A New Design Principle for an Autonomous Robot

Itsuki Doi[1], Takashi Ikegami[1], Atsushi Masumori[1], Hiroki Kojima[1], Kohei Ogawa[2,3], and Hiroshi Ishiguro[2,3]

[1]University of Tokyo, Japan
[2]University of Osaka, Japan
[3]Japan Science and Technology Agency Exploratory Research for Advanced Technology
idoi@sacral.c.u-tokyo.ac.jp

Abstract

Based on the principle of artificial life, we developed an upper-body android called "Alter." Alter is human-like in appearance, receives sensory information from the outside via an autonomous sensor system located around the android, and moves spontaneously using two autonomous systems of internal dynamics. Its body and arms contain a central pattern generator with seven degrees of freedom and hundreds of plastic artificial neurons. We investigated Alter's environmental adaptability and the spontaneity of is behavioral patterns. In addition, we discuss the conditions under which a robot can become lifelike.

Introduction

Autonomous robots have been a major research topic in the field of artificial life from the beginning. New design principles of artificial life have been developed. For example, Brooks' subsumption architecture (Brooks (1991)) was adopted to make behavior-based robots. A recurrent neural network as a robot controller added another dimension in the design principle. Complex behavior is synthesized through the interplay between a robots memory organization and environmental information. However, a global utility function is still required to ensure the desired behaviors. For example, we evolve neural architectures according to a genetic algorithm. Without using a utility function such as this, we cannot expect the emergence of the desired behavior or other meaningful behaviors. In this sense, the construction of an autonomous robot does not take an entirely bottom-up approach.

In this paper, we report our preliminary results in designing an autonomous humanoid robot that uses special learning dynamics called *learning by stimulus avoidance* (LSA). We recently developed this idea using plastic neural networks. While this learning method is based on the Hebbian learning algorithm, its input and output are tightly correlated. That is, the networks output should affect the subsequent inputs. LSA is the first implementation of the biological neural net developed in the experiment by Marom and Shahaf (Shahaf and Marom (2001)). They found that a biological neural net could learn a simple task by turning off the external stimulus at the right time. We reproduced the behavior with an artificial neural net (Sinapayen et al. (2017)) and applied it to a humanoid architecture.

The humanoid robot presented in this study is different from a "normal" humanoid. It can learn from the environment, yet its behavior is very disorganized. In other words, this humanoid, named Alter, has no global utility function to optimize; however, it gradually produces meaningful behaviors according to the principles of LSA. We displayed this humanoid in a public space for two weeks and people came to see Alter every day. We sampled the data collected from the motions of Alter, the neural net and the external environment and analyzed the dynamics of the data flow, to understand how its behavior changed from morning until evening. Our main results are summarized in this paper.

The architecture of Alter comprises three parts. The first is an autonomous sensor system (Maruyama et al. (2013)) developed in 2013. The second is a chaotic rhythm generator, or central pattern generator (CPG), based on coupled-phase oscillators. We use this CPG as the default controller of arm movement. When few people are nearby, Alter moves its arms purely by this CPG. The third is a spiking neural network (each neural state is described by two variables and displays 20 different kinds of neural firings), the weights of which are plastic. LSA is applied to this neural network.

Implementation of Alter's Internal Dynamics

Alter is named after the robot in the movie (`https://www.youtube.com/watch?v=jYWGzpIOqTs`), which has an alternative, non-human mind (Figure1. Alter's body has 42 movable air actuator axes, and its action production is controlled by an air compressor with a specially developed operating system (Geminoid server).

More specifically, its action is controlled by special commands (e.g., MOVEAXIS AXISNAME(int 1-42) POSITION(int 1-255)). We executed these commands from an external computer using TCP/IP through a Geminoid server. Each axis has a potentiometer, which sends the actual value of the axiss position to the computer. The refresh rate is ev-

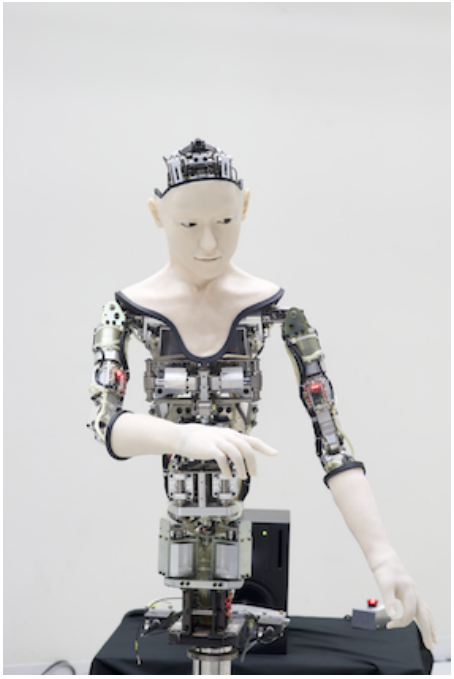

Figure 1: The external appearance of Alter. The android, which comprises only an upper body, has a height of approximately 155 cm. Photo by Kenshu Shintsubo.

ery 50 msec (Figure2). Alter's movements are determined online based on the android's internal dynamic state and sensory inputs, neither of which is fixed in advance. Al-

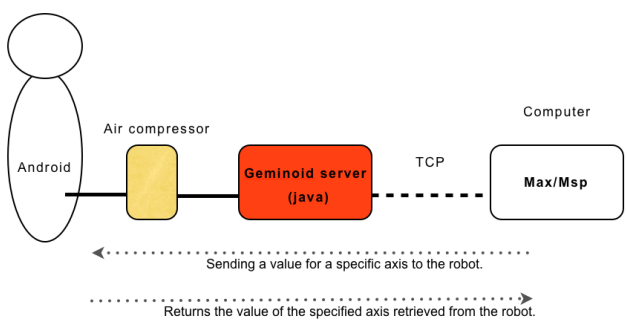

Figure 2: The overview of Alter's controlling system.

ter is surrounded by a sensory system that provides it with visual, auditory, and other sensory inputs. It is worth noting that the sensory system is active rather than passive, since it has its own autonomous dynamics. Furthermore, every axis has a potentiometer that outputs the axis's true position. Alter's internal dynamics comprise a CPG and a plastic neural network (NN), which generate motion. The CPG spontaneously creates a rhythm, and the NN creates variations of action patterns by perturbing external inputs. Therefore, it is possible to simultaneously create complicated movements and respond to the environment. In this study, we used the CPG and NN to generate the motions of Alter's arms, hands (six axes) and head angle. The other axes (for example, up-down motion of the eyes) were controlled by Gaussian noise. The mechanism for accomplishing this is explained in detail below.

Central Pattern Generator (CPG)

Rhythm (e.g., as set by a metronome) is necessary for moving a hand or changing the angle of an arm. The neural mechanism that creates rhythm inside our living systems is called a "rhythm generator" or "pattern generator." In biology, a central pattern generator, such as a neuronal circuit, can produce rhythmic motor patterns, such as walking and running, in the absence of sensory inputs (Marder and Bucher (2001), Duysens and Van de Crommert (1998)). Such CPG-initiated movements represent automatic (subconscious) motions, rather than conscious motions. The equations for the CPG are as follows:

$$\theta_{i,t+1} = \theta_{i,t} + \omega_i + \epsilon_i(pert_i) \tag{1}$$

$$\begin{aligned} pert_i = &\sin(\theta_{i,t} + \varphi) + \sin(\theta_{i+1,t} + \varphi) \\ &+ \sin(\theta_{i,t} + \theta_{i+1,t} + \varphi) + \sin(\theta_{i,t} - \theta_{i+1,t} + \varphi) \end{aligned} \tag{2}$$

where θ represents the position of the axis, ω represents the frequency of the axis's motion, ϵ represents the intensity of the perturbation, and t is the time step. ϵ and ω are randomly selected from the normal uniform noise and they are reselected every 1000 steps.

CPG produces rhythmic motion in each of the android's non-linearly coupled axes (six on either side of the arm and one in each of the two hands). When the axes begin to oscillate, they easily synchronize their phases, but often break up spontaneously to produce chaotic behavior (Herrero et al. (2012)). This emergence of chaos from the torus motion has been considered one of the salient mechanisms of producing chaotic behavior (i.e., the Ruelle Takens Newhouse theorem (Grebogi et al. (1985)). Originally, the CPG was a stable, one-way, periodic motion generator; however, in designing Alter, we used the high-dimensional coupled oscillators to enable spontaneous break-up of its periodic behavior.

Spiking Neural Network (NN)

The spiking neural network is a model of artificial neurons proposed by Eugene M. Izhikevich, who mathematically mimicked the activity of neurons in human and animal brains. (Izhikevich et al. (2003)) The equations for the neu-

ral model and its resulting dynamics are as follows:

$$\frac{dv}{dt} = 0.04v^2 + 5v + 140u + I, \tag{3}$$

$$\frac{du}{dt} = a(bv - u), \tag{4}$$

with the auxiliary after-spike resetting as follows:

$$\text{if } v > 30\text{mV then } v = c, u = u + d; \tag{5}$$

where v represents the membrane potential of the neuron, u represents a membrane recovery variable, I is the external inputs, t is the time, and a, b, c, and d are other parameters.

In Alter, we used 1000 randomly connected neurons, of which 80 % were excitatory and 20% were inhibitory. The parameters of the excitatory neurons were $a = 0.02, b = 0.2, c = -80, d = 8$; those of the inhibitory neurons were $a = 0.1, b = 0.2, c = -65, d = 2$. This ratio of 20% of inhibitory neurons is standard in simulations and close to real biological values (Meinecke and Peters (1987)). The initial weights were randomly chosen (uniform distribution: 0 ¡ w ¡ 5 for excitatory neurons, $5 < w < 0$ for inhibitory neurons). The neurons received zero-mean Gaussian noise m with a standard deviation $= 3mV$ is injected in each neuron at each time step.

The neural net has input neurons and output neurons. The input neurons were chosen randomly from the excitatory neurons; 25 neurons per axis were chosen. They received stimuli from the distance sensors and the potentiometers. The input sources depended on the learning tasks. The output neurons were also chosen randomly from the excitatory neurons; 10 neurons per axis were chosen. The position of the axis was decided on by the number of firing neurons in each time step. Alter's rhythm operates on a (semi-)macro scale; however, this rhythm is supported by neurons on an underlying micro-layer. Neurons send signals to other neurons when they overshoot, producing a macro-ordered pattern that operates at a macro level. Thus, Alter produces action by translating the micro-neural firing into macro-body motion.

Learning of NN: Principle of Avoiding Stimulation (Learning by Stimulus Avoidance)

Spike-timing-dependent plasticity (STDP) (Caporale and Dan (2008)) was discovered in both in vivo and in vitro networks. STDP causes changes in the synaptic weight between two firing neurons, depending on the timing of their activity. We used STDP to evolve the neural architecture of spiking neural networks (Sinapayen et al. (2017)) as a micro-dynamic for Alter. LSA is an emergent property of spiking networks with STDP (i.e., an example of the Hebbian rule (Hebb (1949)) in an environment. LSA states that a network learns to avoid external stimuli by self-organizing the neural architecture. To enable the emergence of LSA, the following conditions are required in addition to STDP:
(i). Once a desired task is achieved, the input must be shut down immediately.
(ii). Burst suppression is needed to prevent the network state from sliding into a trivial state. We often use short-time plasticity (Mongillo et al. (2008)) to suppress global bursting.
(iii). The neural state must be reset by instantly stimulating the input neurons. This is necessary to drive the neural state to the desired state.

LSA is different from a neural adaptation, as it is not a simple neural assimilation to input. By adjusting the network architecture, a network can actually remove the "cause" of the stimulus in the environment. In this sense, LSA is essentially embodied. It requires input from the environment, and the output from the network changes the environment. When a network removes the cause from the environment, the stimulus from the input vanishes and STDP stops modifying the weight strength. Thus the desired input-output relationship is preserved in the system. Even external input stimulation patterns change, as the network learns new behavior to avoid the new stimulation. It is thus homeostatic adaptation but not simple neural adaptation.

In our previous studies using cultured neuronal cells ((Masumori et al. (2015)); Masumori et al. (2016)) on CMOS-electrode arrays (Bakkum et al. (2008)), we showed that even cultured neural networks (about 100 neurons) can learn a desired behavior using a simple protocol (i.e., we stop stimulating a neural cell once it achieves the task).

This suggests that such homeostatic properties exist in the cultured neural networks, that the networks can form sensory-motor coupling to avoid external stimulation, and that networks are scalable (at least, biological neural networks are). In the other paper we presented at this conference, it is shown that LSA can actually work at the larger network sizes.

LSA is clearer when it is installed in a mobile robot. The robot moves around in space and, for example, runs into a wall. But the robot soon learns not to run into the wall, thanks to LSA. The aforementioned conditions 1) and 3) are naturally prepared for the mobile agent. This is why we say that LSA is essentially embodied.

We have only discussed the case of stopping undesired inputs. In order to retain desired inputs, they should be reversed so as to increase the inputs from the environment and decrease the input to the other parts of the neural net. In this way, the desired input from the environment is increased. This idea was applied to the android experiment, as described below.

We studied two tasks with the android:

(i) "Raising hands when people come close" task: In the first task, Alter was programmed to raise its hand when a person came close to it. The distance between Alter and a person was detected by a distance sensor on the autonomous

sensor network. The values of the distance sensors were sent to input neurons as stimuli. In our design, the stimuli to the neural net were canceled for 1.65 sec when Alter raised its hands higher than its shoulders. In order to stop the external stimuli, Alter needed to raise its hands. Thus, we interpreted the result to mean that Alter "learned" to raise its hands.

(ii) "Smooth movement" task: Alter also learned to move its hand in the direction in which it wanted to move. If Alter moved its hand well, it received no stimulation. Since every axis is controlled by air pressure, both the real position (i.e., the value produced by the potentiometer) and the desired position sent by the computer can be varied. The difference between the two was taken as input neurons to the network:

$$Stim(t) = 30|pos_c(t) - pos_p(t)|/255 \tag{6}$$

where pos_c is the value sent from the computer and pos_p is the value sent from the potentiometer.

We expected the neural net would learn to have smoother motions by achieving this task. In these tasks, the android spontaneously created ways to solve the problem at hand. LSA is an embodied bottom-up learning algorithm. But the algorithm functions not only in the android, it functions between the android and the environment.

The success rate of the task was measured as follows:

$$\text{Success late}(t) = \frac{N_s}{N_t} \tag{7}$$

where N_t represents the number of stimuli per minute and N_s is the number of times that $Stim(t)$ fell below 0.1 per minute.

Sensor network

This network was previously developed independently from Alter (Maruyama et al. (2013); Maruyama et al. (2014)). Here, we used it as Alter's sensory organs. The sensor network comprises several units with small CPUs that can be loaded onto various sensors (e.g., a temperature sensor, a humidity sensor, a microphone, a light sensor, a distance sensor [infrared sensor], etc.). In the experiments, we used only distance sensors and ambient light sensors (the intensity range is from $0 \sim 255$). The system allowed the sensory inputs to vary the sampling rates of each unit and each unit autonomously exchanged sensory patterns via ZigBee(zigbee alliance). The system was proposed as a minimal and autonomous model of a packet-switching network for exchanging information packets (Oka et al. (2015)). The amount of information that can be sent simultaneously per unit of time changes depending on the volume of externally sourced sensor signals. If this value exceeds a certain amount, the sampling rate decreases (e.g., so that information can be sent only once per second). The range of the sampling rate is from 0.25∼1.25 Hz. This value is normalized and used to switch between the default moving mode (CPG driven) and the conscious awareness mode (driven by the neural net) (2). This switching is explained below.

Combining the CPG with the Neural Net

Alter uses two motion modes created by a neural network and the CPG (Figure 3).

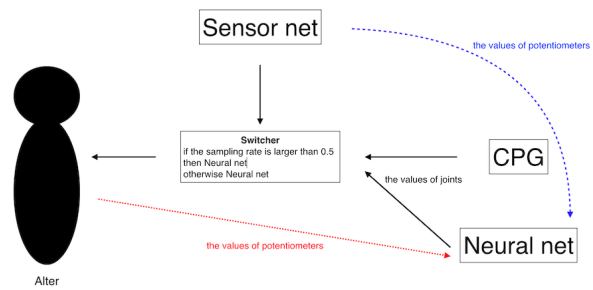

Figure 3: Overview of the combinations of the internal systems. The colored arrows describe the input values to the neural net for the tasks. The red arrow indicates the input for the "smooth movement" task. The blue arrow indicates the input for the "raising hands when people come close" task.

The CPG creates nearly constant periodic motion independently from external stimulus, and the neural net creates non-periodic motion irrespective of external stimulation. The two motion modes are as follows:

(i) Alter's default mode: Defined when the CPG creates a pattern or when the neural network creates a pattern without external input (i.e., no information is supplied from the outside). This pattern represents self-organization of the neural firing structure.
(ii) Alter in action: Defined when the neural net spontaneously self-organizes responses against the external inputs.
(iii) Switching between CPG and a neural network. It is triggered by sensor input information. The CPG works when this sampling rate (i.e., the rate of taking in information from the environment) becomes low; when it goes up, the neural net settles in.

The CPG and the neural nets were separated in this setting. It is hypothesized that complex adult behavior patterns are controlled by the central nervous system, while simple behavior, such as infants stepping behavior, is controlled by the CPG of the spinal cord, free from higher order control (Ivanenko et al. (2013)). Considering this hypothesis, we used the aforementioned sensor network to switch the output of the CPG to that of the neural network in order to link

the CPG and the neural network. As the result of this mediation, we hoped that the neural network would learn to inhibit CPG's spontaneous oscillation and control the output of the CPG. This scenario will be the next challenge.

Alter's Voice

Alter's voice is expressed through opening the mouth in accordance with the volume of the base sound. The sound scrapes the frequency according to Alter's physical exercise. Thus, this voice is generated in real time. If we used a clearly understandable human voice speaking a language such as Japanese or English, Alter would imitate the human voice. However, in the present experiment, we used a software synthesizer (u-he (Berlin) Zebra2) to create a sound pattern that was similar to a human voice but that did not use a human language.

In addition, we developed Alter to fulfill an ALIFE research goalto develop a "life-like," but not necessarily "human-like," robot. Therefore, its face is neutral, and parts of the machine are not covered with skin.

Experiments and Analysis

We conducted the experiments at the National Museum of Emerging Science and Innovation (Tokyo) between July 30 and August 6, 2016. In this "Exhibition of Alter," Alter was located in the middle of the exhibition room, where the audience could see its motions.

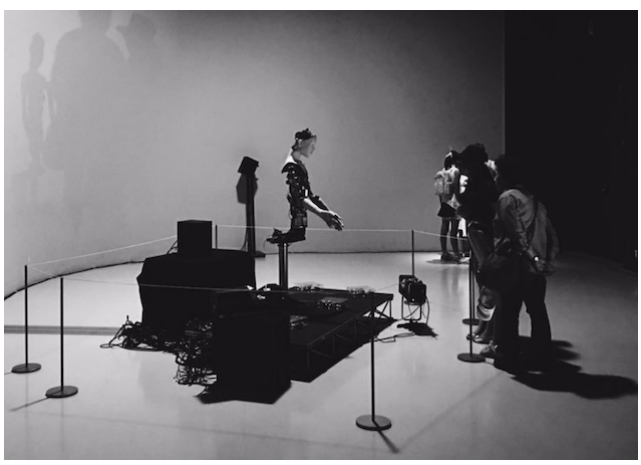

Figure 4: View of the Exhibition of Alter at the National Museum of Emerging Science and Innovation. The audience could clearly see Alter's motions. The sensors were installed at Alter's feet.

During the a week exhibition period, Alter learned two things. The first one is to let Alter to raise its hand when a person come near by. Alter also learns to moves its hand freely. This is aforementioned task 1.

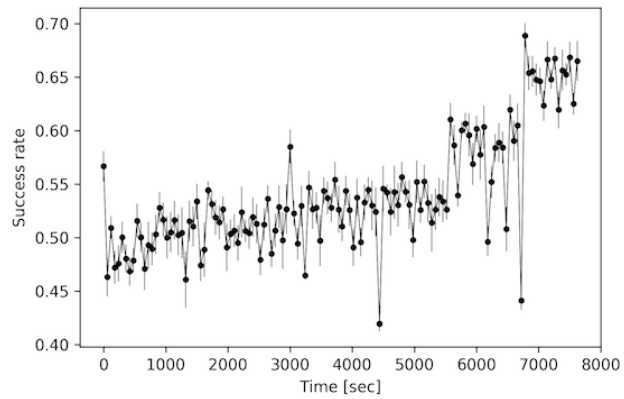

Figure 5: The time series success rate for the "smooth movement" task. The horizontal axis represents time . The error bars represent the standard error.

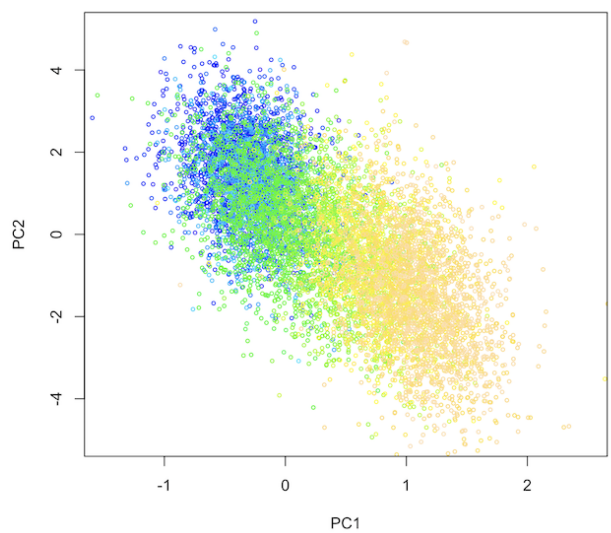

Figure 6: The results of the principal component analysis (PCA) of every potentiometer. Each point describes the average of 100 time steps of each potentiometer. The colors describe the time course (blue to yellow, through green) and the time interval is the same as that depicted in Figure 5.

Figure 5 shows the success rate of the "smooth moving" task (the horizontal axis represents time). These data were collected during the day; however, the success rate was relatively highernear 75%in the evening. In other words, over the course of the day, Alter learned and evolved so that it could move its hands much more smoothly in the evening (i.e., matching the commands from the computer and the actual body motion.) When Alter moved using pneumatic pressure, deviations occurred between the value sent from

the computer (i.e., the brain) and the actual movement. In other words, there was a gap between Alter's intention and its movement. LSA is intended to resolve this discrepancy.

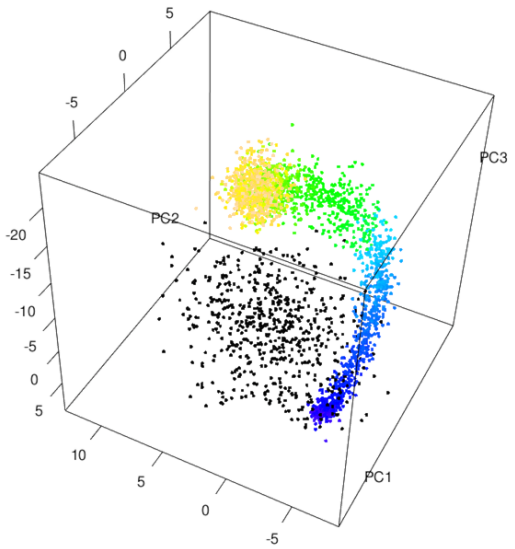

Figure 7: The results of the PCA analysis of every neural activity. Each point describes the average of 100 time steps of every neuron. The colors describe the time course, and the time interval is the same as that depicted in Figure 5 (blue to yellow, through green). The black dots indicate the pattern when the neural net received the uniform random noise (the range is the same as $I(t)$.

We also analyzed the time series for both neural activities and potentiometers. Figure 6 presents the results of the PCA performed for every potentiometer. The colors describe the time course (from blue to yellow, through green). The figure shows that Alter's body dynamics changed gradually over time. Figure7 presents the time course of the neural activities in the PCA space. Their colors correlate with those used in the previous figure. This figure also shows that the activity of the neural net changed gradually over time. In particular, the activity during the second half was more diverse than in the first half. On the other hands, the activities generated by the uniform random noise inputs were widely spreading and there were no explicit time evolutions.

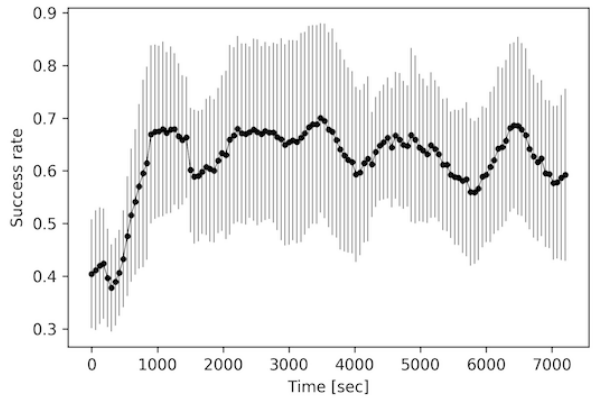

Figure 8: The time series of the success rate of the"raising hands when people com" task. The horizontal axis represents time. The error bars represent the standard error.

Figure 8 presents the success rate of the "raising hands when people come close" task, showing the results for August 5th. Let us further explore this significant increase in experience. On August 5th, there were several interactions. Many children visited, and many of these children came close to Alter and tried to shake hands with it. Perhaps this was the reason for the day's success. If fewer people had come, the android would have had fewer opportunities to use its hands. Therefore, learning progresses faster or slower depending on the characteristics of the day (e.g., the number and kind of people visiting).

The distance sensor values increased when Alter was surrounded by people and the neurons were stimulated. When Alter successfully raised its hands, it was not stimulated. However, when it did not raise its hands, the inside network was stimulated. As a result, Alter seemed to explore the motion style when input ceased. Learning was searched at random, and finally dynamics that would cause Alter to raise its hands were selected.

When neural cells are structured automatically, it is called "self-organization." However, self-organization is blind in the sense that it has no goal. One must consider how to teach neurons to self-organize into a better motion pattern. Here, LSA coupled with the environment as a self-organization engine led to the desired behavior.

Discussion

In this paper, we presented a new type of robot and described its internal mechanisms. We equipped the robot with a previously developed autonomous sensor system, an autonomous rhythm generator (CPG) with a coupled-phase oscillator, and an artificial plastic spiking neural network. We also incorporated a newly developed learning principle: LSA. LSA is a simple algorithm, and the complexity of the external en-

vironment, the sensor networks, and the spiking neural net together created a chaotic system with a large degree of freedom; however, Alter's motions did not attract periodic states (see Fig.6, Fig.7). This experiment was the first challenge in making a humanoid with only 1000 neurons. The results were better than anticipated. That is, we found that LSA functioned correctly in a humanoid.

One challenge with humanoids is to see whether Alter can cross the "uncanny valley." The uncanny valley is the hypothesis that human objects which appear almost, but not exactly, like real human beings elicit negative feelings like eeriness among some observers (Mori (1970)). Some studies have suggested that a humanoid looks "creepier" when it is moving. One reason for such creepy feeling stems from the fact that an android fails to imitate human motion, despite operating using different kinetics. The "smooth movement" task was meant to overcome this difficulty. A detailed analysis of this task will be reported as the future tasks.

References

Bakkum, D. J., Chao, Z. C., and Potter, S. M. (2008). Spatio-temporal electrical stimuli shape behavior of an embodied cortical network in a goal-directed learning task. *Journal of neural engineering*, 5(3):310.

Brooks, R. A. (1991). Intelligence without representation. *Artificial intelligence*, 47(1-3):139–159.

Caporale, N. and Dan, Y. (2008). Spike timing-dependent plasticity: a Hebbian learning rule. *Annu. Rev. Neurosci.*, 31:25–46.

Duysens, J. and Van de Crommert, H. W. (1998). Neural control of locomotion; part 1: The central pattern generator from cats to humans. *Gait & posture*, 7(2):131–141.

Grebogi, C., Ott, E., and Yorke, J. A. (1985). Attractors on an n-torus: Quasiperiodicity versus chaos. *Physica D: Nonlinear Phenomena*, 15(3):354–373.

Hebb, D. O. (1949). *The organization of behavior*. Wiley, New York.

Herrero, R., Pi, F., Rius, J., and Orriols, G. (2012). About the oscillatory possibilities of the dynamical systems. *Physica D: Nonlinear Phenomena*, 241(16):1358–1391.

Ivanenko, Y. P., Dominici, N., Cappellini, G., Di Paolo, A., Giannini, C., Poppele, R. E., and Lacquaniti, F. (2013). Changes in the spinal segmental motor output for stepping during development from infant to adult. *Journal of Neuroscience*, 33(7):3025–3036.

Izhikevich, E. M. et al. (2003). Simple model of spiking neurons. *IEEE Transactions on neural networks*, 14(6):1569–1572.

Marder, E. and Bucher, D. (2001). Central pattern generators and the control of rhythmic movements. *Current biology*, 11(23):R986–R996.

Maruyama, N., Doi, I., Masumori, A., Oka, M., Ikegami, T., Vesna, V., and Taylor, C. (2014). Evolution of artificial soundscape in a natural environment.

Maruyama, N., Oka, M., and Ikegami, T. (2013). Creating space-time affordances via an autonomous sensor network. In *Artificial Life (ALIFE), 2013 IEEE Symposium on*, pages 67–73. IEEE.

Masumori, A., Maruyama, N., Sinapayen, L., Mita, T., Frey, U., Bakkum, D., Takahashi, H., and Ikegami, T. (2015). Emergence of sense-making behavior by the stimulus avoidance principle: Experiments on a robot behavior controlled by cultured neuronal cells. In *13th European Conference on Artificial Life (ECAL 2015)*, pages 373–380.

Masumori, A., Maruyama, N., Sinapayen, L., Mita, T., Frey, U., Bakkum, D., Takahashi, H., and Ikegami, T. (2016). Learning by stimulation avoidance principle on cultured neuronal cells. *In Proc. of annual meeting of Physical Society of Japan2016*, page 71(1) 3092.

Meinecke, D. L. and Peters, A. (1987). Gaba immunoreactive neurons in rat visual cortex. *Journal of Comparative Neurology*, 261(3):388–404.

Mongillo, G., Barak, O., and Tsodyks, M. (2008). Synaptic theory of working memory. *Science*, 319(5869):1543–1546.

Mori, M. (1970). Bukimi no tani (the uncanny valley). *Energy*, 7(4):33–35.

Oka, M., Abe, H., and Ikegami, T. (2015). Dynamic homeostasis in packet switching networks. *Adaptive Behavior*, 23(1):50–63.

Shahaf, G. and Marom, S. (2001). Learning in networks of cortical neurons. *Journal of Neuroscience*, 21(22):8782–8788.

Sinapayen, L., Masumori, A., and Ikegami, T. (2017). Learning by stimulation avoidance: A principle to control spiking neural networks dynamics. *PloS one*, 12(2):e0170388.