Achieving Different Levels of Adaptability for Human–Robot Collaboration Utilizing a Neuro-Dynamical System

Yuxi Li[1], Shingo Murata[1], Hiroaki Arie[1], Tetsuya Ogata[2], and Shigeki Sugano[1]

***Abstract*— Collaborative robots are expected to acquire adequate planning ability when achieving complex tasks such as cooking and cleaning with humans in dynamically changing daily-life environment. To fulfill adequate planning, three levels of adaptability—motion modification, action selection, and turn-taking—should be considered. This study demonstrates that a single hierarchically organized neuro-dynamical system called multiple timescale recurrent neural network (MTRNN) can achieve these levels of adaptability by utilizing the so-called multiple timescale property. The system is implemented in a humanoid robot and the robot is required to collaborate with a human partner by sharing a specific task under dynamically changing environment. Experimental results show that in both learned and unlearned situations, the robot can generate adequate behaviors against different situations, and the aforementioned levels of adaptability can be realized by a single system.**

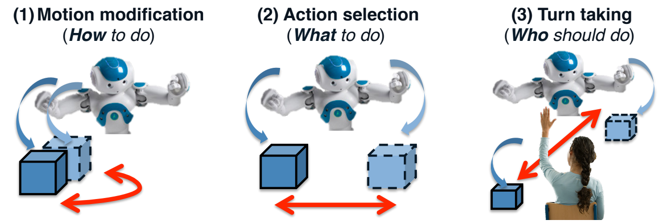

Fig. 1. Different levels of adaptability. (1) Motion modification requires agents to consider how to do one action. (2) Action selection requires agents to consider what action should be done. (3) Turn taking requires agents to consider who should do the action.

I. INTRODUCTION

Collaborative robots situated in daily-life environment are required to perform tasks with spatiotemporal complexity involving humans. The complexity in such human–robot collaboration consists of two aspects, temporal complexity of the task and spatial complexity of the environment. A typical example of task collaboration with spatiotemporal complexity is collaborative cooking. Consider a situation where a human and a robot are working in a shared space to accomplish a common final goal–cooking a soup. This goal consists of multiple subtasks (e.g. washing, cutting, and boiling) to be accomplished sequentially. Temporal complexity is the characteristic of tasks consisting of many sequential subtasks. The spatial complexity of an environment arises from variability and uncertainty in the shared environment, such as object positions and physical and interpersonal constrains from the collaborator [1]. Task collaboration requires mechanisms to maintain multi-step task sequences and adaptability to spatial variations through acquired experience.

Adequate planning is essential for collaborative robots to deal with both temporal and spatial variations during collaboration. When considering humanoid robots engaging in complex tasks with humans in home environment, adequate planning in the context of human–robot collaboration can be realized by three levels of adaptability shown in Fig. 1: (1) motion-level adaptability (motion modification), (2) action-level adaptability (action selection), and (3) role-level adaptability (turn-taking). Motion modification requires agents to solve "how to do" problems by modifying arm trajectories when small scale changes happen. Action selection requires agents to solve "what to do" problems by selecting the suitable arm to perform an action when middle scale changes around the body happen. Turn-taking requires agents to solve "who should do" problems by correctly understanding collaborator intentions and actions and taking their own turn at an appropriate time when large scale changes of role-switching between agents happen [2], [3].

Previous studies have investigated robotic adaptability to spatiotemporal complexity. Regarding (1) motion modification, in a study of a dynamical system with a Gaussian mixture model using a single-arm robot, Khansari et al. [4] showed that the system allows the robot to achieve a single-step task in different environments by modifying motion trajectories through a set of previous demonstrations. Another study conducted by Maeda et al. [5], demonstrated that a mixture model based on probabilistic movement primitives can realize motion modification in a task requiring multiple human–robot interactions. Regarding (2) action selection, Hawkins et al. [6] developed a probabilistic model and inference mechanism that permits a robot to infer current states and choose appropriate actions in human–robot collaboration tasks. In a study of a system based on dynamical movement primitives, Lioutikov et al. [7] enabled a dual-arm robot to achieve motion modification and action selection in multi-step bimanual manipulation tasks. Regarding (3) turn-taking, Calinon et al. [8] and Sheng et al. [9] focused on collaborative table-lifting tasks between a human and a robot. The robot was able to adapt its behavior to human

This work was supported in part by CREST, JST; a MEXT Grant-in-Aid for Scientific Research on Innovative Areas "Constructive Developmental Science" (24119003); a JSPS Grant-in-Aid for Scientific Research (S) (25220005); a JSPS Grant-in-Aid for Young Scientists (A) (16H05878); and the "Fundamental Study for Intelligent Machine to Coexist with Nature" program of the Research Institute for Science and Engineering, Waseda University, Japan.

[1] Yuxi Li, Shingo Murata, Hiroaki Arie, and Shigeki Sugano are with the Department of Modern Mechanical Engineering, Waseda University, Tokyo, Japan.

[2] Tetsuya Ogata is with the Department of Intermedia Art and Science, Waseda University, Tokyo, Japan.

subjects who changed their role (leader or follower) during the collaboration. A simulation study conducted by Awano et al. [10] showed that a robot can determine its action and timing during an object arrangement task in collaboration with a human.

To achieve collaborative tasks with spatiotemporal complexity, each agent should acquire comprehensive adaptability to variances. However, most past studies have only focused on one or two particular aspects of the aforementioned levels, instead of considering their comprehensive integration. In addition, several models were required to integrate levels of adaptability, and their seamless integration and communication is difficult [11]. We therefore consider the integration of different levels of adaptability into a single model.

A hierarchically organized neuro-dynamical system called a multiple timescale recurrent neural network (MTRNN) [12] is well-known for its ability to self-organize functional hierarchies by utilizing its multiple timescale properties. This ability is speculated to contribute to the acquisition of multilevel adaptability. Thanks to the multiple timescale property, MTRNNs can deal with sequential tasks.

Previous studies regarding robot learning with MTRNN have demonstrated its ability of motion modification [12], action selection and turn-taking [13]. However, as far as we know, no studies have succeeded in achieving turn-taking in which robots become a "leader" of interactions and in integrating these different levels of adaptability. We believe that MTRNN has a potential to realize these aspects and the present study tackles this issue. A humanoid robot implemented with MTRNN was required to collaborate with a human partner for sequential tasks among environmental changes. Experimental results show that the single system can successfully achieve multilevel adaptability during human–robot collaboration.

II. Computational Model

A. Overview of MTRNN

MTRNN is a derivation of the so-called continuous-time RNN (CTRNN), whose contextual neural units are modeled by a leaky-integrator neuron with a single time constant. As shown in Fig. 2, our MTRNN consists of two groups of context layers according to their timescale, which is determined by a set time constant. One is a fast context layer with a small time constant (i.e., a lower level) representing action primitives, whose activity changes quickly over the short term. The other is a slow context layer with a large time constant (i.e., a higher level) representing the sequence of the primitives, whose activity changes slowly over the long term. Thanks to the levels of information processing by the multiple timescale property, an MTRNN can self-organize its function hierarchy, allowing it to solve sequential tasks by combining primitives into sequences. To maintain task information for achieving a given task under different environmental situations, we introduce a static vector called the parametric bias (PB) [14] that is directly connected to the slow context layer and is utilized as the highest-level representation of task sequences. Note that unlike other studies utilizing the modification of PB value to achieve behavior adaptation, in this study, we gave constraints on the PB. To achieve adaptation under different situations, PB stayed static to maintain one shared task information under different situations during collaboration.

Adequate planning can be achieved through two information pathways in the system. These pathways integrate both PB information representing a task sequence and sensory inputs representing the current environmental situation. One is a top-down pathway starting from the highest-level representation of PB and ending at the lowest-level sensory predictions through the slow and fast context layers. The other is a bottom-up pathway starting from sensory inputs to the slow context layer through the fast context layer.

The following subsections describe the forward dynamics and the learning method of network parameters.

B. Forward Dynamics

The forward dynamics of the internal state of the ith neuron at time step t corresponding to the sth sequence $u_{t,i}^{(s)}$ is updated in accordance with

$$u_{t,i}^{(s)} = \begin{cases} u_{t-1,i}^{(s)} & (i \in I_P), \\ \left(1 - \frac{1}{\tau_i}\right) u_{t-1,i}^{(s)} + \frac{1}{\tau_i} \left(\sum_{j \in I_I} w_{ij} x_{t,j}^{(s)} \right. & \\ \left. + \sum_{j \in I_P} w_{ij} p_{t,j}^{(s)} + \sum_{j \in I_F \cup I_S} w_{ij} c_{t-1,j}^{(s)} + b_i \right) & (i \in I_F \cup I_S), \\ \sum_{j \in I_F} w_{ij} c_{t,j}^{(s)} + b_i & (i \in I_O), \end{cases} \tag{1}$$

where I_I, I_F, I_S, I_P, and I_O are the neuron index sets for the input, fast context, slow context, PB, and output units, τ_i is the time constant of the ith context unit (τ_{F} or τ_{S}), w_{ij} is the weight of the connection from the jth to the ith neuron, $c_{t,j}^{(s)}$ is the activation value of the jth context neuron, $x_{t,j}^{(s)}$ is the jth external input, and b_i is the bias of the ith neuron.

In this study, the weight w_{ij} was set to zero, which indicated disconnection, in a case where $i \in I_S \wedge j \in I_I$ and $i \in I_S \wedge j \in I_P$ for the connection constrains. Note that not only the multiple timescale property, but also the connection constrains on the information flow are essential for self-organizing the functional hierarchy [15]. From (1), the PB units can be regarded as a particular case of the context neurons, whose time constant is a infinite value. Certain values were set to the PB units in order to represent different task sequence. The value of the initial state $u_{0,i}^{(s)}$ of the fast and slow context neurons were set to zero, which indicated a neutral state regardless of the sequence s.

The respective activation values of the PB unit $p_{t,i}^{(s)}$, the context unit $c_{t,i}^{(s)}$, and the output unit $y_{t,i}^{(s)}$ are calculated by

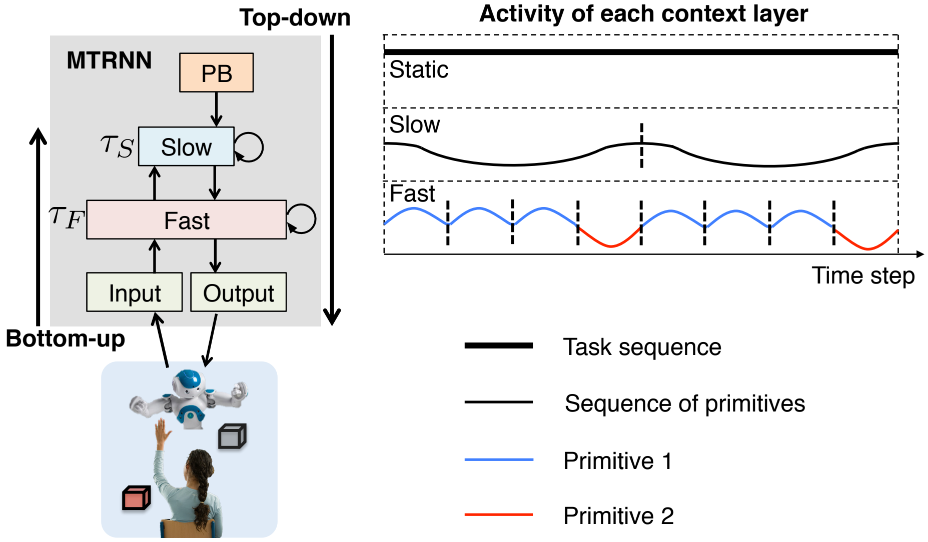

Fig. 2. System structure. MTRNN consists of two groups of context layers according to their timescale. One is a fast context layer (i.e., a lower level) representing action primitives, whose activity changes quickly over the short term. The other is a slow context layer (i.e., a higher level) representing the sequence of the primitives, whose activity changes slowly over the long term. A static vector called the PB units is connected with the slow context layer and is utilized as the highest-level representation of task sequences.

using the nonlinear function $\tanh(\cdot)$ as follows:

$$p_{t,i}^{(s)} = \tanh(u_{t,i}^{(s)}) \qquad (i \in I_P), \tag{2}$$

$$c_{t,i}^{(s)} = \tanh(u_{t,i}^{(s)}) \qquad (i \in I_F \cup I_S), \tag{3}$$

$$y_{t,i}^{(s)} = \tanh(u_{t,i}^{(s)}) \qquad (i \in I_O). \tag{4}$$

C. Learning Method

The network training (optimization of parameters) of MTRNN is based on the mean squared error minimization using the gradient decent method. Suppose that S target sequences are given, each of which consists of $\hat{Y}^{(s)} = (\hat{\boldsymbol{y}}_t^{(s)})_{t=1}^{T^{(s)}}$, where s is the index of the sequences and the $T^{(s)}$ is the length of the sequence. In this situation, the error function E is given by

$$E = \frac{1}{2} \sum_{t=1}^{T^{(s)}} \sum_{i \in I_O} \sum_{s \in I_S} (y_{t,i}^{(s)} - \hat{y}_{t,i}^{(s)})^2, \tag{5}$$

where $\hat{y}_{t,i}^{(s)}$ is the ith value (training data) at time step t of the sth target sequence $\hat{Y}^{(s)}$. The network parameters including weights w_{ij} and biases b_i are optimized to minimize the mean squared error in an off-line manner by utilizing the gradient decent method with back-propagation through time (BPTT).

III. Experimental Setup

A. Task Design

To achieve complex collaborative tasks in an actual environment, the fundamental aspects of human–robot task collaboration should be investigated first. By considering the temporal and spatial characteristics of complex task collaboration, we simplified actual collaborative tasks (such as cooking meals) to a sequential task collaboration of bell hitting between a human and a humanoid robot (NAO; Aldebaran) under dynamically changing environment. Note that both the human and the robot were set as bell hitter (executor).

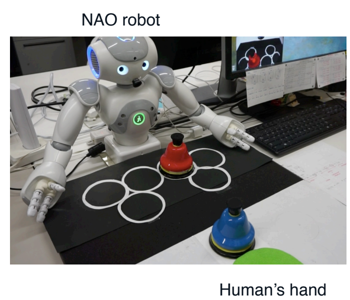

Fig. 3. Experiment environment. One NAO robot, two bells with blue and red colors and one green glove for the human hand were used.

As shown in Fig. 3, the experiment utilized a red bell, a blue bell, and a green glove for the human hand use. One bell was located on the left or right side in front of the robot, and the other was located in front of the human. Hereinafter, these positional relationships between the two bells are called cases, and the total number of the cases were four (two bell colors (blue, red) $\times$ two sides of robot (left, right)). In each case, a bell was placed on one of three fixed positions (bell position 1, bell position 2, and bell position 3) around one hand of the robot, and the other on a fixed position in front of the human (Fig. 4). Two collaborative task sequences were performed, repeating a BBBR (B: hit blue bell; R: hit red bell) sequence, and repeating a RRRB sequence.

B. Training data

For training data, we first guided the robot's arms to hit bells and defined 6 primitive movements each of which was for the six bell positions (three for the left and three for

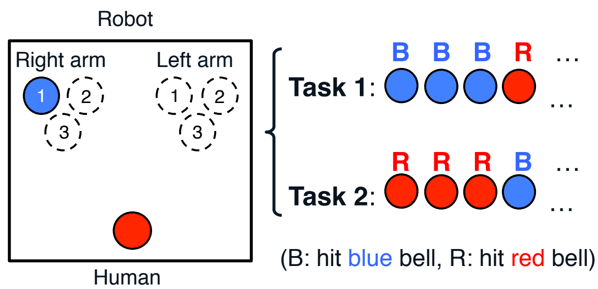

Fig. 4. Bell positions and task sequences. Regarding bell positions, one bell was placed in three of the fixed positions (bell position 1, bell position 2, and bell position 3) around one hand of the robot, and the other bell was located in a fixed position in front of the human. Two collaborative bell hitting task sequences were designed in experiments. One is BBBR task: repeating a BBBR (B: hit blue bell, R: hit red bell) sequence. The other is RRRB task: repeating a RRRB sequence.

the right side of the robot). During task collaboration with the human, one of these defined primitives was called for the robot's action generation. Data for two task sequences (BBBR task and RRRB task) in the aforementioned four cases with three considered bell positions around one hand of the robot were collected three times. Therefore the whole dataset consisted of $2 \times 4 \times 3 \times 3 = 72$ training data.

For each robot arm, four degrees of freedom including shoulder pitch, shoulder roll, elbow yaw, and elbow roll were considered. The centers of gravity (x position and y position of the colored area from an image) and the area ratios (can be regarded as distance or depth) with respect to red, green, and blue area were extracted from an image captured by a camera (originally mounted on the robot) for visual information. Both the vision and arm joint angle values in training data were scaled between -0.8 and 0.8. An example of training data for the BBBR task sequence collaboration with the blue bell located in the bell position 1 on the robot's right side is shown in Fig. 5. Note that we did not include any switching among situations or bell positions in each training data.

C. Parameter setting

MTRNN training was conducted offline using all collected data. The number of the fast context, slow context, and PB units were, respectively, $N_F = 100$, $N_S = 10$, and $N_P = 2$. The time constants for the fast and slow context units were $\tau_F = 2$ and $\tau_S = 50$. The parameters were optimized for 500,000 times.

D. Testing method

Three levels of adaptability were tested during one shared task sequence (BBBR task or RRRB task) collaboration between a human and the robot. For (1) motion modification, different situations were considered by slightly changing bell positions around one of the robot's hands. The robot was expected to adapt to such small scale changes by modifying its arm joint trajectories according to the new position. For (2) action selection, different situations were considered by changing bell positions between both of the robot's hands. The robot was expected to adapt to such middle scale

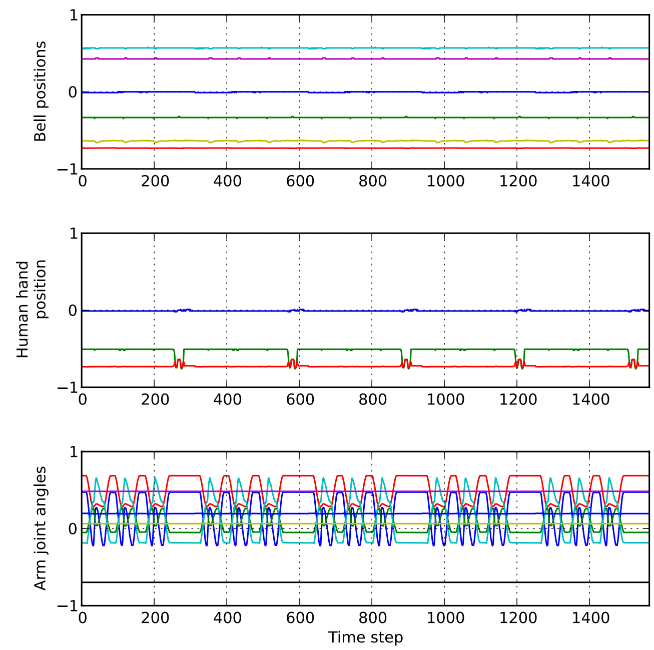

Fig. 5. An example of training data. Here, the BBBR task sequence collaboration with the blue bell located in the bell position 1 on the robot's right side is shown. The top row shows the x, y positions and the colored area ratios of the blue and red bells. Note that there are no situational changes in training data. The middle row shows the corresponding information of the green glove for the human hand use. The bottom row shows the arm joint angles of both left and right arms of the robot.

changes around its arms by selecting the suitable arm for performing the action. For (3) turn-taking, different situations were considered by changing bell positions between the human and the robot. The robot was expected to adapt to such large scale changes between the agents by correctly taking its own turn and acting on the bell.

IV. Results

We implemented the trained MTRNN in the NAO robot for collaboration with the human in an actual environment, and tested the three levels of adaptability by changing the situations as described above. In both BBBR task and RRRB task collaboration, the robot was able to reproduce learned behaviors for all learned situations. For the situational changes among learned and unlearned cases with novel bell positions, the robot also successfully generated adequate behaviors against situational changes through learned experience.

Figure 6 shows an example of the testing results for system adaptability during the BBBR task sequence collaboration in a dynamically changing environment with the human. Here, three conditions were tested. The first condition tested motion modification. The robot first hit the blue bell on its right side three times and then the human hit the red bell once. During this action generation, a third agent changed the bell position on the robot's side from the bell position 1 (R_1) to 2 (R_2), then to a novel position (R_{123}: the center of the three learned positions). The second condition tested action selection, by the third agent moving the bell from the robot's right side to its left (L_3). The third condition tested turn taking, by the third agent switching the blue and red bells between the robot and the human. In this case, the robot should wait for its turn before hitting the red bell.

Figure 7 (a) shows the bell positions (first panel), human hand position (second panel), arm joint predictions of the robot (third panel), PB, slow context, and fast context activations (fourth, fifth, and sixth, respectively) during the aforementioned BBBR task testing. In order to compare the difference of the context activation with a different task collaboration, the RRRB task collaboration under the same environmental setting is shown in Fig. 7 (b). In the first condition, with the slight changes of the bell position in the robot's vision, the motion (arm joint angle prediction) was slightly modified. In the second condition, the robot successfully adapted to repositioning of the bell position changing from the right side to the left side by correctly selecting the left arm to perform the action. In the third condition, the robot successfully took turns. After switching the blue and red between the robot and human, the robot waited for the ending of the human's action of hitting bell three times, then took its turn to hit on the bell in time.

As shown in Fig. 7 (a) and (b), when the robot generated same or similar actions such as hitting a bell with the right arm and hitting a bell with the left arm, similar patterns can be observed in the fast context layer, whose activation changes rapidly. In contrast, in the slow context layer, the activation changes gradually. The activation for the same task sequence collaboration under the different situations is different. Furthermore, The activation for the different (BBBR and RRRB) task sequence collaboration under the same environment setting is also different. Regarding the activation of the PB units, it remains static throughout the testing process. Changes in the slow context layer exhibit a longer period than those in the fast context layer. The slow context seems to have the ability to count the number of bell hits during one BBBR or RRRB collaboration. Maintaining the same value in PB units, which cannot be represented in the other two context layers, is similar to the ability to maintain the BBBR or RRRB task sequence during collaboration in a dynamically changing environment.

In each condition tested above, the two information pathways of the system, namely, top-down task information represented by the PB and bottom-up environmental information represented by the visual input, played an important role for achieving different levels of adaptability during the task collaboration. By receiving the environmental changes from the bottom-up pathway and the given task information from the top-down pathway, MTRNN can generate output against changes based on learned experience. In the first condition, the robot's motion (arm joint angle prediction) was slightly modified by receiving the changes of the bell position around one of the robot's hands as visual input. In the second condition, the robot's action was selected by receiving the change of the bell position (the left or right side of the robot). In the last condition, the turn-taking was realized by receiving the changes of the bell position (the robot or the human side).

V. Conclusion

In this study, we applied a hierarchically organized neuro-dynamical system called MTRNN to achieve a human–robot collaboration. The integration of motion modification, action selection, and turn-taking was thus successfully achieved within a single system thanks to its top-down and bottom-up information pathways.

We trained MTRNN with all considered situations. In future work, instead of using all situations in the training phase, we will attempt using only some situations for learning, and the others for testing. In the testing phase, we will expect the robot to generalize trained situations to new untrained situations. In order to achieve more real task collaboration, using color-based centers of gravity to detect only object positions and human movements is not enough because other visual information like the shape of the objects is also important for object manipulation. Therefore, instead of using color-based centers of gravity, we will consider applying deep learning approach [16] for extracting visual features from raw images.

References

[1] G. Pezzulo and H. Dindo, "What should I do next? Using shared representations to solve interaction problems," *Experimental Brain Research*, vol. 211, no. 3-4, pp. 613–630, 2011.

[2] N. Jarrasse, V. Sanguineti, and E. Burdet, "Slaves no longer: review on role assignment for human-robot joint motor action," *Adaptive Behavior*, vol. 22, no. 1, pp. 70–82, 2013.

[3] G. Pezzulo, "Shared Representations as Coordination Tools for Interaction," *Review of Philosophy and Psychology*, vol. 2, no. 2, pp. 303–333, 2011.

[4] M. Khansari-Zadeh and A. Billard, "Learning Stable Non-Linear Dynamical Systems with Gaussian Mixture Models," *Transactions on Robotics*, vol. 27, no. 5, pp. 943—-957, 2011.

[5] G. J. Maeda, G. Neumann, M. Ewerton, R. Lioutikov, O. Kroemer, and J. Peters, "Probabilistic movement primitives for coordination of multiple human-robot collaborative tasks," *Autonomous Robots*, mar 2016.

[6] K. Hawkins, N. Vo, S. Bansal, and A. Bobick, "Probabilistic Human Action Prediction and Wait-sensitive Planning for Responsive Human-robot Collaboration," *International Conference on Humanoid Robots*, pp. 499 – 506, 2013.

[7] R. Lioutikov, O. Kroemer, G. Maeda, and J. Peters, "Learning manipulation by sequencing motor primitives with a Two-Armed robot," *Advances in Intelligent Systems and Computing*, vol. 302, pp. 1601–1611, 2016.

[8] S. Calinon, P. Evrard, E. Gribovskaya, A. Billard, and A. Kheddar, "Learning collaborative manipulation tasks by demonstration using a haptic interface," *International Conference on Advanced Robotics*, pp. 1–6, 2009.

[9] W. Sheng, A. Thobbi, and Y. Gu, "An Integrated Framework for Human-Robot Collaborative Manipulation." *IEEE transactions on cybernetics*, vol. 45, no. 10, pp. 2030–2041, 2014.

[10] H. Awano, T. Ogata, S. Nishide, T. Takahashi, K. Komatani, and H. G. Okuno, "Human-robot cooperation in arrangement of objects using confidence measure of neuro-dynamical system," *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, pp. 2533–2538, 2010.

[11] H. Chong, A. Tan, and G. Ng, "Integrated cognitive architectures: a survey," *Artificial Intelligence Review*, vol. 28, no. 2, pp. 103–130, aug 2007.

[12] Y. Yamashita and J. Tani, "Emergence of functional hierarchy in a multiple timescale neural network model: A humanoid robot experiment," *PLoS Computational Biology*, vol. 4, no. 11, p. e1000220, nov 2008.

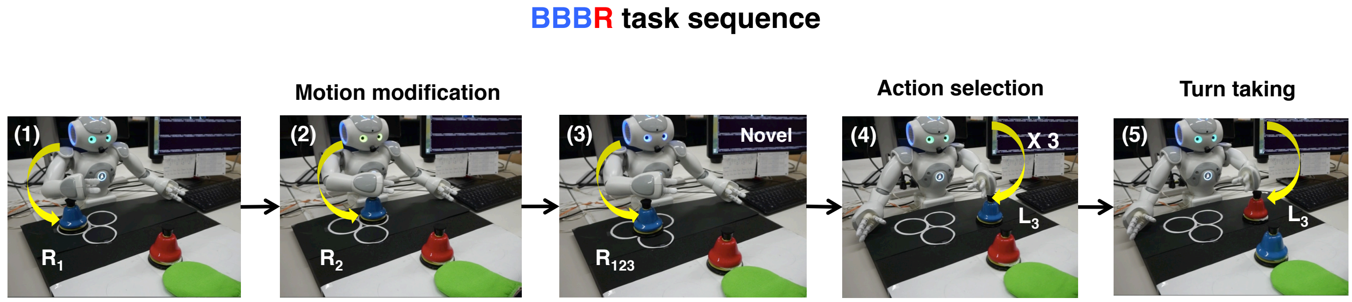

Fig. 6. Robot arm actions in the BBBR task collaboration. Here, the BBBR task sequence collaboration under a dynamically changing environment was tested. The robot generated five different actions. (1) The robot hit the blue bell located in the position 1 (R_1) using the right hand. (2) The robot hit the blue bell located in the position 2 (R_2) using the right hand. (3) The robot hit the blue bell located in a novel position (R_{123}: the center of the positions 1, 2, and 3) using the right hand. (4) The robot hit the blue bell located in the bell position 3 (L_3) using the left hand three times. (5) The robot hit the red bell located in the bell position 3 (L_3) using the left hand.

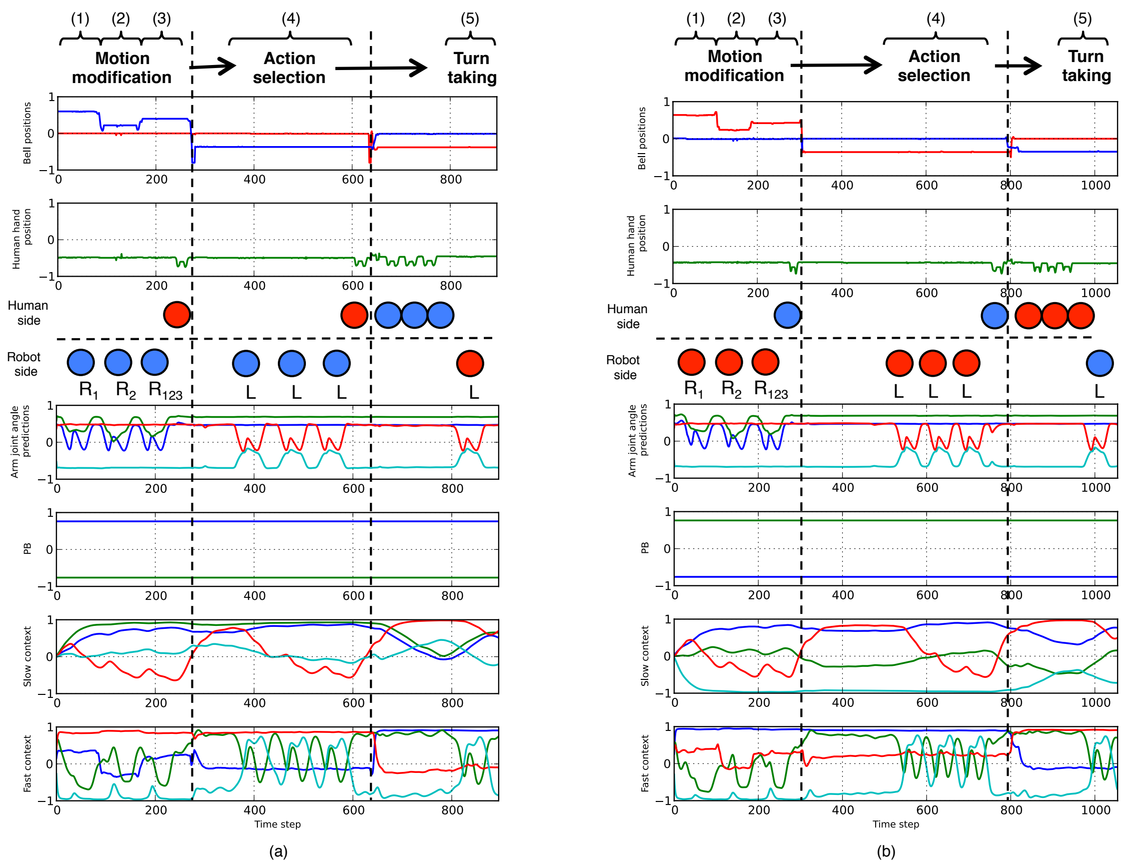

Fig. 7. Time-series data of visual inputs, arm joint predictions, and context activations in BBBR and RRRB task sequence collaborations. The first and second panels show the bell positions (blue: x position of blue bell, red: x position of red bell) and the human hand position (green: y position of green glove), respectively. The third panels show four (blue: right shoulder pitch, green: right elbow yaw, red: left shoulder pitch, and cyan: left elbow yaw) of eight dimensional arm joint predictions. The bell positions are also illustrated between the second and third panels. The fourth, fifth, sixth panels show PB (2 units), slow context (4 of 10 units), and fast context (4 of 100 units) activations, respectively. In both BBBR and RRRB task testing, the testing positions of bells were same. In the first case, motion modification was tested. In the second case, action selection was tested. In the third case, turn taking was tested.

[13] T. Yamada, S. Murata, H. Arie, and T. Ogata, "Attractor Representations of Language–behavior Structure in a Recurrent Neural Network for Human–robot Interaction," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'15)*, vol. 10, no. July, pp. 4179–4184, 2015.

[14] J. Tani, "Learning to generate articulated behavior through the bottom-up and the top-down interaction processes," *Neural Networks*, vol. 16, no. 1, pp. 11–23, 2003.

[15] S. Murata, Y. Yamashita, H. Arie, T. Ogata, S. Sugano, and J. Tani, "Learning to Perceive the World as Probabilistic or Deterministic via Interaction With Others: A Neuro-Robotics Experiment," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–31, 2015.

[16] K. Noda, H. Arie, Y. Suga, and T. Ogata, "Multimodal integration learning of robot behavior using deep neural networks," *Robotics and Autonomous Systems*, vol. 62, no. 6, pp. 721–736, jun 2014.