

Mechanism of Emergent Symmetry Properties on Evolutionary Robotic System

Naohide Yasuda, Takuma Kawakami, Hiroaki Iwano,
Katsuya Kanai, Koki Kikuchi and Xueshan Gao
Chiba Institute of Technology
Japan

1. Introduction

In order to create an autonomous robot with the ability to dynamically adapt to a changing environment, many researchers have studied robotic intelligence, especially control systems, based on biological systems such as neural networks (NNs), reinforcement learning (RL), and genetic algorithms (GA) (Harvey *et al.*, 1993, Richard, 1989, and Holland, 1975). In a recent decade, however, it has been recognized that it is important to design not only robotic intelligence but also a structure that depends on the environment as it changes because the dynamics of the structural system exerts a strong influence on the control system (Pfeifer & Scheier, 1999, and Hara & Pfeifer, 2003). The behavior of a robot is strongly affected by the physical interactions between its somatic structure and the outside world, such as collisions or frictions. Additionally, since the control system, the main part of robotic intelligence, is described as a mapping from sensor inputs to actuator outputs, the physical location of the sensors and actuators and the manner of their interaction are also critical factors for the entire robotic system. Therefore, to design a reasonable robot, it is necessary to consider the relationship between the structural system and the control system, as exemplified by the evolution of living creatures.

From this point of view, several researchers have tried to dynamically design structural systems together with control systems. Sims (Sims, 1994) and Ventrella (Ventrella, 1994) have demonstrated the evolution of a robot with a reconfigurable multibody structure and control system through computer simulation. The Golem Project of Lipson and Pollack has realized the automatic design and manufacture of robotic life forms using rapid prototyping technology (Lipson & Pollack, 2000). Salomon and Lichtensteiger have simulated the evolution of an artificial compound eye as a control system by using NNs and have shown that the robot creates motion parallax to estimate the critical distance to obstacles by modifying the angular positions of the individual light sensors within the compound eye (Salomon & Lichtensteiger, 2000). These researches have shown the importance of adaptation through not only intelligence but also the relationship between morphology and intelligence. However, the mechanism of the function emerging from such relationship or some kind of design principle is not fully understood yet.

Meanwhile, for living creatures, symmetry properties may be a common design principle; these properties may have two phases, that is, the structural and functional phases. For

example, most legged creatures are symmetric in the structural phase and their gait, that is, the manner in which they actuate their left and right legs, is also symmetric in the functional phase. For the locomotion of a biped robot, Bongard *et al.* have demonstrated the importance of a symmetry structure from the viewpoint of energy efficiency (Bongard & Paul, 2000, and Bongard & Pfeifer, 2002). This is an example of effective symmetry structure from the viewpoint of engineering. However, the effectiveness of an asymmetry structure has also been shown in nature. Although insect wings to fly are symmetric, those to sing are generally asymmetric. One claw of the fiddler crab is extremely big as compared with another. The asymmetric brain structure of a fruit fly enhances its long-term memory (Pascual *et al.*, 2004) and an asymmetric ear structure of barn owls allows accurate auditory localization (Kundsén, 2002). These examples indicate that since living beings must have created optimal mechanisms through interactions with the environment, the characteristics of symmetry or asymmetry are extremely important for not only the physical structure but also functionality, including control. Hence, since the symmetry properties and their concomitant functionality show the design principle of the entire system, the clarification of the mechanism of the emergence of symmetry properties can contribute to the development of a methodology for a robotic system that designs its own morphology and intelligence depending on the changing environment.

From this point of view, we have studied the mechanism of symmetry properties emerging from the balance between structural and control systems by using an evolutionary robotic system with reconfigurable morphology and intelligence (Kikuchi & Hara, 1998, Kikuchi *et al.*, 2001, and Kikuch & Hara, 2000). Here, as an example of our studies, we introduce the symmetry properties created by two relative velocity conditions, fast predator vs. slow prey and slow predator vs. fast prey, and by genotype-phenotype noise conditions, genetic errors due to a growth process.

2. Task and Evolutionary Robotic System

In this section, we introduce a task for a robot, a fitness criterion, and an evolutionary robotic system.

2.1 Task and Evaluation

The task given to the robot is to maintain a certain distance D from a target. The robot and the target are in an arena surrounded by walls, as shown Fig. 1. The target moves randomly and the robot behaves by using the morphology and intelligence automatically generated by genetic programming (GP). Note that, the short distance D means that since the robot chases the target, the predator chases the prey. On the other hand, the long distance D means that since the robot departs from the target, the prey runs away from the predator.

A fitness value, F , is calculated according to the performance of the robot. The performance is evaluated by using a multiobjective function that is defined as

$$F = \frac{1}{N} \sum_{i=1}^N \left\{ \frac{1}{T} \int_0^T \alpha (\|P - X\| - D)^2 dt \right\}$$

$$\begin{cases} (\|P - X\| - D) \geq 0 & \alpha = 1 \\ (\|P - X\| - D) < 0 & \alpha = \frac{\sqrt{2}H}{D} \end{cases} \quad (1)$$

where X is the center of the robot, P is the center of the target, t is the time, T is the total evaluation time, H is the side length of the arena, i is the trial number, and N is the total number of trials. The robot obtains a high evaluation if it maintains D . Here, the weight α is determined by the distance between the robot and the target. When this distance is smaller than D , α is $\sqrt{2H/D}$, and when it is larger than D , α is 1. Note that the value of $\sqrt{2H}$ means the maximum distance of the robot and the target. Additionally, the smaller the fitness value, the better the performance. When the robot collides with the target, the fitness value is 1.63 and when the robot maintains an objective distance, it is 0.0.

2.2 Evolutionary Robotic System

The robot is modeled as a cylindrical shape and has two visual sensors and two wheels with motors. The motion is calculated on the basis of a two-dimensional dynamic model that includes realistic physical conditions such as collisions and frictions. The equations of motion are given by

$$\begin{aligned} M\ddot{x} &= \sum_{i=0}^1 \frac{T_i}{R_i} \cos\theta + F_x + P_x \\ M\ddot{y} &= \sum_{i=0}^1 \frac{T_i}{R_i} \sin\theta + F_y + P_y \\ I\ddot{\theta} &= \sum_{i=0}^1 \frac{T_i}{R_i} r + F_\theta + P_\theta \end{aligned} \quad (2)$$

where M is the mass of the robot, x and y are the coordinates of the center of the robot, T_i is the torque of the motor, R_i is the wheel radius, r is the distance from the center of the wheel to the center of the robot (equals to the robot radius), F_* is the friction with the floor, P_* is the impact with the target or a wall, I is the moment of inertia, θ is the direction of the robot, and i is the wheel ID that is 0 for the left wheel and 1 for the right wheel, as shown in Fig. 2. Note that the origin is the center of the arena and the counterclockwise direction is positive, as illustrated in Fig. 1. Using these equations, the motions of the robot and target are simulated by a Runge-Kutta method.

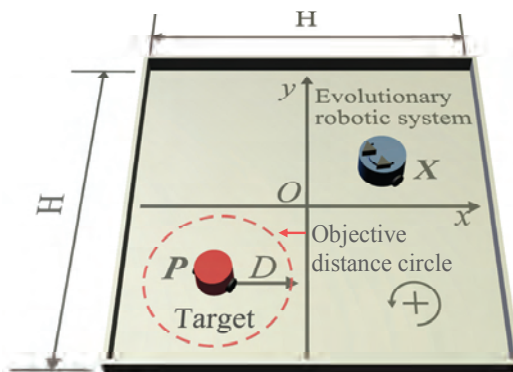


Figure 1. Simulation arena

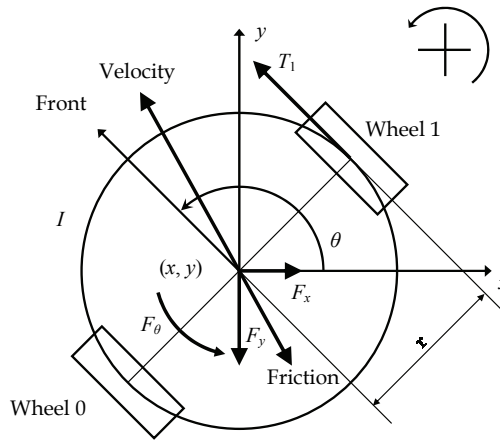


Figure 2. Two-dimensional model of evolutionary robotic system

3. Morphology and Intelligence Genes

In this study, the evolutionary robotic system is optimized through the processes of GP: (1) development, (2) evaluation, (3) selection and multiplication, and (4) crossover and mutation. Under GP, each robot is treated as an individual coded by a morphology gene and an intelligence gene. In this section, we explain the coding method.

3.1 Morphology Gene

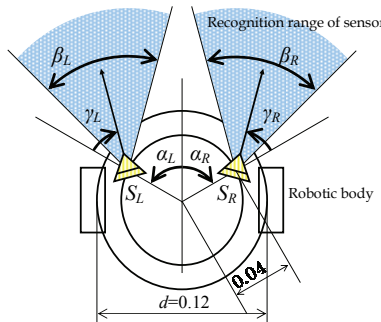


Figure 3. Morphological parameters

A morphology of a robot may be generally defined by using many kinds of elements such as the body shape, size, weight, rigidity, surface characteristics, and sensor-actuator arrangement. In this study, the morphology is represented by the physical arrangement of two flexible visual sensors, two fixed motors, and a cylindrical shape, as illustrated in Fig. 3. Here, two visual sensors S_L and S_R have three degrees of freedom: alpha, beta, and gamma. Alpha corresponds to the arrangement angle of the sensors on a circumference of a circle with a radius of 0.04 m ($0^\circ \leq \alpha_L, \alpha_R \leq 90^\circ$), beta is the range of the field of view

($0^\circ \leq \beta_L, \beta_R \leq 50^\circ$), and gamma is the direction of the visual axis ($-90^\circ \leq \gamma_L, \gamma_R \leq 90^\circ$). Thus, the evolutionary robotic system has six degrees of freedom for the morphology gene. Note that the shaded areas show the recognition areas for the target; the sensor becomes "ON" when the target is recognized in this area. The sensor resolution is set to be 1 for simplicity.

3.2 Intelligence Gene

The intelligence gene of the robot is a computer program described as a decision tree that represents the relationship between the sensor inputs and the motor outputs. The decision tree is created by using two kinds of nodes--terminal nodes and nonterminal nodes--as shown in Table 1. The terminal nodes are the sensor nodes and motor nodes. The sensor nodes *L* and *R* correspond to the state of the two sensors S_L and S_R shown in Fig. 3, with "true" and "false" assigned to "ON" and "OFF." The motor nodes have the action functions such as *MOVE_F* to move forward, *TURN_L* to turn left, *TURN_R* to turn right, *MOVE_B* to move backward, and *STOP* to stop. Figure 4 shows the behavior of these functions. The nonterminal nodes are function nodes, i.e., typical computer language commands such as *IF*, *AND*, *OR*, and *NOT*. The robotic intelligence gene is automatically created by combining these nodes.

Terminal nodes	
Sensor nodes	<i>L, R</i>
Motor nodes	<i>MOVE_F, TURN_L, TURN_R, MOVE_B, STOP</i>
Nonterminal nodes	
Function nodes	<i>IF, AND, OR, NOT</i>

Table 1. Node for decision tree

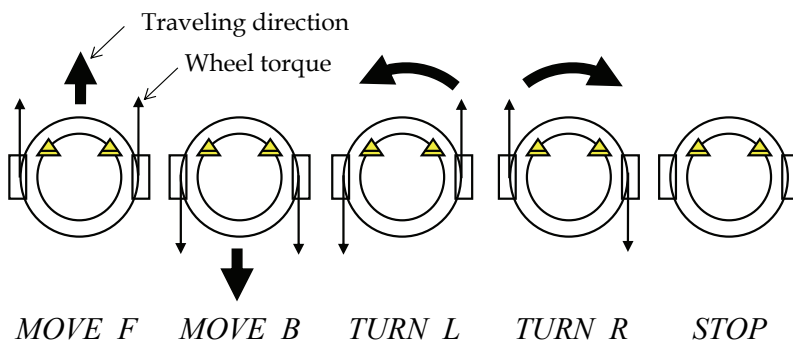


Figure 4. Robotic behaviors for each motor node

4. Evolutionary Simulation I

4.1 Conditions for Simulation

In this study, to clarify the mechanism of emergent symmetry properties, we performed two simulations for different relative velocities of the robot: in Case A, the robot was twice as fast as the target and in Case B, the target was twice as fast as the robot. Since we set an

objective distance D as a short distance of 0.5 m, the robots mean the fast predator in Case A and the slow predator in Case B.

The physical conditions were as follows. The length of one side of the arena H was 4.0m, the diameter of the robot and target d was 0.12 m, the evaluation time T ranged from 20.0 s to 90.0 s, the maximum speeds of the robot and target were 0.2 m/s and 0.1m/s, respectively, in Case A and 0.1 m/s and 0.2 m/s, respectively, in Case B, the sampling time of the sensor was 100 ms, and the weight of the robot and the target M was 0.4 kg. The recognition error of the sensors was set from -3.0° to 3.0° (randomly determined from a normal distribution). The GP parameters were set as follows. The population size was 300, the generation was 300, the selection ratio was 0.8, the crossover ratio was 0.3, and the mutation ratio was 0.1. The initial positions and directions of the robot and target were randomly determined from a uniform distribution within the center region.

4.2 Definition: Indices of Symmetry Properties

To analyze the structural symmetry properties of the robotic system, we defined three indices: $|\alpha_L - \alpha_R|$, $|\beta_L - \beta_R|$, and $|\gamma_L - \gamma_R|$. Hence, the smaller the indices, the higher was the structural symmetry. In the development of the first GP process, these values were uniformly generated to avoid bias.

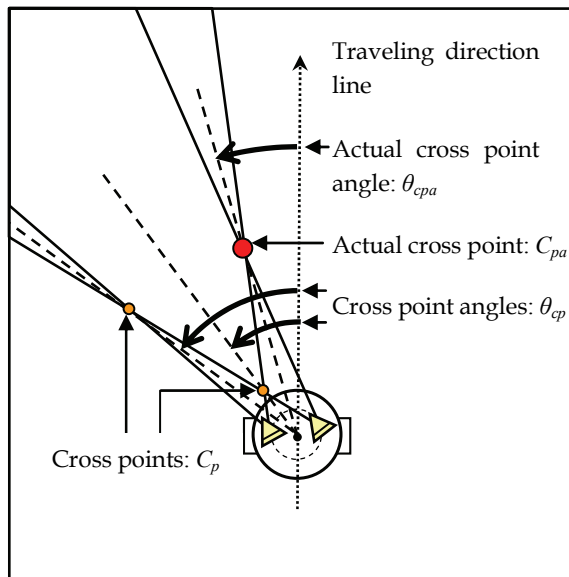


Figure 5. Definition of cross-points and cross-point angles

Additionally, we defined another index for the state space created by the visual sensors. As illustrated in Fig. 5, the values C_p represent the cross-points of the recognition areas of the two visual sensors, and the values θ_{cp} represent the angle between the traveling direction line and the line connecting the cross-points and the center of the robot. Note that the maximum cross-point number is four, since each visual sensor has two edges of recognition

area. We further defined the cross-point that is employed for action assignment as an actual cross-point C_{pa} . Similarly, θ_{cpa} represents the actual cross-point angle. Using these parameters, we performed 20 simulations for each case and analyzed elite individuals in the final generation of each simulation.

4.3 Results

Table 2 shows the fitness averages of the elite individuals obtained in Cases A and B and the standard deviations. The fitness in Case A is better than that in Case B, since the robot is faster than the target and can quickly approach it. Here, the fitness value of 0.218 means that the robot departs averagely 0.14 m inside from the objective distance circle shown in Fig. 1, and 0.278 means that it departs averagely 0.16 m inside.

	Case A	Case B
Ave.	0.218	0.278
Std. dev.	0.056	0.086

Table 2. Fitness in Cases A and B

Morphology genes	$\alpha_R=65, \beta_R=35, \gamma_R=-32$
[deg]	$\alpha_L=37, \beta_L=27, \gamma_L=64$
Intelligence genes	(if (not L) <i>TURN_L</i> <i>MOVE_F</i>)

Table 3. Genotype of typical individual obtained in Case A (Type I)

Table 3 and Fig. 6 show the genotype and phenotype of a typical individual obtained in Case A, respectively. This individual divides the state space into two regions and assigns two actions. Here, we defined this kind of individual as Type I. This type occupies 52.5% out of 200 individuals in Case A and accomplishes the task of maintaining a certain distance from the target by using the following simple strategy. As shown in the intelligence gene of Table 3, if L is not true, then *TURN_L* is executed; in other words, if the left visual sensor does not recognize the target, the robot turns left (State 1 in Fig. 6). Otherwise, if L is true, *MOVE_F* is executed, that is, if the left visual sensor recognizes the target, the robot moves forward (State 2 in Fig. 6). Here, *MOVE_F* in the state space is arranged in the right front of the robot and *TURN_L* occupies the rest of the state space. Further, the robot has two visual sensors, but actually uses only one. In Case A, the robot is two times faster than the target and collides with it if the *MOVE_F* is arranged in front of the robot. Thus, the Type I robot avoids a collision and maintains the objective distance by shifting the *MOVE_F* from the front and rotating frequently.

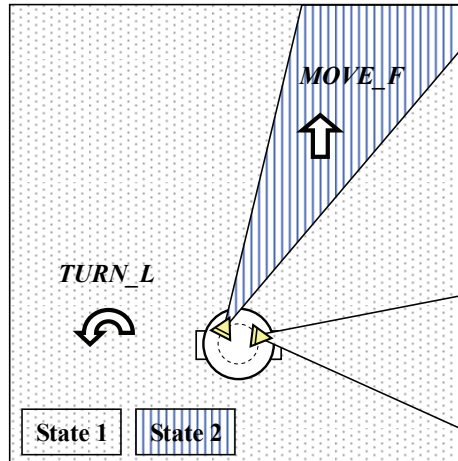


Figure 6. Phenotype of typical individual expressed by the genotype of Table 3 (Type I)

Morphology genes	$\alpha_R=63, \beta_R=27, \gamma_R=82$
[deg]	$\alpha_L=48, \beta_L=50, \gamma_L=30$
Intelligence genes	(if L (if R TURN_L MOVE_F) (if (not R) TURN_R MOVE_B))

Table 4. Genotype of typical individual obtained in Case B (Type II)

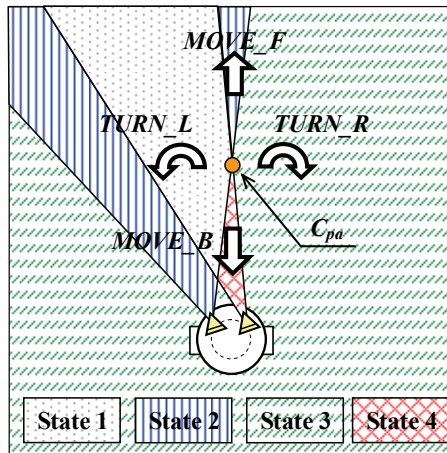


Figure 7. Phenotype of typical individual expressed by the genotype of Table 4 (Type II)

Next, Table 4 and Fig. 7 show the genotype and phenotype of a typical individual obtained in Case B. This individual divides the state space into four regions and assigns four actions. Here, we defined this kind of robot as Type II. This type occupies 57.5% out of 200 individuals in Case B and accomplishes the task by using the following strategy. As shown in the intelligence gene of Table 4, if *L* and *R* are true, then *TURN_L* is executed, that is, if

both sensors recognize the target, the robot turns left (State 1 in Fig. 7). If L is true and R is not true, then $MOVE_F$ is executed; in other words, if the left visual sensor recognizes the target and the right visual sensor does not, the robot moves forward (State 2 in Fig. 7). If both R and L are not true, then $TURN_R$ is executed, that is, if neither the left visual sensor nor the right visual sensor recognizes the target, the robot turns right (State 3 in Fig. 7). If L is not true and R is true, $MOVE_B$ is executed, that is, if the left visual sensor does not recognize the target and the right visual sensor does, the robot moves backward (State 4 in Fig. 7). Here, $MOVE_F$ in the state-action space is arranged in the front of the robot, $TURN_L$ and $TURN_R$ are to the left and right of the $MOVE_F$ region, and $MOVE_B$ is between $MOVE_F$ and the robot. In Case B, the robot is two times slower than the target and needs to approach the target along the shortest path. Therefore, $MOVE_F$ should be arranged in the front of the robot. Additionally, the arrangement of $TURN_L$ and $TURN_R$ for $MOVE_F$ allows a fast search and the centering of the target. Furthermore, when the robot gets too close to the target, it moves backward and maintains a distance between the robot and the target. With this state-action space, Type II obtains better fitness as compared to the others in Case B.

4.4 Discussion: Structural Symmetry Properties

Table 5 shows the averages of the structural indices of the symmetry properties: $|\alpha_L - \alpha_R|$, $|\beta_L - \beta_R|$, and $|\gamma_L - \gamma_R|$, for the elite individuals in the final generation and the standard deviations. Since the standard deviations were high and the averages did not converge, distinguishing structural symmetry properties represented by the arrangement of the two visual sensors were identified. This result shows that a structural symmetry property does not clearly manifest in simulation without considering the physical factor such as sensor weight.

	Case A [deg]		Case B [deg]	
	Ave.	Std. dev.	Ave.	Std. dev.
$ \alpha_L - \alpha_R $	29.5	27.2	25.5	19.4
$ \beta_L - \beta_R $	14.3	7.9	14.5	13.3
$ \gamma_L - \gamma_R $	37.4	29.1	42.4	36.0

Table 5. Results of structural indices obtained in Cases A and B

4.5 Discussion: Functional Symmetry Properties

From the viewpoint of state-action space, we discuss the phenotype of Type II. Figure 8 shows the state-action space of Type II and the physical arrangement of the two visual sensors. As shown by the broken lines, the state-action space of Type II is symmetric about the line between the actual cross-point and the center of the robot, because $MOVE_F$ and $MOVE_B$ are arranged in the front and back for the actual cross-point and $TURN_L$ and $TURN_R$ are to its left and right. In this study, we define such symmetry of the state-action space as functional symmetry. This result shows that from the viewpoint of physical structure, the arrangement of the two visual sensors is not symmetric (the lower area marked by the broken line in Fig. 8), but from the viewpoint of control, the state-action space is symmetric. Table 6 shows the incidence ratio of an individual with functional

symmetry obtained in Cases A and B. Since the ratios are 10.0% in Case A and 57.5% in Case B, the relative velocity difference must be one of factors that generate the functional symmetry. Table 7 shows the average of the actual cross-point angle of the individual with functional symmetry obtained in Cases A and B. Here, if the actual cross-point is 0 [deg] (i.e., exists on the traveling direction line), it means that the state-action space is almost symmetric about the traveling direction. The actual cross-point angle in Case B is lower than that in Case A, that is, the individual obtained in Case B is more symmetric. Furthermore, as shown in Fig. 9, 25% of individuals with functional symmetry in Case A created the actual cross-point within 10 [deg], while the percentage of individual in Case B was 90%. This result suggests that in Case B, most of the robotic systems designed the actual cross-point in front of the robot and assigned actions based on this point. Therefore, the condition in Case B tends to generate functional symmetry about the traveling direction as compared with that in Case A. Furthermore, Fig. 10 shows the relationship between the actual cross-point angle and the fitness in Case B. This result shows that since the correlation is 0.38, the smaller the actual cross-point angle (i.e., the more the functional symmetry), the better the fitness. This is considered to be due to the following reason. In Case B, the robot consumes considerable amount of time in chasing the target because the target velocity is twice that of the robot velocity. Thus, the Type II robot has the fastest approach by creating a region of *MOVE_F* in the travelling direction, as shown Fig. 8. In addition, this type of robot can quickly cope with the random behavior of the target by symmetrically assigning actions based on the actual cross-point. Hence, functional symmetry properties about the traveling direction emerging from the arrangement of the two visual sensors are one of the important design principles in Case B. These result may show that, in nature, since a slower predator, for example, a tiger, compared with a prey must efficiently chase, it almost creates the symmetric stereo-vision.

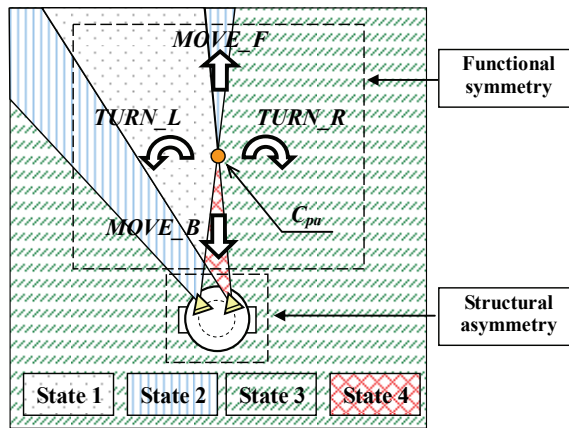


Figure 8. State-action space of individual with functional symmetry

Case A [%]	Case B [%]
10.0	57.5

Table 6. Incidence ratio of individual with functional symmetry obtained in Cases A and B

	Case A [deg]	Case B [deg]
Ave.	12.5	4.6
Std. dev.	3.4	3.7

Table 7. Actual cross-point angle obtained in Cases A and B

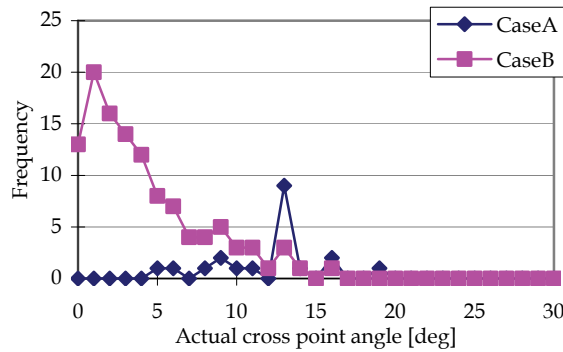


Figure 9. Comparison of actual cross-point angle distribution obtained in Cases A and B

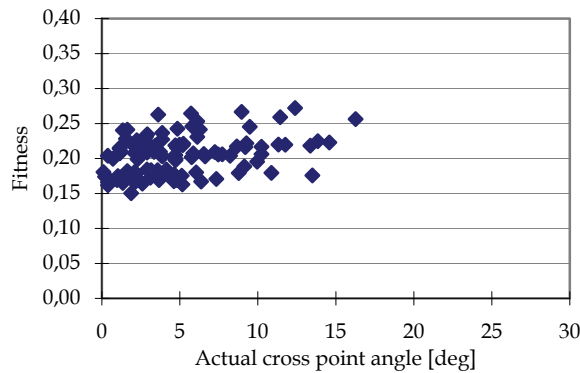


Figure 10. Correlation between actual cross point angle and fitness obtained in Case B

5. Evolutionary Simulation II

5.1 Condition for Simulation

To investigate the influence of genetic noise on the manifestation of symmetry properties, we performed same simulations identical to Evolutionary Simulation I for the noise ratios: 0%, 25%, 50%, 75%, and 100%. Here, the genetic noise is a genotype-phenotype noise (G-P noise) that is added during the transformation process from the genotype to the phenotype. From this, an individual with same genotype is translated to slightly different phenotypes and is given a different fitness value. This G-P noise may be similar to an

acquired disposition in nature. The G-P noise adds a disturbance from -1.0 [deg] to 1.0 [deg] to α , β , and γ of the genotype according to the normal probability distribution. Note that the change in the sensor direction in the traveling direction due to this G-P noise is less than 2.0 [deg], and that the change in the edge of the field of view is less than 2.5 [deg].

5.2 Results and Discussion

Table 8 shows the incidence ratio of an individual with functional symmetry for different G-P noise ratios. From this, we find that in Case A, the incidence ratio of functional symmetry gradually decreases with an increase in the G-P noise, and in Case B, there is a peak in the incidence ratio of functional symmetry depending on the G-P noise ratio. We consider this mechanism as follows. Type I is robust against the G-P noise as compared with Type II. Since Type II designs the state-action space based on the cross-point, a change in the cross-point due to the G-P noise deteriorates the fitness. However, Type I is not affected much, since it does not have a cross-point. Thus, in Case A, Type II is eliminated with an increase in the G-P noise. Consequently, the individuals in Case A lose one visual sensor through evolution and become Type I with high robustness in the presence of G-P noise. The Type I visual sensor has a bias angle (approximately 30 [deg]) for the traveling direction and is asymmetric. Hence, Case A creates functional asymmetry.

On the other hand, Type II must use the state-action space with a cross-point for the fastest chase. Therefore, Type II is not eliminated by increasing the G-P noise in Case B. Moreover, a small G-P noise increases the incidence of functional symmetry. Hence, Case B creates functional symmetry. From this, in this case study, we conclude that Case A, in which the robot is faster, creates functional asymmetry and Case B, in which the target is faster, creates functional symmetry.

G-P Noise	Case A	Case B
0%	10.0	57.5
25%	14.0	57.5
50%	6.0	80.5
75%	5.0	57.5
100%	0.5	56.5

Table 8. Incidence ratios of individual with functional symmetry

6. Conclusions

We focused on symmetry properties and performed computational simulations by using an evolutionary robotic system with reconfigurable morphology and intelligence. First, we investigated the mechanism of emergent symmetry properties for two different relative velocities of the robot and the target and the influence by which the genetic noise gives to the symmetry properties. Although, from the viewpoint of physical structure,

distinguishing structural symmetry properties were identified in simulation without considering the physical factor such as sensor weight, from the viewpoint of control, functional symmetric properties were manifested; functional asymmetry was designed in Case A in which the robot was faster than the target, and functional symmetry was designed in Case B in which the robot was slower than the target. Genotype-phenotype noise, which creates different individuals from the same genotype, improved the robustness of the robot in Case A and raised the incidence ratio of functional asymmetry. On the other hand, a small genotype-phenotype noise improved the incidence ratio of functional symmetry in Case B. In a future study, we intend to investigate the relationship between the sensory system and the driving system using an evolutionary robotic system that is capable of changing not only the sensor arrangement but also the motor arrangement. Additionally, we aim to further investigate the design principles leading to structural symmetry. Furthermore, we will employ physical experiments and attempt to reveal the characteristics of symmetry properties in the real world.

7. References

- Bongard, J. C. & Paul, C. (2000). Investigation Morphological Symmetry and Locomotive Efficiency using Virtual Embodied Evolution, *From animaru to animates 6*, pp. 420-429
- Bongard, J. C. & Pfeifer, R. (2002). A Method for Isolating Morphological Effects on Evolved Behavior, *7th International Conference on the Simulation of Adaptive Behavior*, pp. 305-311
- Hara, F. & Pfeifer, R. (2003). *Morpho-functional Machines : The New Species*, Springer
- Harvey, I. ; Husband, P. & Cliff, D. (1993). Issues in Evolutionary Robotics, *From animals to animates 2*, pp. 364-373
- Holland, J. H. (1975). *Adaptation in natural and Artificial System*, Univ. Of Michigan Press
- Kikuchi, K. & Hara, F. (1998). Evolutionary Design of Morphology and Intelligence in Robotic System Using Genetic Programming, in *Proc. Fifth International Conference of the Society for Adaptive Behavior 98*, pp. 540-545
- Kikuchi, K. & Hara, F. (2000). A Study on Evolutionary Design in Balancing Morphology and Intelligence of Robotic System, *JSME, Journal of Robotics and Mechatronics*, pp. 180-189,
- Kikuchi, K.; Hara, F. & Kobayashi, H. (2001). Characteristics of Function Emergence in Evolutionary Robotic Systems -Dependency on Environment and Task-, in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2288-2293
- Knudsen, E. I. (2002). Instucted learning in the auditory localization pathway of the barn owl, *Nature*, Vol. 417, pp. 322-328
- Lipson, H. & Pollack, J. B. (2000). Automatic design and manufacture of robotic lifeforms, *Nature*, Vol. 406, pp. 974-978
- Pascual, A.; Huang, K.; Neveu, J. & Preat, T. (2004). Brain asymmetry and long-term memory, *Nature*, Vol. 427
- Pfeifer, R. & Scheier, C. (1999). *Understanding Intelligence*, The MIT press
- Richard, S. (1989). *Temporal Credit Assignment in Reinforcement Learning*, PhD thesis, Univ. Cambridge, England

- Salomon, R. & Lichtensteiger, L. (2000). Exploring different Coding Schemes for the Evolution of an Artificial Insect Eye, in Proc. First IEEE Symposium on Combinations of Evolutionary Computation and Neural Networks
- Sims, K. (1994). Evolution 3D Morphology and Behavior by Competition, *Artificial Life 4*, MIT press, pp. 28-39
- Ventrella, J. (1994). Exploration in the Emergence of Morphology and Locomotion Behavior in Animated Characters, *Artificial Life 4*, pp. 436-441