the designed morphology is effective for ontogenetic adaptation (learning). In addition, it is important that the controller be able to adapt after the robot is physically manufactured. On the other hand, Jung investigated the evolutionary layout of state space (sensors) according to the results of learning [10]. No previous research, however, has realized a system that automatically designs morphology effective for learning.

In this paper, we propose a system that automatically designs the sensory morphology of an adaptive robot. This system designs the sensory morphology with two kinds of adaptation, ontogenetic adaptation and phylogenetic adaptation, to optimize the learning ability of the robot. The actuators of the robot are fixed during the design process. This minimizes the cost in terms of computation and mufacturing hardware. Thus we aim at the bottom-up construction of an interface for the "physical world" and an "informational world".

2 The System

2.1 Designing Morphology to Improve Learning Ability

To design a robot with better learning ability, a designer may adopt a trial-and-error method in hardware. In other words, he repeats experiment-adjustment-redesign cycles. However, this method has several problems, such as the cost in time, the cost of redesigning electrical circuits, biased design by humans, etc.

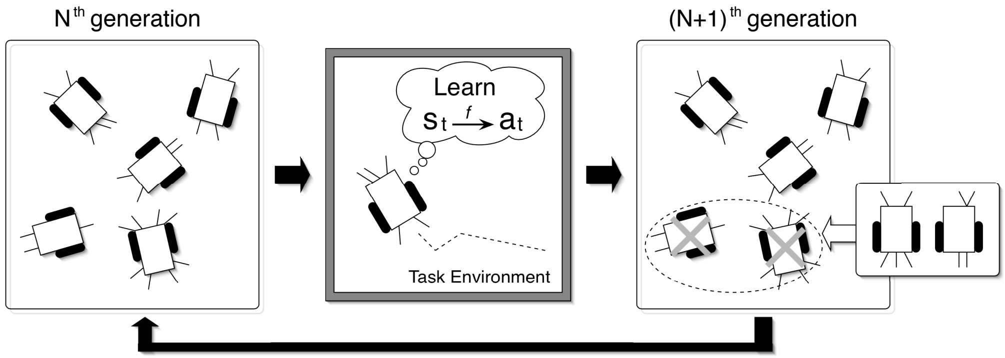

Figure 1. Design flow: ontogenetic adaptation and phylogenetic adaptation

On the other hand, the system we have constructed in this research designs the robot's morphology in software. Our system evaluates a robot based on whether it succeeds in learning certain tasks, and then redesigns its morphology. To optimize the sensory morphology, two kinds of adaptation are used: ontogenetic adaptation and phylogenetic adaptation. In ontogenetic adaptation, individuals with many different sensory morphologies use reinforcement learning to adapt to the task. In phylogenetic adaptation, a Genetic Algorithm (GA) is used to select the morphologies with which the robot can learn the task faster.

Figure 1 shows an outline of the design flow. After a design solution is obtained in simulation, we manufacture a physical robot according to the design. The whole process of design is performed in the robot simulator Webots [11].

2.2 Task Environment

Line following is adopted as a task to be performed by robots. The task consists of following a line drawn on the floor. Figure 2 shows the course (200 cm $\times$ 500 cm) used in the experiments. The course has several checkpoints ($\mathrm{p}_1, \mathrm{p}_2, \mathrm{p}_3$), an S-shaped curve, and a perpendicular corner.

We chose this task environment for the following two reasons. First, changes in the number and layout of sensors strongly affect the meaning of the sensory signals. For example, even though the states observed by two robots with different sensory morphology are the same, they might be in a different situation from an observer's viewpoint. Second, many contests exist for line-following robots. Therefore, we can easily obtain a hand-coded design and compare it with our method.

2.3 Line Following-Robot

We applied our system to the line-following robot shown in Figure 3. The robot is based on the Micom Car[1], and has floor sensors to detect the color of the floor. Sensor values are binarized and used by the robot's controller for learning. Namely, the controller learns how to map states (sensor space) to actions.

In our system, the number and layout of the floor sensors are designed by using an evolutionary method. To evolve the number and layout of the sensors, a simple GA is used. The layout and number of sensors are coded in the genotypes of GA individuals. Each allele of the genotype is 0 or 1, which means whether a sensor is placed at a certain position or not. We made the genome length 32 bits, therefore each individual can have up to 32 sensors. For simplicity, the possible positions are digitized like grid points.

2.4 Ontogenetic Adaptation

In the ontogenetic time scale, the robot uses Q-learning to adapt to the task. Here, the state space is derived from sensor values. Therefore, if the robot has many sensors, the state space will become very large. This brings about both an advantage in accuracy and a disadvantage in learning speed. In contrast, a robot with few sensors has a disadvantage in accuracy but an advantage in learning speed.

In this time scale, a robot is given 100 trials (episodes). The robot is allowed to perform up to 2000 steps at each trial. At each time step t, it observes the current state s_t and outputs an action a_t. Five different actions are possible, allowing the robot to go straight, move left slightly, move left, move right slightly, or move right. The action is drived via ϵ-greedy action selection. The robot can attain rewards r_t according to the distance from the center of the line at every step. We employ the standard update rule:

$$\begin{aligned} Q(s_t, a_t) \leftarrow & Q(s_t, a_t) \\ & + \alpha\big[r_{t+1} + \gamma \max_a Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\big], \end{aligned}$$

where the learning rate $\alpha = 0.8$ and discount factor $\gamma = 0.999$.

The faster the robot becomes at arriving at the goal due to its ontogenetic adaptation, the greater the fitness value it obtains for phylogenetic adaptation.

[1] `http://www.bun-net.co.jp`

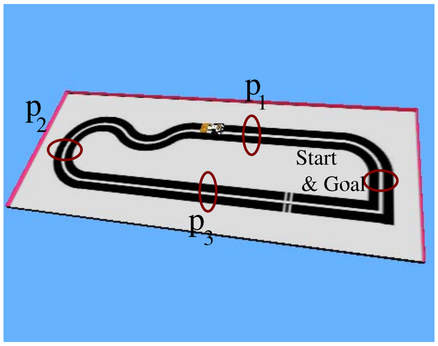

Figure 2. Task environment

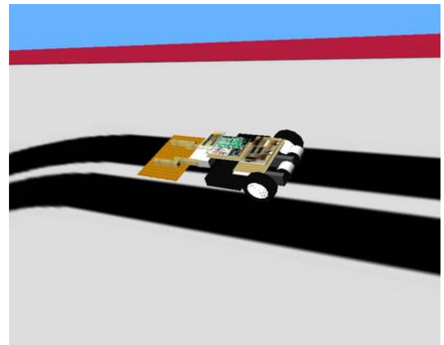

Figure 3. Line-following robot

2.5 Phylogenetic Adaptation

The system uses a simple GA to evolve the number and layout of the sensors. The outline of the design flow (*phylogenetic adaptation*) is as follows.

(1) The initial population is randomly generated.
(2) Ontogenetic adaptation at the n^{th} generation: each individual uses reinforcement learning to learn how to map states to actions.
(3) All individuals are evaluated on time-to-arrive-at-goal, and parent individuals are selected.
(4) The $n+1^{th}$ population is generated from parent individuals.
(5) Repeat (2) to (4) until a termination condition is met.

The fitness value of the genotype is calculated from the time needed to arrive at the goal. More specifically, fitness value Φ is the average of ϕ_i:

$$\phi_i = 1 - \frac{t_i}{T_{max}},$$

where t_i is the time to arrive at the goal in the i^{th} episode, and T_{max} (=2000) is the maximum time steps performed in one episode. If the robot is not able to arrive at the goal, ϕ_i should become zero. However, for practical reason, we give bonus points according to the distance that the robot has traveled. For example, if the robot arrives as far as the j^{th} checkpoint, ϕ_i will be calculated as follows.

$$\phi_i = \frac{j}{4}\left\{1 - \frac{t'_i}{T_{max}}\right\}$$

where t'_i stands for the time needed to arrive at the j^{th} checkpoint.

We use a simple GA with one-point crossover and point mutation as genetic operations. The parent individuals of the n^{th} generation are selected by tournament selection. We conducted experiments under the condition that the size of the population is 50, an the rates of crossover and mutation are 1.0 and 0.03.

3 Experiment 1

3.1 Experimental Setup

This section describes two kinds of experiments which we have carried out using the proposed system. A line-following robot is designed and evaluated in simulation. The outline of the experiments is as follows.

- Experiment 1-A: The system is applied to a line-following robot.
- Experiment 1-B: The obtained design solution is compared with a hand-coded design. To investigate learning ability, both the designed robot and the hand-coded robot learn the same task in simulation. The morphologies of all robots are fixed.

3.2 Results 1-A: Applying the System

First of all, we quantitatively examine the design solution of the proposed system. In Figure 4, the maximum and average fitness of individuals is plotted against the number of generations. The figure also shows the number of sensors possessed by the best individual in each generation. In the figure, the solid, broken, and dotted lines show the maximum fitness (best), the average fitness, and the number of sensors, respectively. Note that the averaged results of 10 experiments are shown in the figure.

Figure 4 shows that the maximum fitness increases with a decrease in the number of sensors. Specifically, the fitness is approximately 0.2 for 7 sensors at the fifth generation, and increases to 0.4 for 5 sensors at the 50th generation. This result indicates that robots with learning controllers need only five sensors to solve this task.

Figure 5 shows examples of the sensory morphology obtained in the experiments. Each figure shows one of the robots with four sensors (left), five sensors (middle) and four sensors (right). In Figure 5, black dots stand for sensors.

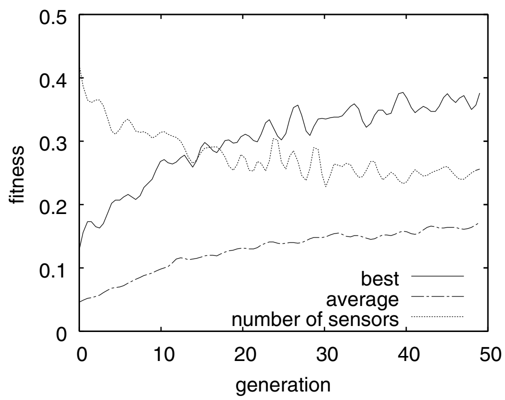

Figure 4. Variation of fitness and the number of sensors

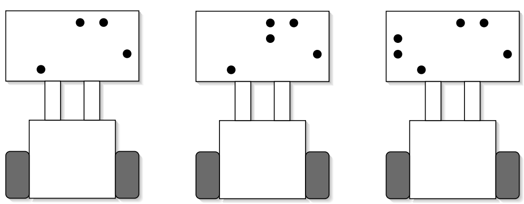

Figure 5. Examples of automatically designed sensory morphology: four sensors (left), five sensors (middle), six sensors (right)

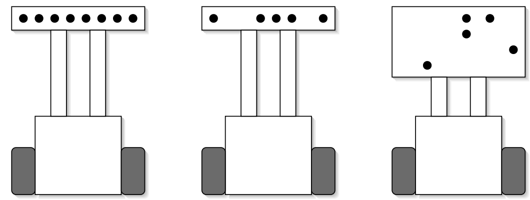

Figure 6. Robots used in Experiment 2: M1 (left), M2 (middle), M3 (right)

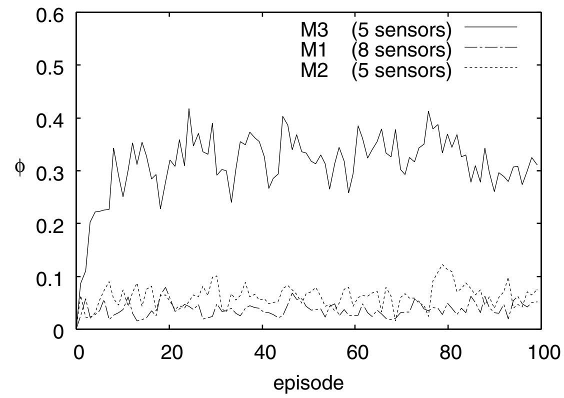

Figure 7. Variation of time to arrive at goal

3.3 Results 1-B: Comparing with Hand-Coded Design

We compared the learning ability of the designed robot with a hand-coded robot. The hand-coded robot *M1* is shown on the left side figure of Figure 6. M1 is a standard Micom Car, which has horizontally aligned sensors. The robot *M3* shown on the right side figure of Figure 6 was designed by our system. The robot *M2* is a robot whose state space is as large as M3's. Therefore, we can directly compare M3 with M2 with regard to the effect of the sensor layout.

In Figure 7, the variation of ϕ, which means how fast the robot arrives at the goal, is plotted against the number of episodes. In the figure, the broken, dotted, and solid lines show the results of M1, M2, and M3, respectively. The figure shows the averaged results of 10 experiments. Figure 7 shows that M3 is the fastest robot in every episode. To support this result, we conducted experiments in which the number of episodes was increased, and obtained similar results. This result means that M3 outperforms hand-coded robots in both line-following accuracy and learning speed.

3.4 Discussion

In this section, we discuss the experimental results in terms of the layout and number of sensors.

The sensor layout of the designed robots is asymmetrical (see Figure 5). This result can be explained by the fact that the course has more left curves than right curves. Thus, the phylogenetic adaptation prefers a morphology that can detect left curves. Figure 5 also shows that the sensors are scattered vertically. This is due to the fact that vertically scattered sensors enable robots to recognize the anteroposterior relationship of the line. On the other hand, a robot with horizontally aligned sensors cannot distinguish whether it is on a straight line or on a curve.

The number of sensors dominates the dimension of state space, thus it affects the learning speed and performance as a line follower. In other words, if we see reinforcement learning as function approximation, there is a trade-off between convergence speed and accuracy after convergence. However, our results showed that M3 outperforms M1 and M2 in terms of both line-following accuracy and learning speed. This result indicates that our system can design adaptive robots by exploiting the interaction between sensory morphology and learning.

4 Experiment 2

4.1 Experimental Setup

We carried out experiments to compare the hand-coded design and the design obtained by our system. Three kinds of physical robots *M1, M2, M3* were manufactured. These robots correspond to the left-hand, middle, and right-hand figures of Figure 6, respectively. Table 1 compares these robots in terms of sensory morphology and controller.

M1 has the standard sensory morphology and controller contained in a Micom Car kit. The sensory morphology of M2 is the same as M1, while its controller is obtained

Table 1. Comparison of manufactured robots

Micom Car	Sensory morphology	Controller
M1	Hand-coded	Hand-coded
M2	Hand-coded	Learned in simulation
M3	Evolved	Learned in simulation

Table 2. Experimental results: average of five trials

Micom Car	Lap time [sec]	Number of laps in succession
M1	15.6	4.4
M2	16.0	3.0
M3	13.5	8.4

in simulation. More specifically, the Q-table obtained in simulation is implemented in M2's controller. Note that the Q-values are fixed during the trials. The design solution of our system is implemented in M3. M3's controller is obtained in simulation as is M2's.

Each robot is given five trials. Once the trial has started, it continues until the robot finishes ten rounds or slides off the course. To investigate performance, the average lap time is compared. In a physical robot, the torque of a motor decreases as battery is consumed. Thus, the longer a physical robot runs, the harder it it to keep running along the course by a single controller. Therefore, we also compare these robots in terms of the number of laps in succession to investigate their robustness.

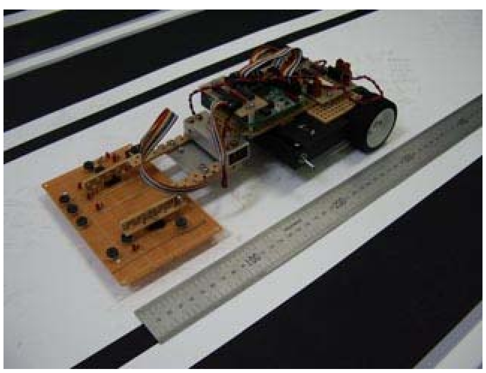

Figure 8. Manufactured line-following robot *M3*

4.2 Results and Discussion

The experimental results are shown in Table 2. Table 2 shows that M3 is the best among the three in terms of average lap time. In addition, the average number of laps in succession for M3 is 8.4, however those of M1 and M2 are 4.4 and 3.0. These results mean that M3 can follow the line more accurately, while M1 and M2 are apt to slide off the track.

Thus we obtained two advantages in the robot manufactured according to the design solution of our system: improvements in average lap time and the number of laps in succession.

These results clearly show that our system can design an adaptive robot that outperforms a hand-coded design in terms of learning.

Finally, we describe an example of emergent behavior from interaction between sensory morphology and learning. We show how M1, M2 and M3 run through a perpendicular corner, which is the most difficult point in the course.

M1 uses the crossline (see Figure 2) before the corner as a cue, and switches its controller to the corner mode. Namely, the designer's knowledge that "a perpendicular corner exists behind the cross line" is used.

On the other hand, because M3 is not given such knowledge, it must run through the whole track using a single controller. Figure 9 shows M3 coming to the corner. First, the black line activates sensors #3 and #4. This state is the same as being on a left curve, therefore M3 moves left slightly. This action, however, is not enough to turn left in this case. Then, sensor #4 detects the white line, and this leads M3 to move left again. Thus M3 can turn left.

However, M2 is apt to fail turning left at the corner. This is due to the fact that the state space constructed based on horizontally aligned sensors cannot detect whether the robot is coming to a corner or not. Also, it is difficult for M2 to learn the desired action mapped from the states.

In contrast, the sensory morphology of M3 is effective for detection and learning, and leads to the smooth learning of adaptive behavior. This result also supports the fact that our system can design adaptive robots by exploiting interaction between sensory morphology and learning.

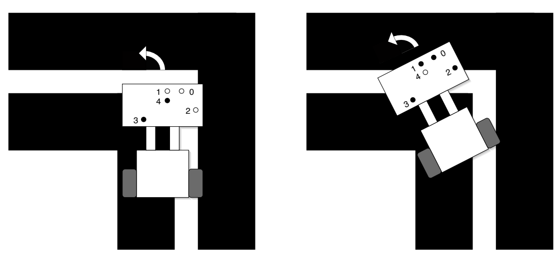

Figure 9. Effective use of morphology at a perpendicular corner

5 Conclusion

In this paper, we proposed a system that automatically designs the sensory morphology of an autonomous robot. This system optimizes the sensory morphology to allow the robot to learn the task faster. Two kinds of adaptation, ontogenetic adaptation and phylogenetic adaptation, are used in our system. In ontogenetic adaptation, individuals with many different sensory morphologies use reinforcement learning to adapt to the task. Phylogenetic adaptation selects morphologies according to the results of learning.

We carried out experiments to make the system design a line-following robot, and compared the design solution with a hand-coded design. The results show that the designed robot outperformed the hand-coded design in line-following accuracy and learning speed, although it has fewer sensors than the hand-coded robots. Then we manufactured the design solution as a physical robot. The robot also outperformed hand-coded design, using its morphology effectively. These results indicate that our system can design adaptive robots by exploiting interaction between sensory morphology and learning.

Acknowledgment

This research was supported in part by the National Institute of Information and Communications Technology.

References

[1] J. von Uexküll, "A stroll through the worlds of animals and men," in *Instinctive Behavior: The Development of a Modern Concept*, C. H. Shiller, Ed. International University Press, 1957, pp. 5–80.

[2] D. B. Dusenbery, *Sensory ecology: how organisms acquire and respond to information*. W. H. Freeman & Co., 1992.

[3] K. Hosoda, "What morphology brings to learning, what learning brings to morphology," *JRSJ*, vol. 22, no. 2, pp. 186–189, 2004, (in Japanese).

[4] R. Pfeifer and C. Scheier, *Understanding Intelligence*. Cambridge, MA.: MIT Press, 1999.

[5] K. Sims, "Evolving 3D morphology and behavior by competition," in *Artificial Life IV: Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*, R. Brooks and P. Maes, Eds. Cambridge, MA.: MIT Press, 1994, pp. 28–39.

[6] H. Lipson and J. B. Pollack, "Automatic design and manufacture of robotic lifeforms," *Nature*, vol. 406, no. 6799, pp. 974–978, 2000.

[7] M. Bugajska and A. Schultz, "Coevolution of form and function in the design of micro air vehicles," in *4th NASA / DoD Workshop on Evolvable Hardware (EH 2002)*. IEEE Computer Society, 2002, pp. 154–166.

[8] K. Dautenhahn, D. Polani, and T. Uthmann, "Special issue on sensor evolution," *Artificial Life*, vol. 7, no. 2, pp. 95–98, 2001.

[9] K. Balakrishnan and V. Honavar, "On sensor evolution in robotics," in *Proceedings of the First International Conference on Genetic Programming*, Stanford University, CA., 1996, pp. 455–460.

[10] T. Jung, P. Dauscher, and T. Uthmann, "Evolution and learning: Evolving sensors in a simple mdp environment," *Adaptive Behavior*, vol. 11, no. 3, pp. 159–177, 2003.

[11] Webots, "http://www.cyberbotics.com," commercial Mobile Robot Simulation Software. [Online]. Available: http://www.cyberbotics.com