

Multiple Chaos Generator by Neural-Network-Differential-Equation for Intelligent Fish-Catching

Yuya Itou

Faculty of Engineering
Okayama University
Okayama, Japan

Email: itou@suri.sys.okayama-u.ac.jp

Takashi Tomono

Faculty of Engineering
Okayama University
Okayama, Japan

Email: gen422236@s.okayama-u.ac.jp

Mamoru Minami

and Akira Yanou

Faculty of Engineering Okayama University
Okayama, Japan

Email: minami, yanou@suri.sys.okayama-u.ac.jp

Abstract—Continuous catching and releasing experiment of several fishes makes the fishes find some escaping strategies such as staying stationary at corner of the pool. To make fish-catching robot intelligent more than fishes' adapting and escaping abilities from chasing net attached at robot's hand, we thought something that goes beyond the fishes' adapting intelligence will be required. Here we propose a chaos-generator comprising Neural-Network-Differential-Equation(NNDE) and an evolving mechanism to have the NNDE generate chaotic trajectories as many as possible. We believe that the fish could not be adaptive enough to escape from chasing net with chaos motions that have many different chaos, since unpredictable chaotic motions of net may go beyond the fishes' adapting abilities to the net motions. In this report we introduce the chaos generating system by NNDE, which can produce many kinds of chaos theoretically, and then analyze the chaos with Lyapunov number, Poincare return map and initial value sensitivity.

I. INTRODUCTION

In recent years, visual tracking and servoing in which visual information is used to direct the end-effector of a manipulator toward a target object has been studied in some researches [1], [2]. A new trend of machine intelligence [3] that differs from the classical AI has been applied intensively to the field of robotics and other research areas like intelligent control system. Typically, the animal world has been used conceptually by robotics as a source of inspiration for machine intelligence. For the purpose of studying animal behavior and intelligence, the model of interaction between animals and machines is proposed in researches like [4]. In our previous research, the fish emotional behavior has also been examined and the robot with adaptive ability to react to the fish status has been conceived. Another crucial characteristic of machine intelligence is that the robot should be able to use input information from sensor to know how to behave in a changing environment and furthermore can learn from the environment for safety like avoiding obstacle. As known universally that the robot intelligence has reached a relatively high level, still the word "intelligence" is an abstract term, so the measurement of the intelligence level of a robot has become necessary. A practical and systematic strategy for measuring machine intelligence quotient (*MIQ*) of human-machine cooperative

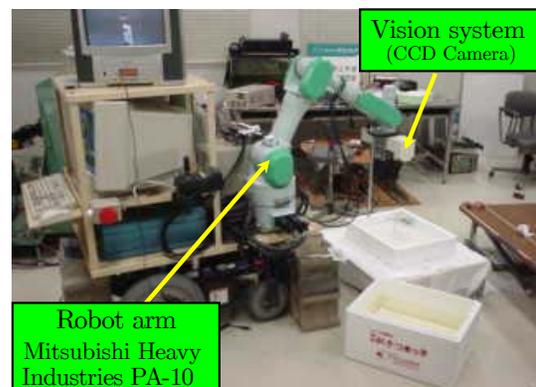


Fig. 1. Fish Catching system PA10

systems is proposed in [5]. In our approach to pursue intelligent robot, we will evaluate the intelligence degree between fishes and the robot by Fish-Catching operation. We think that the system combined with chaos be smarter than the fish when the robot can beat the fish by catching it successfully even after the fish finds out some escaping strategy. As we did not find the research about the intelligence comparison between animal and robot, we mainly dedicate ourselves to constructing a smart system that is more intelligent than the fish. We consider that the competitive relation can be very meaningful as one way to discuss robotic intelligence. So we not only employ the inspiration of animal's behavior for robot intellectualization, we can also conceive a robot that can exceed the animal intelligence. By evolutionary algorithms [6], Visual Servoing and Object Recognizing based on the input image from a CCD camera mounted on the manipulator has been studied in our laboratory(Fig.1) [7], and we succeeded in catching a fish by a net attached at the hand of the manipulator based on the real-time visual tracking under the method of Gazing GA [8] to enhance the real-time searching ability.

We have learned that it is not effective for fish catching to simply pursue the current fish position by visual servoing with velocity feedback control. Actually, the consistent tracking is sometimes impossible because the fish can alter motion pattern suddenly maybe under some emotional reasons of fear. Those

behaviors are thought to be caused by emotional factors and they can also be treated as a kind of innate fish intelligence, even though not in a high level.

While observing the fishes' adapting behavior to escape in the competitive relations with the robot, that is continuous catching/releasing experiments, we found that we can define a "Fish's Intelligent Quotient"(FIQ)[9] representing decreasing velocity of fish number caught by the net through continuous catching/releasing operation. Through this measure we can compare the innate intelligence of the fish and the artificial intelligence of the robot.

It has been well known that many chaotic signals exist in our body, for example, in nerves, in motions of eye-balls and in heart-beating periods [10], [11].

Therefore we thought that imitating such animal's internal dynamics and putting chaos into robots have something meaningfulness to address fishes' intelligence. We embed chaos into the Robot Dynamics in order to supplement the deficiency of our Fish-Catching system.

Therefore what we have to pay attention to the fishes' nature that the fish does conceive always escaping strategy against new stressing situation. This means that robot's intelligence to override the fishes' thinking ability needs infinite source of idea of catching motions. To generate such catching motion, we propose in this report Neural-Network-Differential-Equation(NNDE) that can produce neural chaos and inherently have a possibility to be able to generate infinite varieties of chaos, derived from the neural network's ability to approximate any nonlinear function as accurate as with desirable precision[12], [13].

II. FISH TRACKING AND CATCHING

The problem of recognition of a fish and detection of its position/orientation is converted to a searching problem of $\mathbf{r}(t) = [x(t), y(t)]^T$ in order to maximize $F(\mathbf{r}(t))$, where $F(\mathbf{r}(t))$ represents correlation function of images and fish-shaped matching model. $F(\mathbf{r}(t))$ is used as a fitness function of GA [8]. To recognize a target in a dynamic image input by video rate, 33 [fps], the recognition system must have real-time nature, that is, the searching model must converge to the fish in the successively input raw images. An evolutionary recognition process for dynamic images have been realized by such method whose model-based matching by evolving process in GA is applied at least only one time to one raw image input successively by video rate. We named it as "1-Step GA" [7]. When the converging speed of the model to the target in the dynamic images should be faster than the swimming speed of the fish in the dynamic images, then the position indicated by the highest genes represent the fish's position in the successively input images in real-time. We have confirmed that the above time-variant optimization problem to solve $\mathbf{r}(t)$ maximizing $F(\mathbf{r}(t))$ could be solved by "1-Step GA". $\mathbf{r}(t) = [x(t), y(t)]^T$ represents the fish's position in Camera Frame whose center is set at the center of catching net, then $\mathbf{r}(t)$ means position deviation from net to Fish, means $\mathbf{r}(t) = \Delta\mathbf{r}(t)$ The desired hand velocity at the i -th control

period $\dot{\mathbf{r}}_d^i$ is calculated as

$$\dot{\mathbf{r}}_d^i = \mathbf{K}_P \Delta\mathbf{r}^i + \mathbf{K}_V (\Delta\mathbf{r}^i - \Delta\mathbf{r}^{i-1}) \quad (1)$$

where $\Delta\mathbf{r}^i$ denotes the servoing position error detected by 1-Step GA [7]. \mathbf{K}_P and \mathbf{K}_V given are positive definite matrix to determine PD gain. Now we add chaos items to (1) above, and we also need to redefine the meaning of $\dot{\mathbf{r}}_d^i$.

The simple PD servo control method given by (1) is modulated to combine a visual servoing and chaos net motion into the controller as follows,

$$\Delta\mathbf{r}^i = k_1 \cdot \Delta\mathbf{r}_{fish}^i + k_2 \cdot \Delta\mathbf{r}_{chaos}^i \quad (2)$$

Here $\Delta\mathbf{r}_{fish}^i = [\Delta x_{fish}^i \quad \Delta y_{fish}^i]$, is the tracking error of fish from the center of camera frame, and $\Delta\mathbf{r}_{chaos}^i = [\Delta x_{chaos}^i \quad \Delta y_{chaos}^i]$ denotes a chaotic oscillation in $x-y$ plane around the center camera frame. Therefore the hand motion pattern can be determined by the switch value k_1 and k_2 . $k_1 = 1$ and $k_2 = 0$ indicate visual servoing, and $k_1 = 0$ and $k_2 = 1$ indicate the net will track chaotic trajectory made by NNDE being explained later in this report. The desired joint variable $\dot{\mathbf{q}}_d$ is determined by inverse kinematics from $\dot{\mathbf{r}}_d$ by using the Jacobian matrix $\mathbf{J}(\mathbf{q})$, and is expressed by

$$\dot{\mathbf{q}}_d = \mathbf{J}^+(\mathbf{q})\dot{\mathbf{r}}_d \quad (3)$$

where $\mathbf{J}^+(\mathbf{q})$ is the pseudo inverse matrix of $\mathbf{J}(\mathbf{q})$. The robot used in this experimental system is a 7-Link manipulator, Mitsubishi Heavy Industries PA-10 robot.

III. PROBLEM OF FISH-CATCHING

In order to check the system reliability in tracking and catching process, we kept a procedure to catch a fish and release it immediately continuously for 30 minutes. We released 5 fishes (length is about 40[mm]) in the pool in advance, and once the fish got caught, it would be released to the same pool at once. The result of this experiment is shown in Fig.2, in which vertical axis represents the number of fishes caught in successive 5 minutes and horizontal axis represents the catching time. We had expected that the capturing operation would become smoother as time passing on consideration that the fish may get tired. But to our astonishment, the number of fishes been caught decreased gradually.

The reason of decreased catching number may lie in the fish learning ability. For example, the fish can learn how to run away around the net as shown in Fig.3(a) by circular swimming motion with about constant velocity, having made a steady state position error that the net cannot reach to the chasing fish. Or the fish can stay in the opposite corner against the net in the pool shown in Fig.3(b). And also, the fish can keep staying within the clearance between the edge of the pool and the net shown in Fig.3(c) where the net is inhibited to enter.

To solve these problems, and to achieve more intelligent fish catching systems, we thought chaos behavior of the net with many chaotic varieties can be a possible method to overcome those fishes' escaping intelligence, since huge variety of chaos trajectories seems to be unpredictable for the fish to adapt

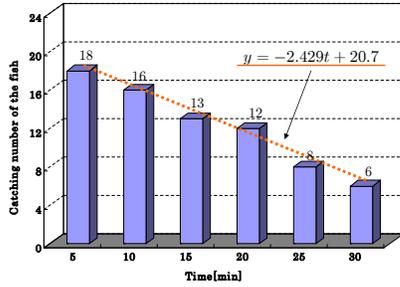
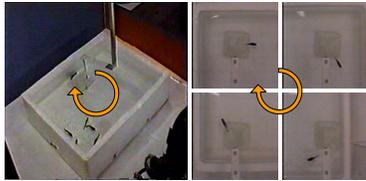


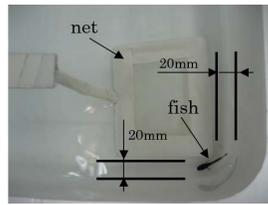
Fig. 2. Result of catching number



(a) Motion (1) of a fish



(b) Motion (2) of a fish



(c) Motion (3) of a fish

Fig. 3. Fish motion

them. This strategy to overcome fishes' adaptive intelligence is based on a hypothesis that unpredictability of the motion of the chasing net produced by plural chaos can made the fishes' learning logic confuse, getting the fish catching robot have made intelligence than the fishes'. Then we propose Neural-Network-Differential-Equation to generate chaos as many as possible.

IV. FISH INTELLIGENCE QUOTIENT

To evaluate numerically how fast the fish can learn to escape the net, we adapted Linear Least-Square approximation to the fish-catching decreasing tendency, resulting in $y = -2.429t + 20.7$ as shown in Fig.2. The decreasing coefficient -2.429 represents adapting or learning velocity of the fishes as a group when the fishes' intelligence is compared with robotic catching. We named the coefficient as "Fish's Intelligence Quotient"(FIQ). The larger minus value means high intelligence quotient of the fish, zero does equal, and plus does less intelligent than robot's. To overcome the fishes' intelligence, more intelligent robotic system needs to track and catch the fish effectively, in other words it comes to the problem on how to use the item Δr_{chaos}^i in (2) effectively to exceed the fish intelligence.

V. VALIDITY OF CHAOS

In 1982, some experiments revealed that mollusk neuron cells and plant cells have irregular excitement and show

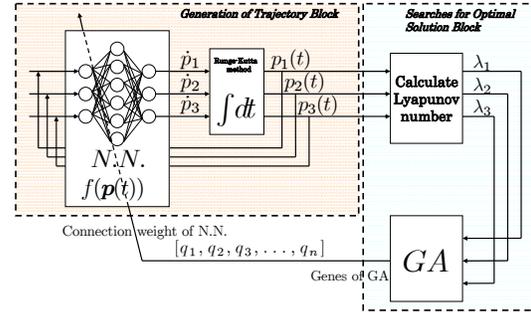


Fig. 4. Block diagram of Chaos Generation

chaotic nature if gave them periodic current stimulation. In addition, also chaotic response for periodic current stimulation had been clarified in the axon of the cuttlefish in 1984. From these studies, it became clear that the chaos is associated with biology. In the late 1980s, the relationship between chaos and function of the nervous system have been discussed. Mpitosos and colleagues examined the pattern of rhythmic firing of motor neurons of sea cucumber and showed that frequency variation of continuous discharge relates to the rhythm of the movement with chaotic behavior. Thus, chaos exists in biological behavior. It is decided whether the nerve cell of the organism is excited by a stimulation signal, and this is because it follows the theory of the chaos. Therefore, animal behavior and strategies can be estimated from point of chaos, and maybe apply to catch fish. There has been presented chaoses with a simplified model of Hodgkin-Huxley (H-H) model or BVP(Bonhoeffer-van der Pol) model. Using one chaos model to produce unpredictable motion add to catching-net behaviors seems to be effective, however we thought single chaos model is not adequate to overcome fishes' escaping idea since the fishes change their behavior continuously.

VI. NEURAL-NETWORK-DIFFERENTIAL-EQUATION

Lorenz and Rossler models renowned as chaos generation comprise three differential equations, producing three-dimensional chaotic trajectory in phase space. Since a Neural-Network(N.N.) has been proven to have an ability to approximate any non-linear functions with arbitrarily high accuracy, we thought it is straightforward to make a differential equation including N.N. so that it can generate plural chaoses by changing N.N.'s coefficients. We define next nonlinear differential equation including N.N. function $f(p(t))$ as

$$\dot{p}(t) = f(p(t)). \quad (4)$$

$p(t) = [p_1(t), p_2(t), p_3(t)]^T$ is state variable. The nonlinear function of $f(p(t))$ in (4) is constituted by N.N.'s connections, which is exhibited in left part of Fig.4 where the N.N. and integral function of outputs of N.N. and the feedback of the integrated value to the inputs of N.N. constitute nonlinear dynamical equation, (4). We call it as Neural-Network-Differential-Equation.

VII. CHAOS VERIFICATION METHODS

Since there have been no simple criterion to determine whether irregular oscillation is a chaos or not, we have to apply plural evaluations over the irregularities of trajectories produced by NNDE. The followings are criteria being used for judging the chaotic characters.

A. Lyapunov exponent

As one of criteria to evaluate a chaos' character of expansion in time domain, Lyapunov exponent expressed by the following equation is well known,

$$\lambda = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=0}^{N-1} \log |f'(x_i)|, \quad (5)$$

where positive value can represent that the irregular oscillation diverts from a standard trajectory, which expands like a function of e^{at} ($a > 0$).

B. Poincare section

The trajectories of the motion made by neural-network-based nonlinear function (4) is examined by using the Poincare section to verify further whether the resulted trajectories can be identified as chaos. Next, the Poincare section is explained. First of all, we examine an simple closed curve in three dimensions as shown in Fig5. The plane "A" that intersects with this closed trajectory pointed by "P" is defined as the Poincare section.

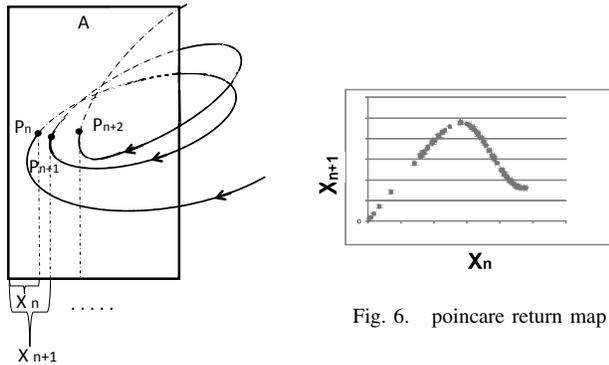


Fig. 5. Poincare section

The intersecting points are named as $P_n, P_{n+1}, P_{n+2}, \dots$, and corresponding x-axis position on A are $x_n, x_{n+1}, x_{n+2}, \dots$, which are all pointed to the Poincare Return Map as x_n, x_{n+1}, \dots as shown in Fig.6. With the Poincare return map of Fig.6 representing a shape of "Λ", the closed curve has the structure of stretching and folding. This structure is a basic character of chaos. Looking at the left half of Fig.6, we can see the inclination coefficient $dx_{n+1}/dx_n > 1$ and right half has $dx_{n+1}/dx_n < -1$, representing that left half has expansion and the other does contraction.

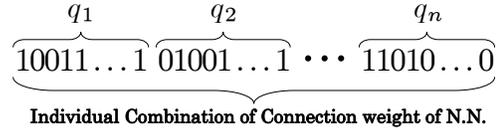


Fig. 7. Gene of GA

VIII. CHAOS GENERATE SYSTEM

Fig.4 represents the block diagram to find chaos by using GA and Lyapunov number. This GA is not 1-Step GA, described in Chapter II but used as a normal GA's procedure that evolves genes representing neural network coefficient's volume. The trajectory $p(t)$ in time domain obtained from Neural-Network-Differential-Equation is used for the calculation of Lyapunov number. Here, $L = [\lambda_1, \lambda_2, \lambda_3]^T$ is a Lyapunov number. Using this L for the evolution of GA, fitness function is defined as follows,

$$g = k_1 \cdot \lambda_1 - k_2 \cdot |\lambda_2| - k_3 \cdot \lambda_3. \quad (6)$$

This fitness function incorporated the chaotic property of the Lyapunov spectrum, which is one of factors to be essential for generating chaos trajectory. Here, because we discuss three-dimensional chaotic attractor in phase space, there are 3 Lyapunov numbers. The relationship between positive and negative Lyapunov spectrum is $(+, 0, -)$, which means resulted time trajectory of (4) may be thought to be chaos. Parentheses indicate the sign of the Lyapunov spectrum. In other words, λ_1 is positive, λ_2 is also positive or negative small values, λ_3 is negative case, the fitness function of (6) appears to have relatively large positive value when $\lambda_1 > 0$, $\lambda_2 \approx 0$, $\lambda_3 < 0$. In addition k_1, k_2 and k_3 are coefficients. The gene of GA is defined as shown in Fig.7, with connection weights of N.N. being $q = [q_1, q_2, \dots, q_n]^T$. In this report we adopted a network of $3 \times 6 \times 3$ as shown in Fig.4, then the number of connections and their coefficients is 48, i.e., $n=48$. The bit length of q is 16 bits. Because the gene is expressed in binary, converted to decimal and normalized into a range from 0 to 1. Then, generating a trajectory $p(t)$ based on a given gene having been determined by GA at one previous generation and calculating Lyapunov number, and evolving new generation of gene are repeated. This GA's evolution can find q to have a highest value of g defined by (6), that means possible chaos trajectory.

IX. VERIFICATION RESULTS

So far we have found four chaos patterns with different neural coefficients explored by GA mentioned in the previous section. We named them with a serial number as chaos 01~chaos 04. The followings are the introduction of those chaos with each individual character.

A. Chaos 01

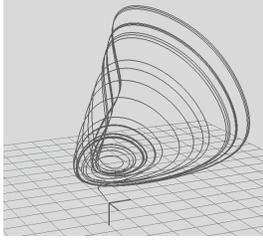


Fig. 8. Generated chaos trajectory 01

1) *Lyapunov number*: Lyapunov numbers are $\lambda_1 = 0.014585$, $\lambda_2 = -0.003314$ and $\lambda_3 = -0.165381$. These are corresponding to the Lyapunov spectrum of chaos, (+, 0, -).

2) *Sensitivity to initial value*: Two time-profile of trajectories with minutely different initial value are shown in Fig.9. The trajectories of $(x_1(t), y_1(t), z_1(t))$ are the results that originated from the initial values of $x_1(0) = 1.00, y_1(0) = 1.00, z_1(0) = 1.00$ and $(x_2(t), y_2(t), z_2(t))$ are from $x_2(0) = 1.01, y_2(0) = 1.01, z_2(0) = 1.01$. Trajectories of x_1 and x_2 are shown in Fig.9 .

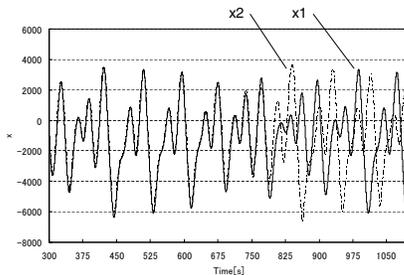


Fig. 9. Generated trajectory 01 of x (300[s] to 1100[s])

We can see from Fig.9 that the two trajectories with minute difference of initial values divert often about 800 seconds having passed, this means the slight different initial values make large separation with each other, indicating sensitivity of initial value, which is one of the character of chaos. As for y and z coordinates, they are similar, omitted to spare the space.

3) *Poincare return map*: Chaos 01's poincare return map is shown in Fig.10. One dimensional map can be seen in Fig.10, from which we can understand that the map represents expanding (left half of the Fig.10) and contracting (right half) that are essential characters to generate chaos.

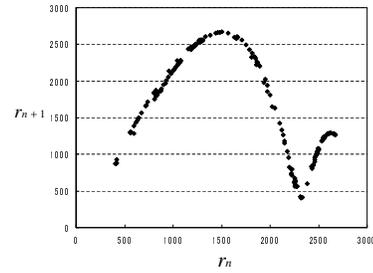


Fig. 10. Poincare return map of Chaos01

Therefore, the property of chaos 01 has been confirmed from the viewpoint of Lyapunov number, an sensitivity of initial value, and the Poincare return map.

B. Chaos 02

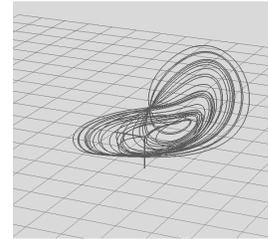


Fig. 11. Generated chaos Trajectory 02

We searched second chaos by similar produce like chaos 01. Up to now we have found other three chaos, 02, 03, 04, where Lyapunov numbers are listed in Table 1, including chaos 01 also.

TABLE I
LYAPUNOV NUMBER

	chaos01	chaos02	chaos03	chaos04
λ_1	0.014585	0.01919	0.015934	0.01208
λ_2	-0.003314	0.00733	-0.002172	-0.00143
λ_3	-0.165381	-0.10379	-0.123026	-0.075448

We confirmed all chaos trajectories have the Lyapunov spectrum of chaos, (+, 0, -).

X. SENSITIVITY OF NEURON'S WEIGHT

We have noticed weight coefficient of N.N. that generated chaos 03 are almost similar to chaos 04's. That is, only one weight coefficient is different, that is " q_1 " in Fig.12. We think " q_1 " is related to the generation of chaos. So we increased the weight slightly from "-1" and compare their trajectories.

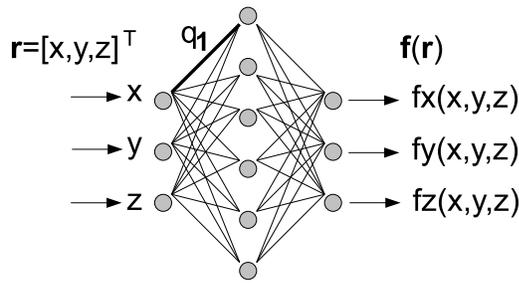


Fig. 12. Neural Network for nonlinear function generation

The range of q_1 is $-1.0 \leq q_1 \leq 1.0$ and q_1 is increased from -1.0 by 0.1. In the case of $-1.0 \leq q_1 \leq -0.4$, $0.1 \leq q_1 \leq 0.2$ and $0.8 \leq q_1 \leq 1.0$, we cannot consider the trajectory to be semi-periodic trajectories. On the other hand, the range of $-0.3 \leq q_1 \leq 0.0$ and $0.3 \leq q_1 \leq 0.7$ made chaos trajectories as shown in Fig.14 and 16. The other cases indicate that the trajectories are expanded to infinity. This result indicated that continuous changing of q_1 can make various chaos, stemming from continuity of real variables.

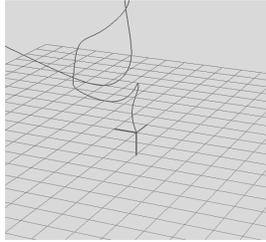


Fig. 13. Weight = -0.7

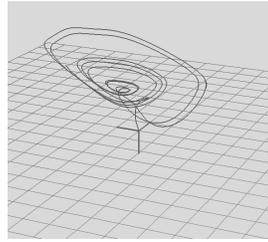


Fig. 14. Weight = 0.0

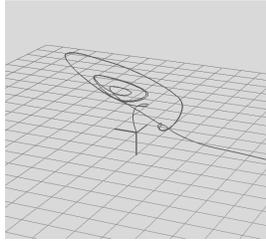


Fig. 15. Weight = 0.2

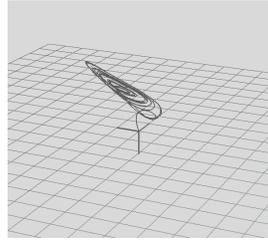


Fig. 16. Weight = 0.7

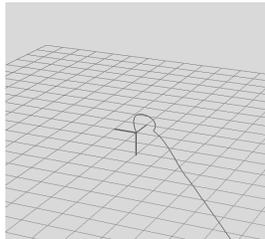


Fig. 17. Weight = 0.9

The Lyapunov number of each trajectory is shown as follows.

TABLE II
LYAPUNOV NUMBER

λ \ q_1	-0.7	0.0	0.2	0.7	0.9
λ_1	0.007314	0.016477	0.008571	0.004668	0.003386
λ_2	-0.004319	0.002719	-0.038939	0.011399	-0.046857
λ_3	-0.030184	-0.106123	-0.061084	-0.136963	-0.048269

XI. CONCLUSION

This paper proposed chaos generating system composed of Neural Network and GA's evolving ability to change the Neural-Network-Differential-Equation to be able to generate chaos. This chaos generating system has exploited the neural network's nature of approximation of any nonlinear function with any desired accuracy. A chaos motion can make fishes confuse, so we think it is effective for fish catching. We will utilize this chaos motion for overcoming fishes' escaping ability from chasing net for now, and confirm the effectiveness.

REFERENCES

- [1] R. Kelly, "Robust Asymptotically Stable Visual Servoing of Planar Robots", *IEEE Trans. Robot. Automat.*, vol.12, no.5, 1996, pp.759-766.
- [2] P.Y. Oh, and P.K. Allen, "Visual servoing by partitioning degrees of freedom", *IEEE Trans. Robot. Automat.*, vol.17, Feb.2001, pp.1-17.
- [3] T. Fukuda, and K. Shimojima, "Intelligent Control for Robotics", *Computational Intelligence*, 1995, pp.202-215.
- [4] M. Bohlen, "A robot in a cage-exploring interactions between animals and robots", *CIRA.*, 1999, pp.214-219.
- [5] Hee-Jun Park, Byung Kook Kim, Kye Young Lim, "measuring the machine intelligence quotient (MIQ) of human-machine cooperative systems", *IEEE Trans.*, vol.31, 2001, pp.89-96.
- [6] M. Minami, H. Suzuki, J. Agbanhan, T. Asakura, "Visual Servoing to Fish and Catching Using Global/Local GA Search" *Int. Conf. on Advanced Intelligent Mechatronics*, Proc., 2001, pp.183-188.
- [7] M. Minami, J. Agbanhan, and T. Asakura, "Manipulator Visual Servoing and Tracking of Fish using Genetic Algorithm", *Int. J. of Industrial Robot*, Vol.29, No.4, 1999, pp.278-289.
- [8] H. Suzuki, M. Minami, J. Agbanhan, "Fish Catching by Robot Using Gazing GA Visual Servoing", *Transaction of the Japan Society of Mechanical Engineers*, C-68-668, 2002, pp.1198-1206(in Japanese).
- [9] Jun Hirao and Mamoru Minami: "Intelligence Comparison between Fish and Robot using Chaos and Random", *International Conference on Advanced Intelligent Mechatronics*, 2008, pp.552-557.
- [10] K. Aihara, "Chaos in Neural System", , 1993 , pp.126-151(in Japanese).
- [11] R. FitzHugh, "Impulses and physiological states in theoretical models of nerve membrane", *Biophys.J.1*, 1961 , pp.445-466.
- [12] C. T. Lin and C. S. Lee, "Neural Fuzzy Systems", *Englewood Cliffs*, NJ:Prentice Hall PTR, 1996.
- [13] Limin Peng and Peng-Yung Woo, "Neural-Fuzzy Control System for Robotic Manipulators", *IEEE Control Systems Magazine*, 2002, pp.53-63.
- [14] M. Suzuki, "Evolutionary acquisition of complex behaviors through intelligent composite motion control", *Computational Intelligence in Robot. Automat. CIRA*, vol.27-30, 2005, pp.231-236.
- [15] R. Endo, J. Hirao and M. Minami, "Intelligent Chaos Fish-Catching Based on Neural-Network-Differential-Equation", *SICE Annual Conference*, 2010, pp.1957-1962.
- [16] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, "NUMERICAL RECIPES in C", *The Art Scientific Computing Second Edition*, 2002.
- [17] M. Minami, J. Hirao, "Intelligence comparison between fish and robot using chaos and random" *Int. Conf. on Advanced Intelligent Mechatronics*, Proc., 2008, pp.552-557.
- [18] Jun Hirao, Mamoru Minami, Yasushi Mae and Jingyu Gao, "Emergence of Robotic Intelligence by Chaos for Catching Fish", *SICE Annual Conference 2007*, pp. 969-975.