# Epsilon-Greedy Babbling -Exploitation and Exploration in Online Learning under Motor Babbling-

Chyon Hae Kim, Kanta Watanabe, Shun Nishide and Manabu Gouko

***Abstract*—Motor babbling allows an agent sampling trajectory data without a priori knowledge about self-body dynamics. We discuss about the efficiency of motor babbling through the example of drawing task. We propose an exploitation babbling and $\epsilon$-greedy babbling. In order to implement the proposed babblings, we developed dynamics learning tree (DLT). DLT is an online incremental learning algorithm that has constant calculation order O(1). The proposed exploitation babbling and $\epsilon$-greedy babbling improved the rate of effective data at 8 and 7 % from previous babbling respectively. $\epsilon$-greedy babbling converged its prediction error fastest among the three babblings. Using $\epsilon$-greedy babbling, a humanoid robot with wired flexible hand successfully drew a figure without a priori knowledge about the dynamics among self-body, pen, and pen tablet.**

## I. Introduction

Motor babbling is an important factor of cognitive development [1], [2]. Through motor babbling, infant tries learning the relationship between joint inputs and resulting motions. In this paper, we discuss exploration and exploitation in motor babbling through constructive approach [1].

Saegusa *et al.* developed the humanoid robot that learns reaching motions through artificial motor babbling [3]. Schillaci *et al.* modeled the dynamics of a robot arm [4]. Dawood *et al.* and Grimes *et al.* developed the robots that imitate motions [5], [6]. Especially, Mochizuki *et al.* developed the humanoid robot that learns the dynamics of its arm and a grabbed pen using Tani's artificial neural network (NN) [7], [8]. Nishide *et al.* improved its performance by setting stopping/pausing sequences in training data [9]. Nishide's theory is supported by several researches [10], [11]. In these researches, natural motor babbling has been modeled without considering exploitation.

On the improvement of motor babbling, Dearden *et al.* combined multiple forward models [12]. Baranes *et al.* proposed competence based intrinsic motivation that drives a robot to goals [13]. Rolf *et al.* introduced goal directed exploration [14]. Takanishi *et al.* proposed motor babbling in task space [15]. In these researches, motor babbling exploits previously acquired knowledge in order to sample new data effectively. However, the balance of exploration and exploitation has not been considered.

Chyon Hae Kim is Associate Prof. of the Dept. of Systems Innovation Engineering, Faculty of Science and Engineering, Iwate Univ., Japan. Chyon Hae Kim is Director (CTO) of AISing Ltd. Kanta Watanabe is with Info Tech Solution Co., Ltd., Japan. Shun Nishide is with Institute of Technology and Science, Tokushima Univ., Japan. Manabu Gouko is with Dept. of Mechanical Engineering and Intelligent Systems, Faculty of Engineering, Tohoku Gakuin Univ., Japan.

E-mail:tenkai@iwate-u.ac.jp (Chyon Hae Kim)

In the field of reinforcement learning, the balance of exploration and exploitation has been a popular issue. In the exploration process, an agent explores better action set through a random action selection strategy. In the exploitation process, an agent tries its best performance. In order to assure good learning performance, these processes must be mixed. In authors' insight, we must solve similar problem in motor babbling in order to obtain better learning performance.

In this paper, we propose an exploitation babbling and $\epsilon$-greedy babbling. $\epsilon$-greedy babbling combines exploration babbling and exploitation babbling seamlessly using the concept of $\epsilon$-greedy method [18].

The definitions of exploration and exploitation babblings are as follows:

1) Exploration babbling: random babbling that is open loop. No knowledge is reused.
2) Exploitation babbling: random babbling that reuses acquired body knowledge.

We validated the proposed babblings in drawing task, which has predecessors in engineering approach. Kudoh et al. developed a robot system that extracts 3D model of a real object using a stereo camera [19]. The extracted model is drawn by a robot arm. Kulvicius et al. proposed a motion-primitive jointing method. Motion primitives are modified to highly precise trajectories [20]. Yokoyama et al. developed a motion sensing robot system that captures professionals' drawing motion. The captured motion was regenerated by a robot arm [21]. Giorelli *et al.* applied neural network for a soft robot arm in order to handle its inverse statics without motor babbling [16]. Compared with motor babbling, these methods more depend on a priori task and/or platform knowledge. In order to relax the dependency, we tried this task using motor babbling.

As Luquet mentioned [22], drawing motion development is illustrated with following 5 phases: Scribbling (1-3 years), Fortuitous Realism (2-4 years), Failed Realism (3-7 years), Intellectual Realism (4-8 years), and Visual Realism (8+ years). In Scribbling, infant tries motor babbling in order to learn the relationship between its joint inputs and resulting drawn lines. In this phase, it does not concern about the meaning of the drawn lines. In Fortuitous Realism, children gradually understand the relationship between real objects and drawing. They enlarge their imitative motivation. In Failed Realism, children make a push to draw a real object. But, this trial fails because of insufficient capability. In Intellectual Realism, children draw imaginary things. In Visual Realism, perfect imitation is achieved. In this paper, we focuses on Scribbling and Fortuitous Realism. Although this developmental process is researched in [7], [9], [17], it is not completed. In the previous work, their babbling models do not take into account the balance of exploration and exploitation. This balance can be seen clearly in the development of human behavior. In this paper, we obtained an evidence that the balance in motor babbling enhances the learning efficiency of drawing task.

## II. $\epsilon$-GREEDY BABBLING

We can see a transition from random exploration to ordered exploration in the developmental process of human. After an infant grows to a child, his/her behavior is not just random already. For example, when a child is trying to open a bottle, the child may try random manipulation around the bottle. But, in this case, motor babbling is not completely random in joint input level, because he/she keeps his/her hand close to the bottle. For another example, when a child is trying to draw something on a canvas, the child may try to move a pen randomly on the canvas. As same as the first case, the child's joint input is not completely random, because the pen tip position is controlled on the canvas. In the above mentioned examples, the child is exploiting his/her knowledge in order to keep his/her hand/pen tip position around the bottle/canvas, while keeping the exploration (random babbling) of manipulation/drawing motion.

In order to realize the transition, we propose an exploitation babbling and $\epsilon$-greedy babbling. Fig. 1 shows the flow of an exploration babbling (naive motor babbling). In this babbling, joint input is decided at first. An action is taken using the input. Finally, the relationship between the joint input and the action result is learned. In previous work [7], [9], neural network was employed for the learning process. This action does not take any constraint from previously learned self-body knowledge.

Fig. 2 shows the proposed exploitation babbling (Proposed Babbling 1) that uses forward prediction. After deciding joint input, the action result of the joint input is predicted by an online incremental learning algorithm. According to the prediction result, taking the action or not is judged. If the prediction result is accepted, the joint input (action) is taken. Finally, the actual result is feedbacked in order to improve the prediction of the online incremental learning algorithm.

The learning algorithm of Proposed Babbling 1 should have online and incremental characteristics, in order to enhance its knowledge directly after observing a new action result, because the judgment process does not work correctly when the learning system returns poor prediction (especially when directly after starting its learning). On the other hand, starting from poor prediction is inevitable because the learning algorithm is not able to learn anything before observing some actual action results. For the implementation, we developed a new learning algorithm, dynamics learning tree (DLT).

The proposed exploitation babbling is able to keep constraints by rejecting the inputs that do not meet the constraints. But simultaneously, this motor babbling makes sticky actions around predictable body states (well learned actions). In the

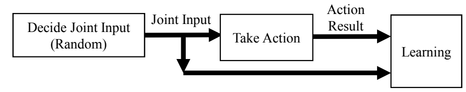


Fig. 1. Previous babbling (naive exploratory babbling)

In previous motor babbling, the action result of random joint input was learned by an offline learning algorithm.

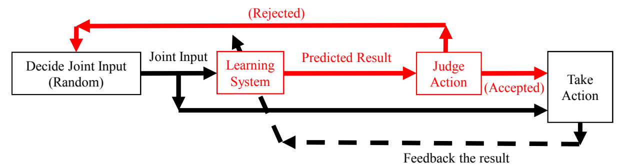


Fig. 2. Proposed babbling 1 (proposed exploitation babbling)

In Proposed Babbling 1, a result of an action is predicted using a learning algorithm before taking action. When the prediction satisfies the criteria of the judgment process, the action is taken. Directory after taking the action, the online incremental learning system is updated.

proposed $\epsilon$-greedy babbling (Proposed Babbling 2), one of exploration and exploitation processes is selected using parameter $\epsilon$ in order to balance these processes.

## III. DYNAMICS LEARNING TREE

In order to implement Proposed Babbling 1 and 2, we need to employ an online incremental learning algorithm. We developed Dynamics learning Tree (DLT) for the implementation.

DLT [27], [28] is the world first supervised learning algorithm that is based on smithing/hammering process. In smithing process, a mister forms arbitrary shaped metal by hitting it with a hammer. The features of smithing process are as follows:

1) The shape of metal monotonically and quickly converges to a target shape.
2) Every hammering is done independently (suitable for online calculation).

The characteristics of DLT is listed in Table III. Although DLT is not suitable to handle 'large volume input data like image data', it is able to perform high latencies (around several

TABLE I
COMPARISON OF LEANRNING ALGORITHMS

DLT: Dynamics Learning Tree, NN: Neural Network, NNA: the algorithms that use Nearest Neighbor Algorithm.
Det: Deterministic, Rand: Randomized.
*1: Depends on the number of iterations
*2: Depends on the algorithm
$A \geq B \geq C \geq D$

| Specifications | DLT | NN | NNA |
|---|---|---|---|
| Available Input Data Volume | C | A | B |
| Generalization | B | A | B |
| Learning Latency | A<br>O(1) | -<br>*1 | $\leq$A<br>$\geq$O(1) |
| Prediction Latency | A<br>O(1) | A<br>O(1) | B<br>$\geq$O(log(n)) |
| Parameter Adjustment | A | D | $\leq$A |
| Proof of Convergence | A | B | A |
| Online Learning | A | C | A |
| Incremental Learning | A | D | B |
| Computation | Det | Rand | *2 |

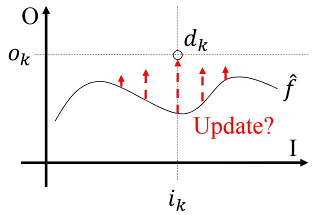


Fig. 3. Forming a function through smithing consept

$\mu$s) in both learning and prediction processes under constant calculation order O(1). Also, it is free from complex parameter adjustment, because its monotonic convergence is guaranteed. DLT supports fine online and incremental learning that handles every training record with same weight regardless of data order. Moreover, DLT's computation is deterministic. Thus, its computational process is governed by some rigid rules completely.

In previous researches, neural network was employed for the learning. However, supporting effective online incremental learning using NN is difficult because of the problem of catastrophic interference/forgetting [23], [24]. French et al. showed the forgetting problem (catastrophic forgetting) that occurs in online incremental learning of neural network [24]. Catastrophic forgetting is the phenomenon in that already learned knowledge on NN disappears rapidly by the online incremental learning of new training data. French et al. showed the actual example that a small weight change on a synapse erases a large amount of learned knowledge from NN. Ans et al. and Robins et al. proposed consolidation learning (CL), which applies BP method after mixing new and pseudo training data obtained from NN [25], [26]. CL is effective to avoid catastrophic forgetting. However, forgetting problem in online incremental learning is not completely avoided.

DLT realizes similar function update as smithing process. In Fig. 3, $\hat{f}$ is the function that is represented by a supervised learning algorithm, $d_k = i_k, o_k$ is new training data, $i_k$ is training input data, $o_k$ is training output data. We determine $f$ as a background function of the data.

In this update, $\hat{f}(i_k)$ must be updated to $o_k$. If $o_k$ does not include any noise the update rate should be 1. Also, the output of $\hat{f}$ around $i_k$ should be updated close to $o_k$ when we hypothesize that $f$ is continuous function against its I/O.

In the analogy with smithing process, we want to reshape the bulge $\hat{f}(i_k)$ of metal plate $\hat{f}$ to $d_k$. Thus, we hammer point $i_k$ until point $\hat{f}(i_k)$ moves to $d_k$ (update rate = 1). In smithing process, the metal that surrounds $(i_k, \hat{f}(i_k))$ gets the impact of the hammering. Thus, the metal takes replacement within the length between 0 and $d_k - \hat{f}(i_k)$ ($0 \leq$ update rate $\leq 1$).

In our smithing based update algorithm, we apply a monotonically decreasing update function that takes maximum (1) at the center ($i_k$). We call this update and update function as 'Smithing Update (SU)' and 'Smithing typed Update Function (SUF)' respectively. For DLT, we used a specific SUF that is determined by the input space representation of a tree structure.

Dynamics learning tree is a tree typed multi-layered learning algorithm. Fig. 4 shows the example of DLT with $N$ layer 2 sub-layers (dimensions) 2-ary tree. DLT's root node corresponds the whole region of $n$ dimensional input space. Each main layer has $n$ sub-layers (dimensions) with $d$-ary. The leaf nodes represent the divided sub-space as numbers 1-4 in Fig. 4.

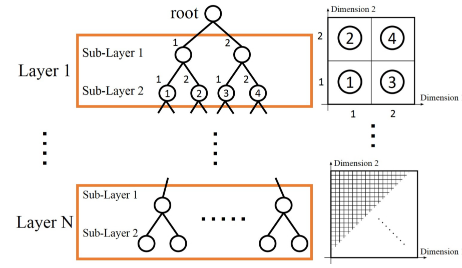


Fig. 4. DLT (N Layer 2 Dim. 2-ary).

The tree type data structure of DLT (left) represents the division of input space (right). In the Layer 1, assigned numbers 1-4 in the tree correspond the numbers 1-4 of the divisions of the input space. In this figure, all of the branches are illustrated, although these branches are incrementally created in the actual learning process.

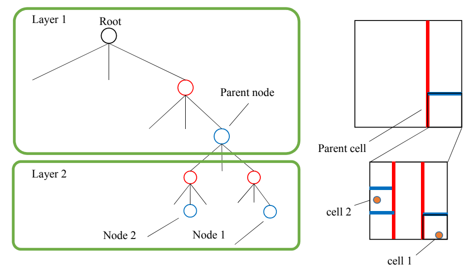


Fig. 5. Input space and DLT

This figure shows the data structure (left) and represented state space (right) of DLT when two data records are learned. The colors of nodes (left) correspond to those of partition (right). When a data record comes, DLT makes partitions around the input data of the record.

The example of its learning is shown in Fig. 5. When input data is given as the circle, a sub-space is created according to the input, so that the input data is in the sub-space. According to the position-on-the-tree of the node that represents the sub-space, A sequence of nodes are created from Root. Resulting tree of DLT is shown in Fig. 5 upper left. In these figures Cell 1 and Cell 2 are corresponding to Node 1 and 2 respectively. In every node of DLT, an average output vector was retained using following update functions:

$$\hat{O}_{Cell\ n} \leftarrow (N_{Cell\ n} \times \hat{O}_{Cell\ n} + O)/(N_{Cell\ n} + 1) \quad (1)$$

$$N_{Cell\ n} \leftarrow N_{Cell\ n} + 1 \quad (2)$$

where $N_{Cell\ n}$ is the learned number of output data in Cell $n$, $\hat{O}_{Cell\ n}$ is the average vector learned by Cell $n$, $O$ is training output vector. DLT cancels Gaussian noise around the training output vectors by the averaging process. These equations are characterized by online incremental update processes statistically. Also, its averaging process assures the same weight of learning between previously and currently learned training data. This update function is applied to a sequence of nodes from Root to the leaf node that corresponds to training output vector $O$. In this tree, the nodes that are close to the root (far cells from the input of the training data) retain larger $N_{Cell\ n}$ than those close to leaf. Thus, the total update of the above mentioned update for a cell results in SU. The results of the learning process do not depend on the order of training data

Fig. 6. Used humanoid robot and pen tablet

In order to validate the proposed babbling models, we used a humanoid robot, NAO, which has flexible fingers. This robot must learn a nonlinear relationship between its joint angles and pen tip position/pressure in order to draw a precise figure.

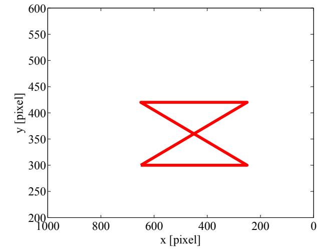


Fig. 7. Answer

We used this figure for the validation of the learning. After the robot learned the relationship between its joint angles and pen tip information, the robot tried tracking the figure using the pen. The drawing direction was the same among all experiments. The robot started from the upper left. Then, it tried moving the pen to upper right.

records.

In the prediction process of DLT, DLT returns the averaged output vector of the node that represents the partition of the input space in that input data is placed. Thus, when the number of learned training input data is sparse around the new input data, DLT's output is calculated using shallow nodes in the tree. On the other hand, when it is dense, deep nodes are used. Intuitively, DLT is controlling its output-complementation method against a new input according to the density of the training input data.

Compared with Gaussian process, DLT provides autonomous learning without kernel and hyper parameter tunings. Compared with regression tree, which requires offline statistics, DLT allows online and incremental learning.

## IV. Experiment

We conducted three experiments. In Experiment 1, the three babblings are compared in a drawing task in order to validate the effectiveness. In Experiment 2, Scribbling and Fortuitous Realism were examined using the three babblings. In Experiment 3, we validated the drawing capability of Proposed babbling 2 in another configuration.

### A. Settings

*1) Experiment 1:* We validated the proposed babblings using the 5 right arm joints of a humanoid robot NAO (Fig. 6). We obtained the position and pressure of the pen grabbed by NAO using a pen tablet.

We obtained pen pressure values from the pen tablet in the range of [0,1] (0: without pressure). The value is 0 when the pen is far from the tablet. On the pen tablet, 1 pixel is about 0.25 [mm].

DLT with 6 layer 5 dim. 3-ary was employed. The input and output of DLT were set at 5 joint angles and pen tip information respectively. The pen tip information is composed of $x, y$ coordinate on the pen tablet and pen tip pressure. We made two categories for the pen tip data in order to implement the judgment function of exploitation babbling. The first is effective data that was sampled when pen tip is on the tablet. The second is ineffective data that was sampled out of the tablet. When pen tip position was $y > 500$ or pen pressure is 0 or 1, the input was rejected. The data whose pen pressure is 1 was rejected exceptionally, because the pen was too strongly fitting on the tablet in this state. Also, for the initialization of the learning system of the proposed babbling models, 100 data records were sampled using previous babbling model. We set $\epsilon$ = 0.5.

*2) Experiment 2:* In order to realize Scribbling and Fortuitous Realism, we conducted an imitation experiment. A figure was copied by NAO.

We generated the data of the original figure using the technique of stopping/pausing sequences [9]. The predicted figures were given by the rehearsal of DLT that was trained in Experiment 2. The procedure of this rehearsal is as follows:

1) DLT got a target pen tip position from the original data
2) 1000 sets of random joint angles were inputted to DLT in order to predict resulting pen tip state.
3) A set of joint angles that give the most close pen tip position as the target was selected from the 1000 sets.
4) Go to 1) again.

In the drawing the joint angles that were calculated by the rehearsal were used as target joint angles of the robot. In the actual implementation, the rehearsal and drawing were done simultaneously. Thus, the prediction was included in the online control process of the robot.

*3) Experiment 3:* We examined $\epsilon$-greedy babbling (Proposed babbling 2) in another configuration that has the feedback of error ($E$) between an actual drawing point and the target figure. In this paper, we defined $E$ using Euclidean distance on the pen tablet.

In this experiment, we used two joints of the left arm of NAO, named LShoulderRoll and LElbowRoll. The random joint angles of the motor babbling was decided using the following equations:

$$\theta = \theta_C + \Delta\theta(\bar{E}) \tag{3}$$

where $\theta$ is random joint angle, $\theta_C$ is current joint angle, $\Delta\theta(\bar{E})$ is randomized feed amount that depends on average error $\bar{E}$.

We examined two formulations for $\Delta\theta(\bar{E})$.

$$\Delta\theta(\bar{E}) = \begin{cases} U(\beta\bar{E}^{\gamma}/\delta^{\gamma}) & (\bar{E} < \delta) \\ U(\beta) & (\bar{E} \geq \delta) \end{cases} \tag{4}$$

where $U(\circ)$ is uniform distribution between minus $\circ$ to $\circ$, $\beta$ is constant base width of the uniform distribution, $\delta$ is constant threshold of the base.

This formulation controls the magnitude of exploration in motor babbling. In this experiment, we used $\beta = 0.25$ rad,

TABLE II

RATE OF EFFECTIVE DATA

| Model | Sampled data | Effective data (ED) | Rate of ED[%] |
|---|---|---|---|
| Previous (Exploration) | 5000 | 4192 | 84 |
| Proposed 1 (Exploitation) | 5000 | 4614 | 92 |
| Proposed 2 ($\epsilon$-greedy) | 5000 | 4574 | 91 |

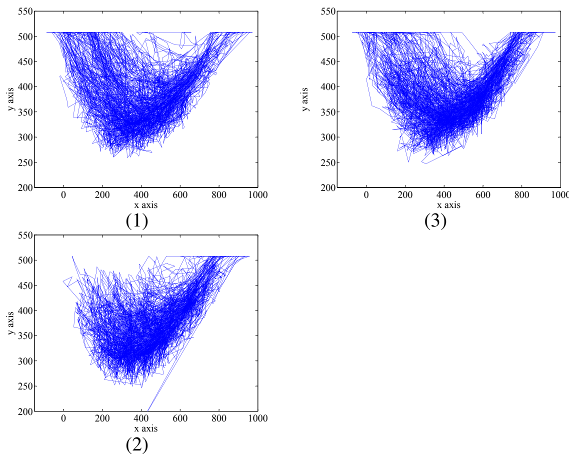


Fig. 8. Sampled pen tip positions

(1)–(3) show sampled pen tip positions of Previous babbling (exploration), Proposed babbling 1 (exploitation), and Proposed babbling 2 ($\epsilon$ - greedy) respectively. The humanoid robot NAO was placed on the top of the figures. It moved a pen using its right hand. The result data of Proposed Babbling 1 has a bias compared with the others. Proposed Babbling 1 has sticky tendency to repeat already known (trustable) actions.

$\delta = 20$ pixel. We used 100 error average for $\bar{E}$. We examined the effectiveness of $\gamma = 2$ using the procedure in Experiment 2.

## B. Results

*1) Experiment 1:* Proposed Babbling 1 and 2 sampled larger number of effective data compared with previous babbling (Table II).

Figs. 8 show the pen tip positions that were used for the data sampling. Proposed Babbling 1 generated the sampling positions with a bias. The region of distributed data in Fig. 8 (2) lacks upper left part compared with the others.

By using 1268 test data records that were sampled by previous babbling, the prediction error values of the three babbling models were validated. For the test data set, the data records that have pen pressure values in the range of (0,1) were used. Fig. III shows the prediction error. Proposed babbling 2 converged the fastest.

*2) Experiment 2:* Figs. 10 (1)-(3) show the original, predicted, and actually drawn figures of three babblings. Figs. 10 (4)-(6) show the error between 'original and prediction', 'prediction and drawn', and 'drawn and original' of them. Table III shows the average prediction error that was calculated from the data of Figs. 10 (4)-(6).

From Figs. 10 (4)-(6), the error between 'original and prediction' is not so large, but the others. Even if the learning models' prediction is on the original figures, their actual results have large error in some cases. Among the three models, Proposed Babbling 2 showed smallest error between drawn and original. This means that the prediction of Proposed Babbling 2 is more trustable than the others. As in Figs. 10

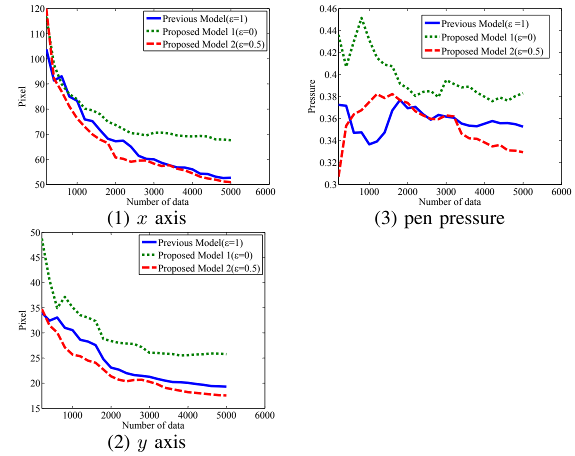


Fig. 9. Prediction error

Proposed Babbling 2 converged its error the fastest among the three babbling models.

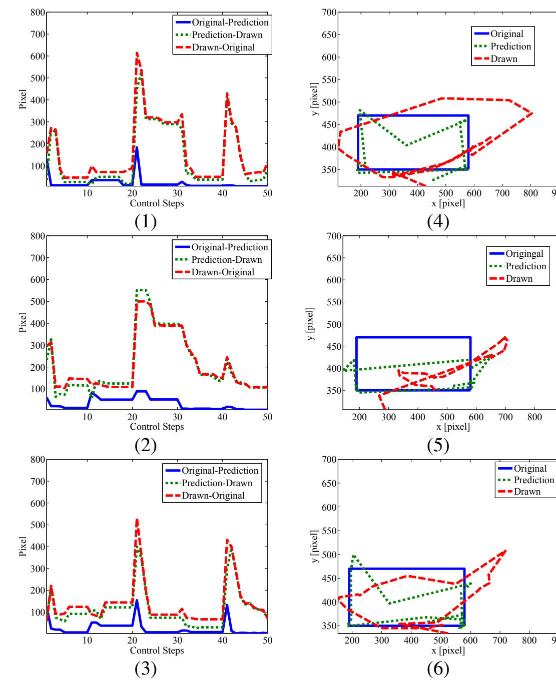


Fig. 10. Results of imitation

Figs. (1)-(3) show the error of Previous Babbling (Exploration), Proposed Babbling 1 (Exploitation), and Proposed Babbling 2 ($\epsilon$-greedy), respectively. Figs. (3)-(6) show the trajectories of these methods on the tablet respectively.

(1)-(3), the drawn lines of Proposed Babbling 2 is closest to the original.

*3) Experiment 3:* The final error between predicted positions and answer was 13.0 pixel in 20 trial average of the same task. Fig. 11 shows the drawing result after finishing the learning.

TABLE III

AVERAGE ERROR OF IMITATION TASK (SINGLE TRIAL)

| Model | Original & prediction | Prediction & drawn | Drawn & original |
|---|---|---|---|
| Previous (Exploration) | 19.49 | 131.52 | 157.63 |
| Proposed 1 (Exploitation) | 31.61 | 210.58 | 212.70 |
| Proposed 2 ($\epsilon$-greedy) | 23.00 | 114.19 | 137.01 |

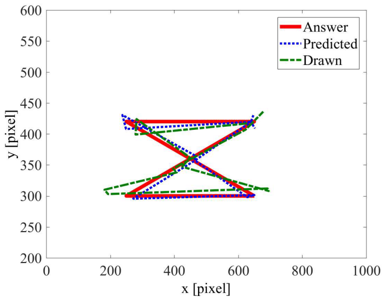


Fig. 11. Examination after finishing learning

After finishing 1000 data learning, motor babbling was stopped in order to examine the best performance of the current robot. The robot precisely drew a figure.

## V. Discussion

### A. Validation of exploitation babbling

From Table II, exploitation babbling (Proposed Babbling 1) sampled the largest number of the effective data in that pen tip position is on the tablet. The number of ineffective data was about the half of that of Previous Babbling. Proposed Babbling 1 successfully learned how to keep the constraint that was defined by the judgment process simultaneously with maintaining babbling.

### B. Validation of $\epsilon$-greedy babbling

From the results of Experiment 1 and 2, the learning performance of $\epsilon$-greedy babbling always exceeded the other babblings. This reason might be as follows. Previous Babbling does not sample data efficiently, because it does not keep constraint. Thus, it wastes the number of sampling trials to sample not required data that is out of the constraint. Proposed Babbling 1 samples data on the constraint. However, it sticks around the confidential pen tip positions that were already learned. As a result, Proposed Babbling 1 lacks its exploration capability, and makes lack of data like in Fig. 8 (2). Proposed Babbling 2 relaxes the sticky tendency of Proposed Babbling 1 by using Previous Babbling probabilistically. While Proposed Babbling 2 is performed, the pen tip position was almost in the constraint. Therefore, while Previous Babbling was performed, small exploration of a new pen tip position was examined around the constraint. Thus, in Proposed Babbling 2, the explored region does not expand too far from the constraint like Previous Babbling.

### C. Similarity with Scribbling and Fortuitous Realism

As in Experiment 2, Proposed Babbling 2 realized better imitation than the others. We could confirm the following similarity (1-4) with above mentioned Scribbling and Fortuitous Realism of Luquet [22]. In Scribbling and Fortuitous Realism, children learn how to move their body through drawing trials. This learning is characterized by online incremental improvement of their prediction for pen tip movement (1. online incremental learning). Also, before the learning, children do not know even how to keep constraint of the canvas (2. inexistence of pre-knowledge for the task). The constraint keeping and motion prediction are learned simultaneously (3. simultaneous learning for constraint and motion). Once knowledge for the prediction is learned, the knowledge is used for the imitation task (Fortuitous Realism) that is first experience without additional learning (4. acquired knowledge is applicable to a variety of tasks). Also, even in the first experience task, Proposed Babbling 2 can improve its prediction from the sampled data using online incremental learning process. Especially, the characteristics of 1 and 3 have not been realized in previous babbling models.

## VI. Conclusion and Future Work

We proposed two motor babblings, an exploitation babbling and $\epsilon$-greedy babbling. Both proposed babblings successfully sampled more number of effective training data than previous exploration babbling. The sampled training data of $\epsilon$-greedy babbling effectively improved the learning performance of DLT. Using $\epsilon$-greedy babbling, the drawing learning that was closer to Scribbling and Fortuitous Realism than previous researches was realized. The robot learned how to draw a figure without a priori knowledge about drawing dynamics using a specific configuration of $\epsilon$-greedy babbling. The theory of $\epsilon$-greedy babbling has much potential to be applied to a variety of tasks and a variety of robots than drawing.


## Acknowledgement

A part of this work was supported by JSPS KAKENHI Grant Numbers, 16H05877, 15K00363, 15K20850, 24119003, and SCAT Technology Research Foundation.


## References


[1] M. Asada, K. M. Dorman, H. Ishiguro, and Y. Kuniyoshi: "Cognitive developmental robotics as a new paradigm for the design of humanoid robots," *Robotics and Autonomous Systems*, 33 185–193, (2001).

[2] R. Yokoya, T. Ogata, J. Tani, K. Komatani and H. G. Okuno:"Experience-based imitation using RNNPB," *Advanced Robotics*, Vol. 21, No. 12, pp. 1351–1367, (2007).

[3] R. Saegusa, G. Metta, G. Sandini:"Active Learning for Multiple Sensorimotor Coordination Based on State Confidence," *IEEE/RSJ Inter. Conf. Robotics and Systems*, 2598–2603, (2009).

[4] G. Schillaci and V. V. Hafner:"Random Movement Strategies in Self-Exploration for a Humanoid Robot," *IEEE/ACM Inter. Conf. on Human Robot Interaction*, 245–246, (2011).

[5] F. Dawood and C. K. Loo:"Humanoid Behavior Learning through Visuomotor Association by Self-Imitation," *SCIS&ISIS*, 922–929, (2014).

[6] D. B. Grimes and R. P. N. Rao: "Learning Actions through Imitation and Exploration: Towards Humanoid Robots That Learns from Humans," Creating Brain-Like Intelligence, *Lect. Notes in Computer Science*, 5436 103–138, (2009).

[7] K. Mochizuki, S. Nishide, H. G. Okuno, T. Ogata: "Developmental Human-Robot Imitation Learning of Drawing with a Neuro Dynamical System," IEEE Inter. Conf. Systems, Man, and Cybernetics, pp. 2336–2341, (2013).

[8] Y. Yamashita and J. Tani: "Emergence of Functional Hierarchy in a Multiple Timescale Neural Network Model: a Humanoid Robot Experiment," *PLoS Comput. Biol.*, 4(11), e1000220, (2008).

[9] S. Nishide, K. Mochizuki, H. G. Okuno, and T. Ogata: "Insertion of Pause in Drawing from Babbling for Robot's Developmental Imitation Learning," *IEEE Inter. Conf. on Robotics and Automation*, 4785–4791, (2014).

[10] Y. Nagai, M. Asada, and K. Hosoda: "Learning for joint attention helped by functional development," *Advanced Robotics*, 20(10) 1165–1181, (2006).

[11] S. Nakaoka, A. Nakazawa, K. Yokoi, H. Hirukawa, and K. Ikeuchi: "Generating Whole Body Motions for a Biped Humanoid Robot from Captured Human Dances," *IEEE Inter. Conf. on Robotics and Automation*, 905–3910, (2003).

[12] A. Dearden and Y. Demiris: "Active learning of probabilistic forward models in visuo-motor development," Adaptation in Artificial and Biological Systems, (2006).

[13] A. Baranes and P.-Y. Oudeyer: "Active Learning of Inverse Models with Intrinsically Motivated Goal Exploration in Robots," Robotics and Autonomous Systems, 61(1), pp. 49-73, (2012).

[14] M. Rolf, J. J. Steil, and M. Gienger: "Goal Babbling Permits Direct Learning of Inverse Kinematics," IEEE Trans. on Autonomous Mental Development, 2(3), pp. 216–229, (2010).

[15] L. Jamone, L. Natale, K. Hashimoto, G. Sandini, and A. Takanishi: "Learning Task Space Control through Goal Directed Exploration," IEEE Inter. Conf. on Robotics and Biomimetics, pp. 702–778, (2011).

[16] M. Giorelli, F. Renda, M. Calisti, A. Arienti, G. Ferri, and C. Laschi: "Neural Network and Jacobian Method for Solving the Inverse Statics of a Cable-Driven Soft Arm With Nonconstant Curvature," IEEE Trans. on Robotics, 31(4), pp. 823–834, (2015).

[17] K. Watanabe, A. Numakura, S. Nishide, M. Gouko, and C. H. Kim: "Efficient body babbling for Robot's Drawing Motion," *IEEE Inter. Conf. on Mechatronics and Automation*, (2015).

[18] R. S. Sutton and A. G. Barto:"Reinforcement Learning: An Introduction," *MIT Press*, (1998).

[19] S. Kudoh, K. Ogawara, M. Ruchanurucks, and K. Ikeuchi: "Painting robot with multi-fingered hands and stereo vision," *Robotics and Autonomous Systems*, Vol. 57, No. 3, pp. 279–288, (2009).

[20] T. Kulvicius, K. Ning, M. Tamosiunaite, and F. W'org'otter, "Joining movement sequences: Modified dynamic movement primitives for robotics applications exemplified on handwriting," *IEEE Trans. on Robotics*, Vol. 28, Iss. 1, pp. 145–157, (2012).

[21] T. Yokoyama, S. Kudomi, T. Mitsui: "Developing and Implementation of Sensing Methods for Novel Sophisticated Robot Teaching -Facilitate The Reproduction of experienced Painter's arts-," *Research Report of Gifu Prefectural Research Institute of Information Technology*, Vol. 5, pp. 47–50, (2003), In Japanese.

[22] G. H. Luquet: "Le Dessin Enfantin," (1927).

[23] https://en.wikipedia.org/wiki/Catastrophic_interference

[24] R. M. French: "Catastrophic forgetting in connectionist networks," *Trends in Cognitive Science*, Vol. 3, No. 4, (1999).

[25] B. Ans and S. Rousset: "Avoiding catastrophic forgetting by coupling two reverberating neural networks," *Neuroscience*, Vol. 320, pp. 989–997, (1997).

[26] A. Robins and S. McCallum: "The consolidation of learning duaring sleep: comparing the pseudorehearsal and unlearning accounts," *Neural Networks*, (1999).

[27] K. Eto, Y. Kobayashi and C. H. Kim:"Vehicle Dynamics Modeling using FAD Learning," *Lect. Notes in Artificial Intelligence*, (2016).

[28] A. Numakura, S. Kato, K. Sato, T. Tomisawa, T. Miyoshi, T. Akashi and C. H. Kim: "FAD learning: Separate Learning for Three Accelerations -Learning for Dynamics of Boat through Motor Babbling-," *In Proc. of IEEE Inter. Conf. on Robotics and Automation*, pp. 5609–5614, (2016).