Learning to Perceive the World as Probabilistic or Deterministic via Interaction with Others: A Neuro-Robotics Experiment

Shingo Murata, Yuichi Yamashita, Hiroaki Arie,
Tetsuya Ogata, Shigeki Sugano, and Jun Tani *

Abstract

We suggest that different behavior generation schemes, such as sensory reflex behavior and intentional proactive behavior, can be developed by a newly proposed dynamic neural network model, named as stochastic multiple timescale recurrent neural network (S-MTRNN). The model learns to predict subsequent sensory inputs, generating both their means and their uncertainty levels in terms of variance (or inverse precision) by utilizing its multiple timescale property. This model was employed in robotics learning experiments in which one robot controlled by the S-MTRNN was required to interact with another robot under the condition of uncertainty about the other's behavior. The experimental results show that self-organized and sensory reflex behavior—based on probabilistic prediction—emerges when learning proceeds without a precise specification of initial conditions. In contrast, intentional proactive behavior with deterministic predictions emerges when precise initial conditions are available. The results also showed that, in situations where unanticipated behavior of the other robot was perceived, the behavioral context was revised adequately by adaptation of the internal neural dynamics to respond to sensory inputs during sensory reflex behavior generation. On the other hand, during intentional proactive behavior generation, an error regression scheme by which the internal neural activity was modified in the direction of minimizing prediction errors was needed for adequately revising the behavioral context. These results indicate that two different ways of treating uncertainty about perceptual events in learning, namely, probabilistic modeling and deterministic modeling, contribute to the development of different dynamic neuronal structures governing the two types of behavior generation schemes.

*This work was supported in part by the "Information Environment and Humans," JST PRESTO; Grant-in-Aid for Scientific Research on Innovative Areas "Constructive Developmental Science" (24119003), MEXT; Grant-in-Aid for Scientific Research (C) (25330301), JSPS; Grant-in-Aid for Scientific Research (S) (25220005), JSPS; "Fundamental Study for Intelligent Machine to Coexist with Nature," RISE, Waseda University, Japan. J. Tani was supported by grants from the Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore (AH/OCL/1082/0111/I2R) and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2014R1A2A2A01005491).

S. Murata and S. Sugano are with the Department of Modern Mechanical Engineering, Waseda University, Tokyo, Japan.

Y. Yamashita is with the Department of Functional Brain Research, National Center of Neurology and Psychiatry, Tokyo, Japan; and also with the Cognition and Behavior Joint Research Laboratory, RIKEN Brain Science Institute, Saitama, Japan.

H. Arie and T. Ogata are with the Department of Intermedia Art and Science, Waseda University, Tokyo, Japan.

J. Tani is with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea (e-mail: tani1216jp@gmail.com).

Correspondence should be sent to J. Tani.

1 Introduction

Our surrounding environment is perceived to change probabilistically. In such a fluctuating environment cognitive agents, including both humans and artifacts, are required to behave adaptively depending on the situation by dynamically recognizing environmental changes. To achieve adaptive and flexible behavior generation, the brain needs to develop strategies by constructing an internal model of the external world [1,2] through perceptual experiences. This internal model can be formulated in terms of a predictive coding [3] framework considering both action and perception, which is also called predictive *processing* [4], as connectionist [5–7] or Bayesian [8–11] schemes under the principle of prediction error minimization [12]. This paper considers two related but distinct advances in the modeling of behavior. First, we consider a key extension to our existing schemes that enable probabilistic representations of sensorimotor contingencies, through the encoding of uncertainty or variance (or inverse precision). Second, we consider not only learning schemes in which parameters are optimized to minimize (precision-weighted) prediction errors, but also adaptation or inference schemes in which internal states are dynamically modulated to minimize the prediction errors online. When combined, these two developments enable an agent (or robot) to infer the context in which it is acting and dynamically optimize estimates of uncertainty associated with bottom-up sensory information and top-down priors.

Tani and colleagues [5–7] proposed a deterministic connectionist scheme using a recurrent neural network (RNN) [13–16] based hierarchical generative model, called RNN with parametric biases (RNNPB). RNNPB can learn to map top-down priors to predictions about visuo-proprioceptive consequences of an action by means of prediction error minimization. The parametric biases (PBs) are higher-level static vectors corresponding to the top-down priors that determine the characteristics of the forward dynamics of a lower-level network in a manner similar to the bifurcation parameters, also known as control parameters, of nonlinear dynamical systems. They demonstrated that learning, generation, and recognition of multiple visuo-proprioceptive consequences of actions produced by a robot can be formulated as prediction error minimization. Under this formulation, the learning of action sequences is the process of optimizing network parameters including synaptic weights, biases that are shared by all sequences, and the PBs that are specific to each sequence, in order to regenerate the action sequences given visuo-proprioceptive sequences. After the learning process, a robot equipped with the trained network can regenerate each learned action sequence in a top-down manner based on the corresponding PB value, which represents a top-down prior, by sending the predicted proprioceptive state to the motor controller as the next target state of the robot. During top-down behavior generation, a dynamic recognition process can also be performed in a bottom-up manner by inferring the PB value that can regenerate a given visuo-proprioceptive state through prediction error minimization with fixed weights and biases. This scheme for the recognition process is referred to as the error regression scheme (ERS). The ERS introduces a fundamentally different sort of inference scheme in which the prediction errors are dynamically minimized online by the internal states (as well as the parameters).

When learning visuo-proprioceptive sequences with RNNs, through optimizing synaptic weights and biases or parameters, context sensitivity can be modeled using sensitivity to initial conditions of the internal neural dynamics. By allowing for precise or imprecise specifications of the initial conditions, one can nuance the context-sensitivity of subsequent action generation and recognition. Heuristically, the initial conditions encode different contexts by starting in different basins of attraction that give rise to attractor dynamics with distinct forms. In deterministic chaos, small differences in initial conditions can yield widely diverging state trajectories. A monkey electrophysiological study [17] suggests that preparatory activity in motor

and premotor cortex sets the initial state of a neural dynamical system whose evolution produces reaching movement activity. Nishimoto et al. [18] showed that the three aforementioned essential functions of learning, generation, and recognition can be achieved by using the sensitivity to initial conditions or initial precision characteristics of the context state of a continuous time RNN (CTRNN) instead of PBs, although the recognition process was a static rather than dynamic process.

Friston and colleagues [9–11] proposed a Bayesian predictive coding scheme, called "active inference", which entails the Bayesian brain hypothesis [19, 20] and is based on a free-energy principle [8,21,22]. Active inference can be organized by a hierarchically structured probabilistic generative model in which neural states at higher levels provide empirical priors on lower-level states in a top-down manner, and lower-level states provide prediction errors to higher levels for inference in a bottom-up manner. Under this scheme, prediction errors can be reduced by changing the externally given sensory signals being predicted and the internally generated predictions, through action and perception, respectively. As described in [11, 23], active inference is a generic Bayesian perspective on the above mentioned connectionist scheme using RNNPB. The key aspect of this Bayesian approach is the ability to deal with uncertainty or precision, which has been related to attentional mechanisms [8, 24–26]. The implicit estimation of uncertainty has not been considered in the deterministic connectionist scheme. In what follows, we will distinguish between probabilistic models and deterministic models, where probabilistic models develop the ability to dynamically predict context-sensitive fluctuations in the precision of sensory information. We will see later that this capacity only emerges in learning when top-down prior has a narrow or precise distribution, which enables violations to be recognized.

It can be considered that individuals construct an internal model through perceptual learning [27, 28] by making their own interpretation of perceptual experiences or observed events. In particular, when perceptual events are observed as occurring probabilistically, there could be two interpretations. One assumes a deterministic causal rule from the background or the context of the current perception and the other assumes a probabilistic rule. For example, let us suppose that one has already observed two sequences "AB" and "AD" where A, B and D are perceptual events. When one next perceives A, a probabilistic rule would predict the occurrence of B or D with equal probability. On the other hand, if one uses a deterministic rule, then the prediction of both occurrences would be made deterministically by inferring a distinct background or context for each case. More specifically in the current example, if different contexts C' or C'' can lead to the observation of A, the prediction of the next perceptual state as B or D is made deterministically depending on the context inferred. The problem here is ill-posed because one can use both types of rule even though the past experience is exactly the same.

It is presumed that the choice to use the probabilistic rule or the deterministic rule would affect significantly the method of behavior generation by agents while interacting with the world and with others. In what follows, we will distinguish two types of behavior generation schemes based on the origin of their causes [29], namely, "sensory reflex behavior" and "intentional proactive behavior." The former generation scheme corresponds to exogenously formed behavior in which actions are determined by external causes such as sensory inputs of the moment [30, 31]. In contrast, the latter scheme corresponds to endogenously formed behavior in which whole action sequences are represented by particular intentional states (internal causes) of agents [10, 32]. More specifically, intentional states here mean "prior intention" introduced by Searle [33] to distinguish from the other intention called "intention in action." In the case of intentional proactive behavior generation, different action sequences can be produced in terms of predictions (prior expectations) about visuo-proprioceptive consequences of actions depending on the intentional states. If the next perceptual state can be predicted only in a probabilistic manner, agents would generate sensory reflex behavior in which the next action to be taken will

be determined optimally after the perception is confirmed, as reaction to observed events. On the other hand, if the next and further perception can be predicted deterministically with confidence, agents would generate intentional proactive behavior in which the next and succeeding actions would be generated proactively based on a particular intentional state without waiting for the sensory input. In the case of sensory reflex behavior based on probabilistic prediction, action generation will be delayed whereas in the case of intentional proactive behavior, an over dependence on own prediction can lead to inflexibility in action modification when the prediction fails. There is another behavioral distinction, for example, between habitual behavior and goal-directed behavior. The former behavior can be acquired by model-free reinforcement learning (RL) approaches and the latter by model-based RL approaches [34–36]. It should be noted that our behavioral distinction in this paper is different from this distinction in terms of presence or absence of specific intentional states representing whole visuo-proprioceptive consequences of actions.

Tani and colleagues [32, 37, 38] demonstrated that not only deterministic sequences but also probabilistic transition sequences can be embedded in RNN-based deterministic models. In the context of action imitation learning by cognitive agents, Namikawa et al. [32] showed that a functional hierarchy [39, 40], which accounts for spontaneous behavior generation, can be self-organized in a multiple timescale RNN (MTRNN) [41]. The MTRNN consisted of a lower-level network containing a set of "action primitives" with fast dynamics and a higher level with slow dynamics that drove the lower-level network for combining the primitives. In their experiments, a humanoid robot controlled by the trained network was able to generate "pseudo-stochastic" action sequences by deterministic chaos self-organized in the higher-level network. When the transition probabilities of training data or observed events were changed, the network was able to reconstruct the probabilities in a deterministic manner by using the self-organized chaotic dynamics. Although this can be considered as one approach to the generation of stochastic sequences, it only models intentional proactive behavior generation and does not consider sensory reflex behavior. Furthermore, the problem of inflexibility in action modification has not been addressed.

Taking an alternative approach to deal with stochasticity, Namikawa et al. [42] proposed a novel CTRNN that can learn to predict not only the mean but also the variance of an observable variable at each time step, where inverse variance is called precision. This is a connectionist implementation of the previously mentioned stochasticity or uncertainty of observed events considered in the Bayesian approaches using the probabilistic generative model of Friston and colleagues. Instead of minimizing prediction error, the learning was conducted by maximizing an objective function in which prediction error is weighted by the predicted precision. In fact, Murata et al. [43] demonstrated that their CTRNN, referred to as stochastic CTRNN (S-CTRNN), can learn to reproduce both artificial and human-made stochastic sequences by correctly estimating both the time-varying mean and variance. Furthermore, Tan et al. [44] demonstrated the practical application and effectiveness of the S-CTRNN by conducting experiments on learning of object manipulation behavior using a mobile platform robot equipped with a pair of robotics arms.

Essential questions that have not been considered yet are how observed events are modeled as a probabilistic model, resulting in sensory reflex behavior generation, or a deterministic model, resulting in intentional proactive behavior generation, and what is the essential difference or relationship between these models. In the present study, we hypothesize that these different schemes can be produced by a single neural mechanism depending on the learning conditions. To explore the possible mechanisms, we developed a novel hierarchical dynamic neural network model, referred to as a stochastic MTRNN (S-MTRNN) model, based on our previous studies [18, 32, 41–43], and conducted a robotics learning experiment. The proposed

network has four main characteristics: First, the network has the ability to learn to predict both the means of subsequent sensory inputs and the corresponding uncertainty levels in terms of variances (inverse precisions) [42, 43]. One important aspect of the implementation of a variance prediction mechanism in a dynamic neural network is that the network still can be an intention-causal deterministic model if the network predicts zero variance. This means that the S-MTRNN has the potential to become both a probabilistic model and a deterministic model by self-organizing its distinct variance prediction mechanisms. Second, the network employs the multiple timescale dynamics of context states, which enables the self-organization of the functional hierarchy through learning of complex perceptual sequences [32, 41]. Third, the network can be regarded as a generative model that can regenerate different learned temporal sequences in a top-down manner by setting corresponding initial states, which are optimized through a learning process in the same way as synaptic weights and biases [18, 32]. Finally, we include a novel ERS that enables mutual interactions between internally performed top-down and externally originated bottom-up processes by dynamically modulating context states.

The robotics learning experiment using the proposed model concerns the problem of cooperative interaction with others under the assumption of potential uncertainty in the other's behavior. By analyzing the experimental results, we demonstrate that two different schemes of behavior generation, namely, sensory reflex behavior generation and intentional proactive behavior generation, can be developed depending on the condition of learning in the proposed model. More specifically, we show that different ways of treating unpredictable perceptual events in learning, namely, probabilistic modeling and deterministic modeling, contribute to the development of different dynamic neuronal structures governing these two types of behavior generation schemes.

2 Neural Network Model

2.1 Overview

The S-MTRNN, our proposed hierarchical dynamic neural network, can be regarded as a generative model for generating predictions of subsequent sensory inputs (i.e., visuo-proprioceptive states of a humanoid robot) in terms of the mean and variance of a Gaussian distribution with respect to a given intentional state [43]. Here, the variance corresponds to the uncertainty of variables and the reciprocal of the variance is called precision. The network consisted of input, context, output, and variance neural units. The input units received the current visuo-proprioceptive state, and the output and variance units generated the predictions of the mean and variance states for the next step, respectively. Thanks to the ability to predict not only the mean of sensory inputs but also the time-varying variance, the network can extract stochastic or fluctuating structures hidden in temporal visuo-proprioceptive sequences. The dynamics of the context units are described by a conventional firing rate model in which each unit's activity represents the mean firing rate over a group of neurons. The context units were divided into two groups characterized by a difference in time constants of neural activity. Hereinafter faster timescale units with a smaller time constant ($\tau_{\mathrm{FC}} = 5$) are called fast context (FC) units, and slower timescale units with a larger time constant ($\tau_{\mathrm{SC}} = 100$) are called slow context (SC) units. In addition to the multiple timescales of the neural activity, the two groups of context units had different connectivities to introduce constraints on information flow. The FC units with the smaller time constant were connected with all units. On the other hand, the SC units with the larger time constant were only connected with the context units. Yamashita and Tani [41] demonstrated that a difference of timescales enables the self-organization of a functional hierarchy in which a set of action primitives can be stored in the FC units, and sequential

combinations of the primitives can be represented in the SC units. The dynamics of the FC units started from a neutral initial state (zero value), and those of the SC units started from a particular initial state that had been optimized during the learning process [32] as described later. Thus, the two groups of FC and SC units can be regarded as a lower-level and a higher-level network, respectively. The higher level with SC units and the lower level with FC units may correspond to the rostral and the caudal part in cortex creating a so-called "rostro-caudal gradient" of timescales [45]. From the viewpoint of the integration of information processing and memory, Hasson et al. [46] proposed a hierarchical "process memory" framework based on their neurophysiological and neuroimaging studies. In their framework, the processing timescale of each area of cortex is characterized by the temporal receptive window (TRW), which is analogous to time constants in our model. They argue that the TRW gradually increases from early sensory areas to higher-order areas. Figure 1 (left) shows a schematic illustration of the proposed network.

The generation and training methods are described in the remaining subsections.

2.2 Generation Method

The forward dynamics of the internal states of the ith FC, SC, output, or variance units at time steps $t \geq 1$ corresponding to the sth sequence $u_{t,i}^{(s)}$ are given by

$$u_{t,i}^{(s)} = \begin{cases} \left(1 - \dfrac{1}{\tau_i}\right) u_{t-1,i}^{(s)} + \dfrac{1}{\tau_i} \left(\displaystyle\sum_{j \in I_{\mathrm{I}}} w_{ij} x_{t,j}^{(s)} + \sum_{j \in I_{\mathrm{FC}} \cup I_{\mathrm{SC}}} w_{ij} c_{t-1,j}^{(s)} + b_i \right) & (i \in I_{\mathrm{FC}} \cup I_{\mathrm{SC}}), \\ \displaystyle\sum_{j \in I_{\mathrm{FC}}} w_{ij} c_{t,j}^{(s)} + b_i & (i \in I_{\mathrm{O}} \cup I_{\mathrm{V}}), \end{cases} \tag{1}$$

where I_{I}, I_{FC}, I_{SC}, I_{O}, and I_{V} are the index sets for the input, FC, SC, output, and variance units, respectively, τ_i is the time constant of the ith context unit (τ_{FC} or τ_{SC}), w_{ij} is the synaptic weight of the connection from the jth to the ith unit, $c_{t-1,j}^{(s)}$ is the neural activation state of the jth context unit at time step $t-1$ corresponding to the sth sequence, $x_{t,j}^{(s)}$ is the jth input state at time step t corresponding to the sth sequence, and b_i is the bias of the ith unit.

In the present study, the synaptic weights w_{ij} were set to 0, which indicated disconnection, in a case where $i \in I_{\mathrm{SC}} \wedge j \in I_{\mathrm{I}}$ for the connection constraints. In addition to the multiple timescale property, the constraint on information flow derived from the connection setting is also essential for the self-organization of functional hierarchy [41]. The value of the initial internal state $u_{0,i}^{(s)}$ of the FC units ($i \in I_{\mathrm{FC}}$) was also set to 0, which indicated a neutral state regardless of the sequence s, and that of the SC units ($i \in I_{\mathrm{SC}}$) was optimized for each sequence as described in the following subsection. These values of the initial state mean that differences among multiple temporal sequences are represented only in the initial state space of the SC units.

The neural activation states of context unit $c_{t,i}^{(s)}$, output unit $y_{t,i}^{(s)}$, and variance unit $v_{t,i}^{(s)}$ are calculated by using the activation functions

$$c_{t,i}^{(s)} = \tanh(u_{t,i}^{(s)}) \qquad (i \in I_{\mathrm{FC}} \cup I_{\mathrm{SC}}), \tag{2}$$

$$y_{t,i}^{(s)} = \tanh(u_{t,i}^{(s)}) \qquad (i \in I_{\mathrm{O}}), \tag{3}$$

$$v_{t,i}^{(s)} = \exp(u_{t,i}^{(s)}) \qquad (i \in I_{\mathrm{V}}). \tag{4}$$

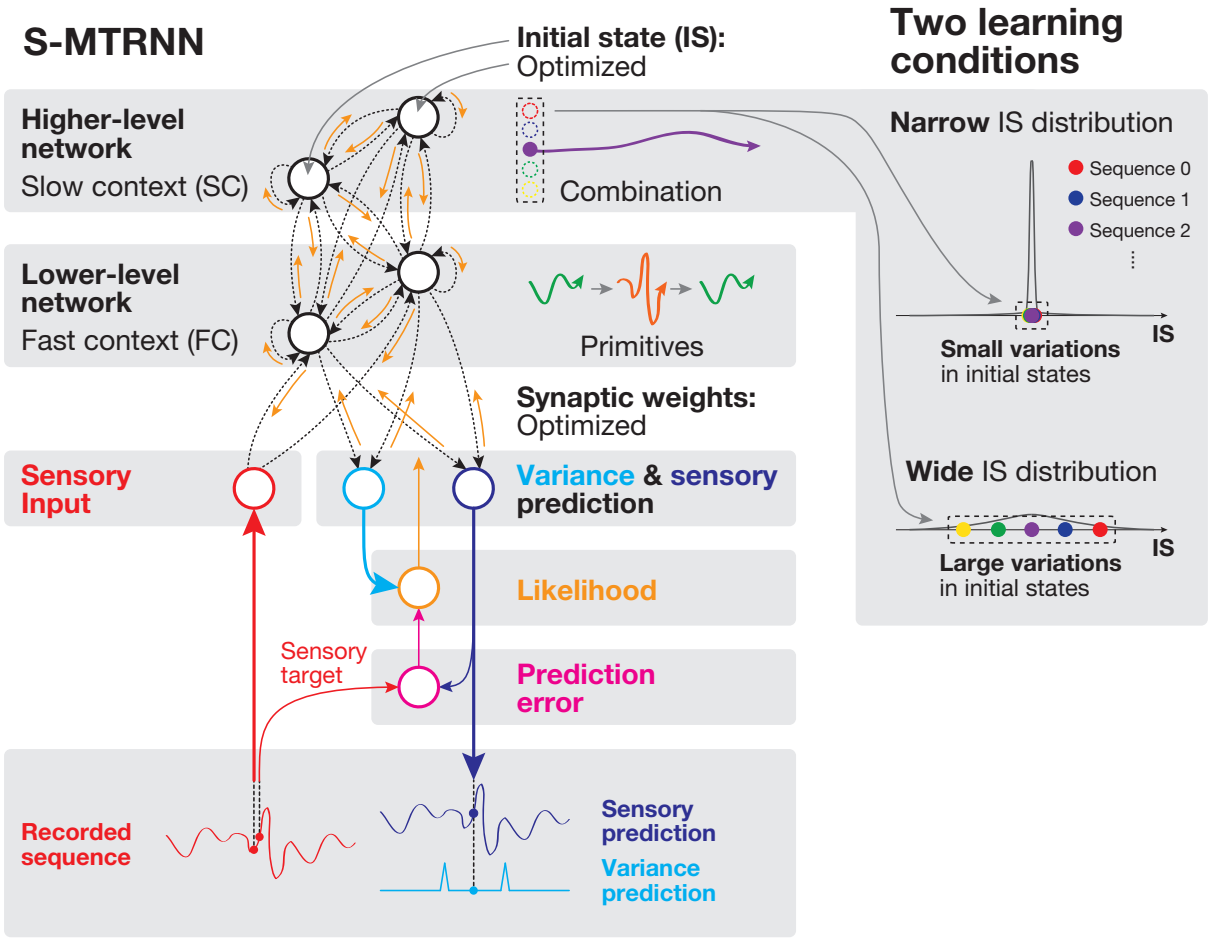

Figure 1. Schematic of hierarchical neural network model. The left part of the figure shows an S-MTRNN, comprising a higher-level network with SC units whose time constant was larger ($\tau_{\mathrm{SC}} = 100$) and a lower-level network with FC units whose time constant was smaller ($\tau_{\mathrm{FC}} = 5$). Input, FC, SC, output, and variance units are represented by nodes. Synaptic weights are represented by links (black dashed lines) between the nodes. The network was trained to predict the mean and variance of subsequent sensory inputs (i.e., visuo-proprioceptive states of a humanoid robot) under the principle of model likelihood maximization. The green and orange curves represent action primitives stored in the lower-level network of FC units. These primitives are combined by the dynamics of the higher-level network of SC units (a purple curve) whose internal state starts from a particular initial value (a purple circle) for generating a corresponding sequence. The right part of the figure illustrates different learning conditions on the initial internal state. By the "initial internal state" we mean the initial values of the internal states of the SC units ($u^{(s)}_{0,i}$). These initial conditions were optimized during learning and confer context sensitivity on the subsequent dynamics. The curves represent probability distributions over the value of the initial internal state (horizontal axis). Each colored circle represents an initial internal state associated with a particular visuo-proprioceptive sequence. Upper: narrow IS distribution condition ($\sigma^2_{\mathrm{IS}} = 0.00001$) resulting in the attenuation of the precision of slow contextual states. Lower: wide IS distribution condition ($\sigma^2_{\mathrm{IS}} = 10$) resulting in the opposing effect.

2.3 Training Method

The network was trained by means of supervised learning to maximize the likelihood

$$L_{\mathrm{out}} = \prod_{s \in I_{\mathrm{S}}} \prod_{t=1}^{T^{(s)}} \prod_{i \in I_{\mathrm{O}}} \frac{1}{\sqrt{2\pi v^{(s)}_{t,i}}} \exp\left(-\frac{(y^{(s)}_{t,i} - \hat{y}^{(s)}_{t,i})^2}{2v^{(s)}_{t,i}} \right), \tag{5}$$

where I_{S} is the sequence index set, $T^{(s)}$ is the length of the sequence, and $\hat{y}_{t,i}^{(s)}$ is the target value in the training sequence. It should be reminded that this objective function corresponds to the accuracy component of variational free energy. In the present study, the network was trained to predict future visuo-proprioceptive states of a humanoid robot with a time delay ξ (in the experiment $\xi = 5$) by observing the state $x_{t,i}^{(s)}$ at the current time step t. Therefore, the target value in (5) satisfies $\hat{y}_{t,i}^{(s)} = x_{t+\xi,i}^{(s)}$. This setting indicates that at the end of network training, if the network successfully learns to generate an output value $y_{t,i}^{(s)}$ that is exactly equal to the target value $\hat{y}_{t,i}^{(s)}$ along with a sufficiently small (almost zero) variance $v_{t,i}^{(s)}$, the network can autonomously regenerate target sequences without receiving external visuo-proprioceptive inputs by feeding the predicted state $y_{t-\xi,i}^{(s)}$ to the current input state $x_{t,i}^{(s)}$. The gradient ascent method with a conventional back-propagation through time (BPTT) [47] was used to maximize the likelihood. The contribution of the variance prediction mechanism is that the predicted variance functions as an inverse weighting factor for the sensory prediction errors, which are back-propagated in the process of learning. This is important because, if the training data contain unpredictable components, the learning cannot be conducted effectively without such an error scaling. Through this mechanism, the network can autonomously control the amount of back-propagated prediction error.

During the learning process network parameters, including synaptic weights, biases, and the initial internal states of the higher-level network composed of SC units, were optimized. Although both the synaptic weights and the biases were common parameters for all sequences, initial states were provided for each sequence. Because of this learning scheme, if the network possesses enough capacity to memorize whole visuo-proprioceptive sequences given for learning, each sequence can be regenerated in a top-down manner as a predictive distribution consisting of mean and variance states by choosing the corresponding initial state of the higher-level network. Thus, the initial states associated with corresponding visuo-proprioceptive sequences can be thought as intentional states for action generation.

One important consideration is that we assigned the following Gaussian distribution L_{init}, which defines the resemblance of states, to the initial internal state of the SC units:

$$L_{\mathrm{init}} = \prod_{s \in I_{\mathrm{S}}} \prod_{i \in I_{\mathrm{SC}}} \frac{1}{\sqrt{2\pi}\sigma_{\mathrm{IS}}} \exp\left(-\frac{(u_i - u_{0,i}^{(s)})^2}{2\sigma_{\mathrm{IS}}^2} \right), \tag{6}$$

where $u_{0,i}^{(s)}$ is the optimized initial internal state of the ith SC unit for the sth sequence, u_i is the optimized mean value of the initial states for the ith SC unit, and σ_{IS}^2 is a predefined variance that represents the variability of a set of initial states. The likelihood objective function in (6) corresponds to the (empirical) prior terms of variational free energy. Note that the prior beliefs implicit in (6) contribute only to the objective function and therefore only affect learning. Here, the σ_{IS}^2, corresponds to the inverse precision of the implicit prior beliefs about the (time-averaged) differences between slow contextual states and their (context-sensitive) initial conditions. In the present study, we employed two distributions: one with a small variance ($\sigma_{\mathrm{IS}}^2 = 0.00001$) and one with a large variance ($\sigma_{\mathrm{IS}}^2 = 10$) (see the right-hand part of Fig. 1). For simplicity, we refer to the former and the latter cases as the "narrow IS distribution" and the "wide IS distribution", respectively, where IS means initial state. By imposing these different learning conditions, we investigated the effect of the difference in the precision of the distribution over initial (SC) states on the self-organization of hierarchical prediction mechanisms.

The network parameters $\boldsymbol{\theta}$, consisting of weights, biases, initial internal states of the SC units, and the mean values of these initial states, are optimized to maximize the logarithm of the corresponding likelihood L. The parameters at learning step n ($\boldsymbol{\theta}(n)$) are updated by the

gradient ascent method with a momentum term:

$$\boldsymbol{\theta}(n) = \boldsymbol{\theta}(n-1) + \Delta\boldsymbol{\theta}(n), \tag{7}$$

$$\Delta\boldsymbol{\theta}(n) = \alpha\left(\frac{\partial \ln L}{\partial \boldsymbol{\theta}} + \eta\Delta\boldsymbol{\theta}(n-1)\right), \tag{8}$$

where α is the learning rate and η is a parameter representing the momentum term.

For updating the parameters $\boldsymbol{\theta}_{\text{share}}$, consisting of weights w_{ij} and biases b_i that are shared for the generation of all sequences, the likelihood L and the learning rate α in (8) are replaced with L_{out} and α_{share}, respectively. For updating the other parameters $\boldsymbol{\theta}_{\text{init}}$, consisting of the initial internal states of the SC units $u_{0,i}^{(s)}$ that are provided for the generation of each sequence s and their mean values u_i, the likelihood L and the learning rate α in (8) are replaced with $L_{\text{all}} = L_{\text{out}}L_{\text{init}}$ and α_{init}, respectively. Details about the calculation of gradients $\frac{\partial \ln L_{\text{out}}}{\partial \boldsymbol{\theta}_{\text{share}}}$ and $\frac{\partial \ln L_{\text{all}}}{\partial \boldsymbol{\theta}_{\text{init}}}$ are provided in Appendix A.

2.4 Parameter Settings for Network Training

The numbers of the input, output, and variance units were $N_{\text{I}} = N_{\text{O}} = N_{\text{V}} = 8$, respectively. These were determined by the dimensionality of the visuo-proprioceptive state of a humanoid robot (two-dimensional visual inputs, two-dimensional head joint angles, and four-dimensional right arm joint angles as described in the following section). The numbers of FC and SC units were $N_{\text{FC}} = 30$ and $N_{\text{SC}} = 10$, respectively. The time constants of the FC and SC units were $\tau_{\text{FC}} = 5$ and $\tau_{\text{SC}} = 100$, respectively. These parameters characterizing contextual dynamics were empirically determined from $N_{\text{FC}} = \{30, 60\}$, $N_{\text{SC}} = \{10, 20\}$, $\tau_{\text{FC}} = \{5, 10\}$, and $\tau_{\text{SC}} = \{70, 100\}$ whose combinations were used in previous studies of MTRNNs [41,48,49]. It is considered that the ratio between τ_{SC} and τ_{FC} (tau-ratio: $\tau_{\text{SC}}/\tau_{\text{SC}}$) should be larger than the ratio between the length of training sequences and that of primitives in the sequences (length-ratio: sequence-length/primitive-length or the number of primitives in each sequence) [41]. We chose the smaller and the larger time constant for the FC and SC units, respectively, and the smaller numbers of the FC and SC units from the candidate parameters. In our present study, each sequence consisted of five primitives as described in the following section. Therefore, the minimum necessary requirement of the tau-ratio under the current task setting is five and the set value $\tau_{\text{SC}}/\tau_{\text{FC}} = 20$ was sufficiently larger than the required value. We confirmed that the above setting of the time constants and the numbers of the FC and SC units can achieve the convergence of the model likelihood (or training error) in the wide IS distribution condition. The same parameters were employed for the training in the narrow IS distribution condition for comparison.

Synaptic weights w_{ij} were initialized with values randomly chosen from a uniform distribution on the intervals $\left[-\frac{1}{N_{\text{I}}}, \frac{1}{N_{\text{I}}}\right]$ (if $j \in I_{\text{I}}$) and $\left[-\frac{1}{N_{\text{C}}}, \frac{1}{N_{\text{C}}}\right]$ (otherwise), where $N_{\text{C}} = N_{\text{FC}} + N_{\text{SC}} = 40$ is the number of context units. Biases b_i were initialized with values randomly chosen from a uniform distribution on the interval $[-1, 1]$. Initial internal states $u_{0,i}^{(s)}$ and the mean values u_i of the initial states were set to 0 and values randomly chosen from a uniform distribution on the interval $\left[-\frac{1}{N_{\text{C}}}, \frac{1}{N_{\text{C}}}\right]$, respectively. Since the maximum value of L_{out} depends on the total length T_{total} of the training sequences and the dimensionality $N_{\text{O}} = 8$ of the output units, the learning rate α_{share} for updating weights and biases was scaled by a parameter $\tilde{\alpha}$ satisfying the relation $\alpha_{\text{share}} = \frac{1}{T_{\text{total}}N_{\text{O}}}\tilde{\alpha}$. The learning rate for updating initial internal states and mean values was $\alpha_{\text{init}} = \frac{1}{N_{\text{O}}}\tilde{\alpha}$. In all experiments presented here, $\tilde{\alpha}$ and the momentum term were $\tilde{\alpha} = 0.0001$ and $\eta = 0.9$, respectively, which were determined based on our previous study of S-CTRNNs [43]. To accelerate network training, we employed the adaptive learning rate scheme based on the works of Namikawa et al. [32,38]. Details are provided in Appendix B.

It should be noted that because most parameters are scaled by the features of training data such as the total length and the dimensionality of the data, the above setting can be reused for other trainingdata. Although the non-scaled parameters including the time constants and the numbers of the context units should be tuned by trial and error, learning results are not so sensitive to their setting. Our preliminary trials demonstrated that the other combinations of the time constants and the numbers of FC and SC units also provide the convergence of the model likelihood (or training error) in addition to the adopted setting.

2.5 Error Regression Scheme

After the network training, a dynamic recognition of situational changes can be achieved by inferring the internal state of the higher-level network with SC units that can reproduce the perceived visuo-proprioceptive states within a certain time window of the immediate past. More specifically, the internal states of the SC units are modulated in a way that maximizes a part of the likelihood defined in (5) by regressing past perceptual experiences. This is a formal extension of active inference [11] and also analogous to the online error regression of the PB [6] applied to the problem of dynamic recognition of others' actions.

The internal states of the SC units at time step $t-W$ ($u^{(s)}_{t-W,i}$) are dynamically modulated by using the gradient ascent method to maximize the likelihood

$$L_{\rm reg} = \prod_{t'=t-W}^{t} \prod_{i\in I_{\rm O}} \frac{1}{\sqrt{2\pi v^{(s)}_{t',i}}} \exp\left(-\frac{(y^{(s)}_{t',i} - \hat{y}^{(s)}_{t',i})^2}{2v^{(s)}_{t',i}}\right), \tag{9}$$

where the same BPTT scheme [47] adopted for the learning process was used without changing the connectivity weights, and $W = 30$ is the length of the time window that shifts along with the increment of the time step t'. This error regression was conducted for 50 regression steps for each time step t. In this study, the mean and variance predictions of visuo-proprioceptive states and the context states for time steps from $t-W+1$ to t were generated using closed-loop generation in which the "re-interpreted" or "postdicted" [49] mean visuo-proprioceptive state was used as an input for the next time step. During this generation mode, the set of internal states of the SC units at time step $t-W$ serves as an initial internal state within the time window for ERS. Although we can consider a balance between the forward dynamics state predicted in the past and that state postdicted at the present with the regression, in the present study we assumed that the former state is completely overwritten with the latter state. In the robot experiments described in the forthcoming sections, we investigated the difference of behavior produced by the robot driven by the trained network with or without dynamic recognition using the ERS.

2.6 Parameter Settings for ERS

For updating internal states of the SC units using ERS, the likelihood L and the learning rate α in (8) are replaced with $L_{\rm reg}$ and $\alpha_{\rm reg}$, respectively. The learning rate was set to the same value used in the updating of the initial internal states of the SC units. During the error regression process, the adaptive learning rate scheme was not used for real-time computation and, therefore, the value was fixed.

3 Robot Experiment

In the first set of experiments without ERS we optimize the parameters of the agent's forward or generative model by learning, while in the second set of experiments with ERS, we consider

optimization of both the parameters by learning and the internal states by inference through maximizing an objective function. In both cases, this objective function is an approximation to the Bayesian evidence or marginal likelihood of the generative model. Under some simplifying assumptions this can be approximated with the sum of squared prediction errors that are weighted by their precision. Crucially, these prediction errors arise at all levels of a hierarchical model, thereby accommodating prior beliefs at higher levels. This objective function has exactly the same form as that used in Bayesian filtering and free energy formulations of active inference. The important thing in the present setting is that to evaluate this model likelihood, the agent has to estimate variance or precision in a context sensitive way.

3.1 Design of Robot Experiment

We employed two small humanoid robots ("NAO" developed by Aldebaran Robotics). One robot called the "self-robot" was required to generate action sequences corresponding to those generated by the "other-robot" via learning in a supervised manner. More specifically, the self-robot faced the other-robot and learned to predict how the positions of an object held by the other-robot change in time based on visual sequences. It also learned its own corresponding arm movements in terms of proprioceptive sequences.

Figure 2 (left) shows a schematic illustration of the task. The self-robot was controlled by the S-MTRNN model and the other-robot followed action sequences pre-programmed by the experimenter; for simplicity, only the self-robot was required to generate adaptive behavior and the other-robot's action sequences were not affected by the self-robot. In the task, the other-robot arbitrarily repeated action primitives involving moving a colored object either to the left (labeled as "L") or to the right (labeled as "R") from the view point of the self-robot. The self-robot was required to generate corresponding behavior in terms of moving its right arm in the same direction and simultaneously with the other-robot at each time step. For the purpose of simultaneous action generation, the self-robot was required to predict the direction in which the object was about to be moved before the other-robot actually generated its movement. The self-robot acquired this skill in a supervised learning phase. In the context of adaptation to the other-robot's behavior, the task for the self-robot can be considered as a cooperative interaction task in which the self-robot attempted to generate cooperative behavior with the other-robot. It should be noted that, after the learning phase, the S-MTRNN model implemented in the self-robot cannot predict the visuo-proprioceptive sequence completely in the test phase in this task because the decision about whether to move the object to the left or to the right at each branching level is generated at random by the pre-programmed other-robot. In this context, the objective of the experiment was to examine how, after learning, the self-robot can generate cooperative behavior by adapting to the other-robot's behavior even when the self-robot occasionally fails to make a correct prediction about the branching points.

3.2 Experimental Procedure

The robotics learning experiments consisted of (1) obtaining visuo-proprioceptive training sequences from the self-robot interacting with the other-robot, (2) training the S-MTRNN in an offline manner using the obtained training sequences, and (3) an action generation test in which the self-robot was controlled by the trained network.

In the first phase of data recording, the self-robot was directly tutored on its own movements in terms of head angle and right arm posture for generating cooperative behavior that matched the actions of the other-robot for action primitive sequences consisting of five-level decision branching. Each branch corresponded to moving the object to the left or to the right. In the current experiment, 32 pattern sequences covering all possible five-level branching sequences

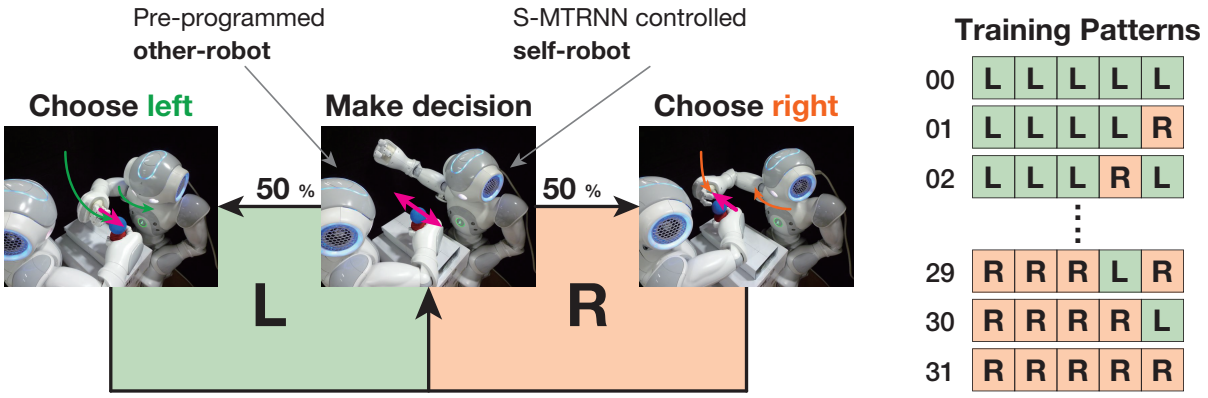

Figure 2. Task design. The left part of the figure shows a schematic of a cooperative interaction task. The "self-robot" controlled by the S-MTRNN and a pre-programmed "other-robot" were used in the experiment. In the task, the other-robot arbitrarily moved a colored object either to the left or to the right with equal probability ($P = 0.5$). The self-robot was required to learn to cooperatively interact with the other-robot in terms of moving its right arm in the same direction at the same time as the other-robot (left and right photographs in the figure). The right part of the figure illustrates the 32 training patterns used in the experiment. These patterns covered all the transition patterns in the case where the transition was repeated five times in each sequence.

(such as "RRLLR") were recorded as a training data set (see the right hand part of Fig. 2). Each sequence consisted of a time series of a two-dimensional vector representing the object's center position in visual images obtained from the camera mounted on the tutored self-robot, a two-dimensional vector representing head joint angles (yaw and pitch), and a four-dimensional vector representing right arm joint angles (shoulder pitch, shoulder roll, elbow yaw, and elbow roll) of the self-robot. These data were recorded 10 times a second, and the length of each sequence $T^{(s)}$ was about 470.

In the second phase of the network training, visuo-proprioceptive states recorded in training sequences were mapped to values ranging between -0.8 and 0.8 because the activation function of the output units of the S-MTRNN was tanh (ranging between -1.0 and 1.0). For each learning condition (narrow IS distribution and wide IS distribution) 10 learning trials were carried out with different initial parameters $\boldsymbol{\theta}(0)$ by changing the seed of a random number generator. The S-MTRNN was trained to predict the mean and variance states of the visuo-proprioceptive state at the next time step by using the current state in the training sequence.

In the third phase of actual action generation, an arbitrary set of initial internal states of the SC units was given to the trained network. The network generated a prediction of the visuo-proprioceptive state at time step $t+\xi$ by observing that state at the current time step t. At this time, the proprioceptive state predicted at the time step $t-\xi+1$ by the trained network was remapped to values within the range of the robot's joint angles. The remapped values were sent to the self-robot in the form of target joint angles that acted as motor commands for the robot to generate movements. Changes in the environment, including changes in the object position and changes in the actual joint angle positions were sent back to the network as sensory feedback. In both the first and third phases, the self-robot's head joint angles were controlled to automatically fix on the center of the target object, regardless of the robot's behavior.

3.3 Training Results

We trained the S-MTRNN with two different learning conditions, one with the narrow IS distribution ($\sigma^2_{\mathrm{IS}} = 0.00001$) and the other with the wide IS distribution ($\sigma^2_{\mathrm{IS}} = 10$), for 500 000 training steps. One training trial took about 3 h in our computation environment. Details of

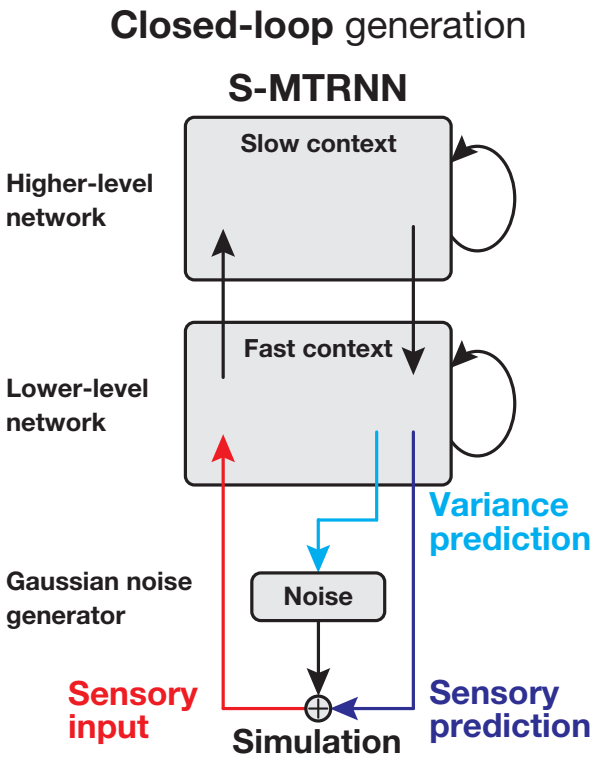

Figure 3. Scheme for the closed-loop generation corresponding to the simulation of actions.

parameter settings are provided in Section 2.4.

After the network training, we first analyzed the visuo-proprioceptive sequences produced by each trained network with a closed-loop generation before the self-robot commenced the actual cooperative interaction with the other-robot. In this generation mode, an optimized initial internal state of the SC units is set for the higher-level network, and the input state at the current time step is derived from the predicted mean value to which Gaussian noise with the variance predicted at the previous time step is added [43]. Because this generation mode does not receive actual sensory feedback, the process can be regarded as a simulation of action generation or motor imagery [32, 41, 50] taking fluctuations into account. Figure 3 shows a schematic illustration of the closed-loop generation mode.

The trained network can deterministically regenerate learned sequences if adequate initial states are acquired for each sequence [32] and if the estimated variances remain sufficiently small (almost zero) through time. If the estimated variances are non-zero, for example at decision branching points, the sequences are generated stochastically by the effect of Gaussian noise added in accordance with the predicted variance.

Figure 4A and 4B show examples of sequences reproduced by the trained network in the narrow and wide IS distribution conditions. The sequences include time series of sensory targets (training data), sensory (mean) predictions, variance predictions, SC states, and FC states obtained from the S-MTRNN with closed-loop generation in which the initial state of the higher-level network was set to the "RRLLR" sequence.

In the narrow IS distribution condition, the network was unable to reproduce any learned sequence associated with a particular initial state with closed-loop generation. In Fig. 4A, for example, although the initial state for the network was set with the values optimized for the "RRLLR" sequence in the learning process, the generated sequence was "LRLLR". The figure

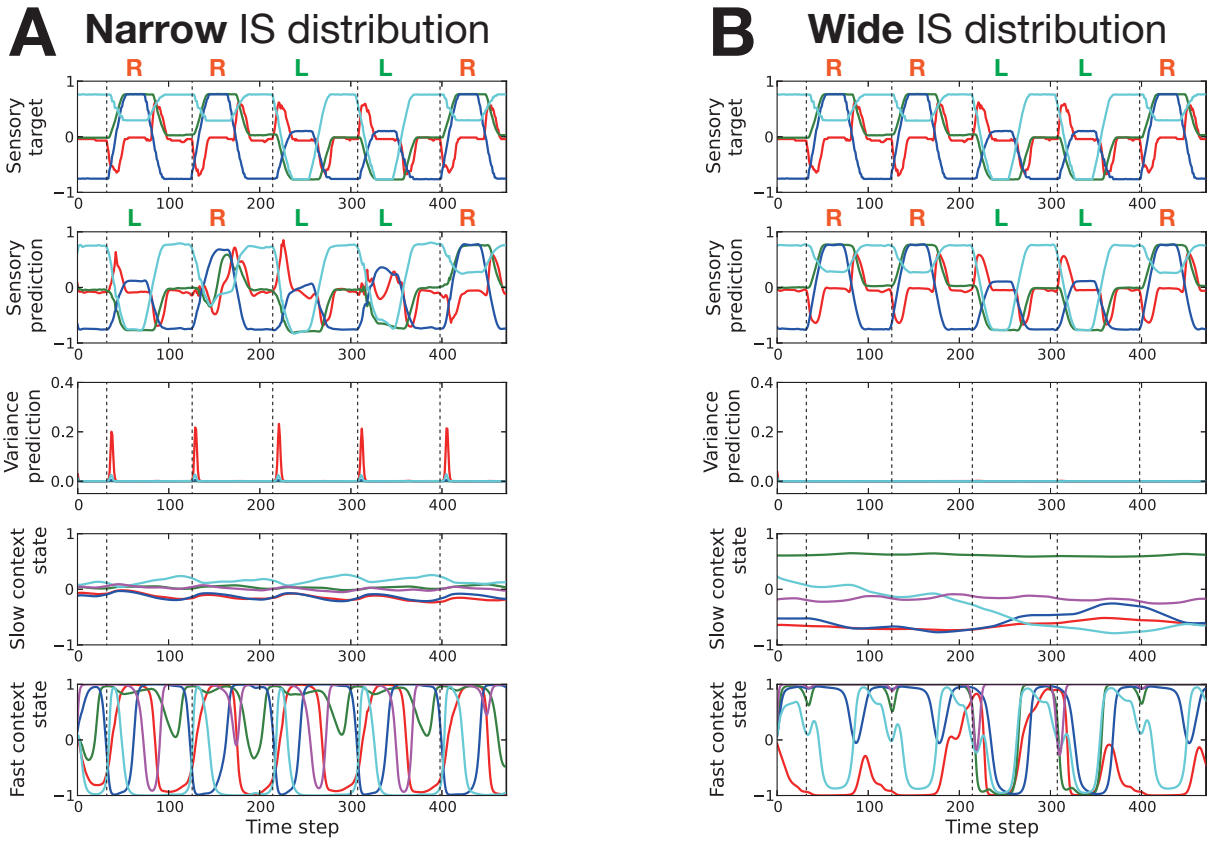

Figure 4. Time series obtained in the experiment. Time series of sensory targets (training data), sensory predictions (network mean output), variance predictions, SC states, and FC states during the closed-loop operation of the networks trained with (A) the narrow IS distribution and (B) the wide IS distribution. One time step corresponds to 100 ms. In the upper three panels, the red, green, blue, and cyan lines indicate the horizontal position of the object in a visual image, the head yaw angle, the shoulder pitch angle, and the elbow yaw angle, respectively. In the lower two panels, neural activities of five selected units (from 10 SC units and 30 FC units) are shown. The vertical dashed lines indicate branching points from which the object was moved either to the left or to the right by the other-robot in the training sequence. The labels over the panels of sensory targets and sensory predictions denote the action performed by the self-robot.

suggests that a large variance is predicted at each branching point, indicating that the network regards the forthcoming perceptual event as unpredictable or uncertain. This prediction of a large variance or small precision at each branching point is reasonable because the visual input derived from the other-robot's behavior is essentially unpredictable. We confirmed that the network was able to produce various combinatorial sequences derived from visual perturbations caused by self-generated noise with the predicted variances added at each branching point, instead of utilizing sensitivity to initial conditions of the higher-level network. These results indicate that the network developed a probabilistic prediction model at the branching points and a deterministic one for the other segments. It can be said that the regenerated sequences contained deterministic chunks (moving to either the left or the right) with probabilistic transitions.

In contrast, in the wide IS distribution condition, the network was able to reproduce every learned sequence from the acquired initial state set in the higher-level network. In Fig. 4B it can be seen that the predicted variance at all times is nearly zero. These results indicate that the branching sequences are reproduced as deterministic dynamic sequences depending on the

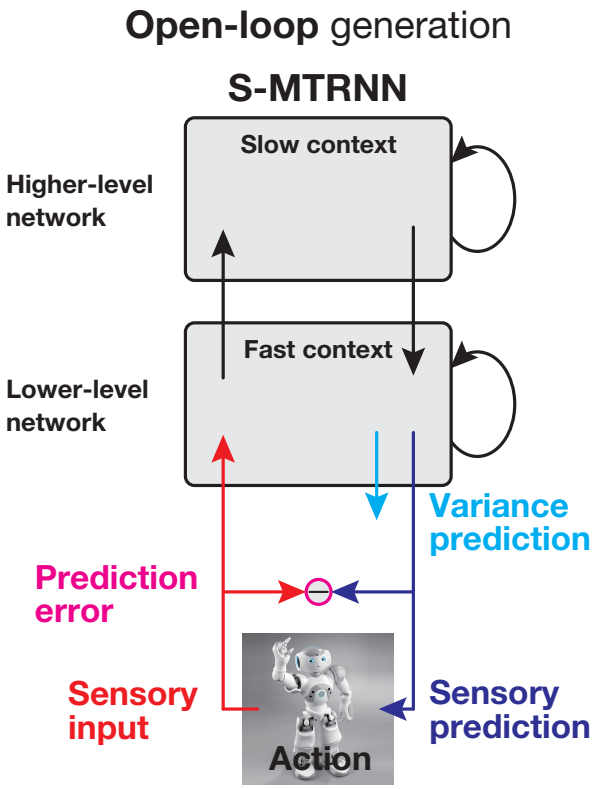

Figure 5. Scheme for open-loop generation.

initial state.

The two IS distributions also show differences in terms of FC and SC unit states. In the narrow IS distribution condition, the initial value of each SC unit is not widely spread and is located near 0. The activities of both FC and SC units at the branching points seem to be almost the same, regardless of future sensory predictions. Therefore, it can be said that transitions are determined not by the internal context dynamics but by the self-generated noise, where the variance shows a sharp peak at the branching point. In contrast, in the wide IS distribution condition, both SC and FC units exhibit specific activation patterns. The dynamics of the SC units gradually change and those of the FC units have distinct forms at each branching point by which the subsequent sensory predictions can be discriminated. In summary, top-down prediction does not take place at branching points in the narrow IS distribution condition, whereas it does when using the deterministic neural dynamics developed with the initial precision characteristics in the wide IS distribution condition.

3.4 Action Generation without ERS

After evaluating the training results, we performed experiments looking at actual cooperative interactions between the self-robot and the other-robot. In these experiments, the self-robot was controlled by the trained S-MTRNN with open-loop generation, in which the input state at the current time step is derived from actual sensory feedback during action generation, and the predicted proprioceptive state was sent to the robot in the form of target joint angles in order to control the robot's movement based on the network prediction. Figure 5 shows a schematic illustration of the open-loop generation mode. The other-robot was pre-programmed and its behavior was not affected by the self-robot.

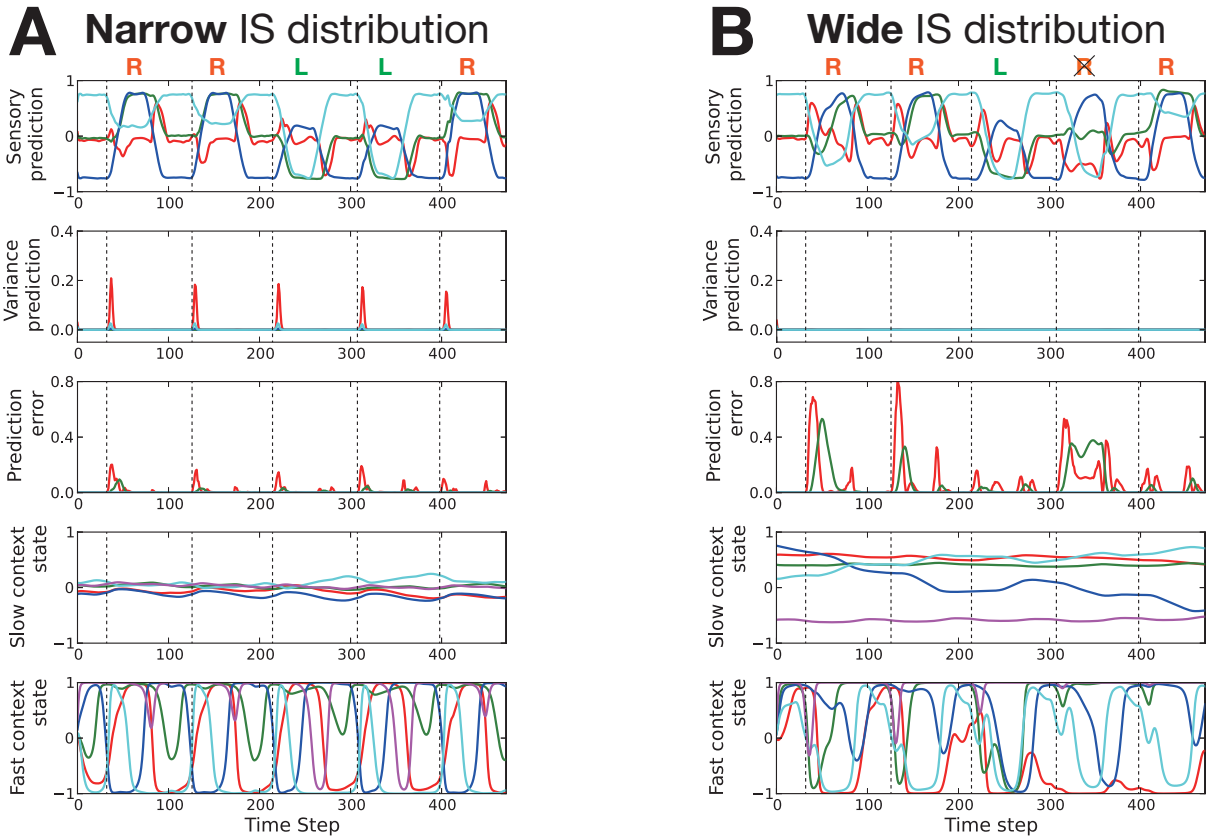

Figure 6. Time series obtained in the experiment. Time series of one-step sensory predictions, variance predictions, prediction errors, SC states, and FC states during action generation of the robot using the network trained with (B) the narrow IS distribution and (C) the wide IS distribution. The format of these panels is the same as that in Fig. 4B and 4C, except that here prediction errors are shown instead of sensory targets. In the case of the wide IS distribution, the cross on one label "R" represents a failure to predict the behavior of the other-robot, in which the hand of the self-robot collided with that of the other-robot.

To observe how the self-robot adapted to the other-robot's unpredictable behavior, we set an arbitrarily selected initial state for the trained S-MTRNN controlling the self-robot. Therefore, there is a discrepancy between the actions of the other-robot as anticipated by the self-robot and the actual actions generated by the other-robot during the interaction. Through the action generation test, we investigated how unmatched internal context dynamics can be modified in order to adapt to the unpredictable behavior of the other-robot in the two learning conditions. Success or failure in this test phase was distinguished by means of the self-robot's hand position. A movement was judged to be successful when the hand was moved from the center of the body to the object's side, while it was judged to be a failure when the hand moved to the opposite side of the object or collided with the other-robot.

Figure 6A and 6B show examples of time series of online-sensory predictions, variance predictions, prediction errors, SC states, and FC states obtained from the S-MTRNN for narrow and wide IS distribution conditions implemented on the self-robot. In this example, the initial state for the self-robot was set with the values optimized for the "LLRRL" sequence in the learning process, whereas the other-robot was programmed to generate the "RRLLR" sequence for both the narrow IS and the wide IS distribution conditions. This setting means that if a particular initial state encodes the learned action sequence of "LLRRL", this initial state can regenerate the same sequence for the self-robot's action plan which becomes exactly opposite

to the action program to be generated by the other-robot.

In the narrow IS distribution condition, it was observed that the self-robot succeeded to generate the "RRLLR" sequence which corresponds to the sequence generated by the other-robot without any failure movements. This is evidenced by the observation that the one-step prediction profile is rather similar to the sensory target profile of the same sequence shown in Figure 4A. Also, it can be seen that some prediction errors were generated at each branching point. The states of both SC and FC units at each branching point are almost the same.

In contrast, for the wide IS distribution condition, the one-step prediction sequence was significantly poorer than that for the narrow IS distribution condition. In fact, the prediction error at each branch point often became significantly larger than the one in the narrow IS distribution condition. In this situation, the movement of the self-robot became erratic. Although the self-robot seemed to try to follow the movements of the other-robot, its movements were significantly delayed. Furthermore, after three transitions between action primitives (i.e., after 300 time steps), the hand of the self-robot collided with that of the other-robot because the two robots moved their arms in opposite directions.

At first glance, it may appear counterintuitive that the narrow IS distribution can cope with violations of top-down predictions, whereas the wide IS distribution could not. Heuristically, one can understand this as follows: because we are optimizing the dynamics through the parameters, then the self-robot (after learning) is only optimal when the world behaves as expected. Crucially, these expectations include beliefs about precision. Therefore, paradoxically, an agent with a narrow IS distribution at the highest level learn that (precise) beliefs at lower levels can be violated and therefore contextualize sensory information by modulating the precision of prediction errors at those levels. In other words, only an agent with a narrow IS distribution can recognize its beliefs at lower levels are not always true.

To evaluate the effect of differences in initial states on cooperative interactions, we measured the reaction time of the self-robot, which is the number of time steps before the self-robot generated cooperative action primitives corresponding to the other-robot's action, for the two learning conditions. In each condition, two initial types of state for the trained network were considered. In one case the initial state corresponded to the other-robot's action sequence (this is referred to as the "corresponding IS" case) and in the other case an arbitrarily selected initial state was used (this is referred to as the "non-corresponding IS" case). In the current experiment, the initial state optimized for the "LLRRL" sequence was adopted for the non-corresponding IS case regardless of the other-robot's action sequences. This analysis was conducted not on physical interaction results but on ones simulated by using a set of recorded data and 10 trained sample networks with differently randomized initial parameters for each IS distribution condition. For details of the measurement, please refer to the Appendix C. We computed mean reaction times over the 10 trained sample networks, each of which generated 32 sequences including all combinations of the five transitions of the two action primitives (or 160 branches), for the corresponding and non-corresponding IS cases in each learning condition.

Figure 7 shows the computed mean reaction times in each case and results of t-test. In the narrow IS distribution condition, there is no significant difference ($t(18) = 0.71$, n.s.) in reaction times between the corresponding IS case ($M = 17.77$, $SD = 4.37$) and the non-corresponding IS case ($M = 17.80$, $SD = 4.36$). This result indicates that initial precision characteristics were no longer utilized on learning of different visuo-proprioceptive sequences and that the self-robot's behavior after learning relied not on internally generated context dynamics but on externally given sensory inputs. In contrast, in the wide IS distribution condition, there is a significant difference ($t(18) = 11.63$, $p < 0.001$) in reaction times between the corresponding IS case ($M = 0.004$, $SD = 0.008$) and the non-corresponding IS case ($M = 36.64$, $SD = 4.19$). In both the corresponding and non-corresponding IS cases, there are significant differences of

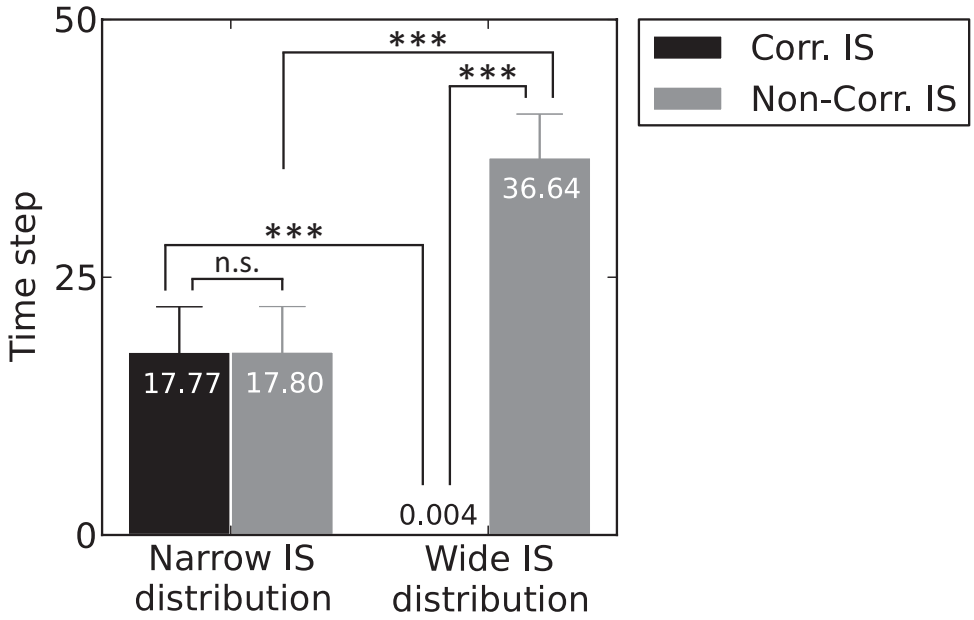

Figure 7. Reaction times of the self-robot during action generation. Times are given for the results of the simulated self-robot's action generation using the network trained with the narrow and wide IS distribution conditions. Bars and numbers in the graph correspond to mean reaction times over 10 trained sample networks for each IS distribution condition, each of which generated 32 sequences including the five transitions of the two action primitives. Black bars show the reaction times for initial states corresponding to the other-robot's action sequences and the gray bars show times for an arbitrarily selected (non-corresponding) initial state. In this case, the initial state optimized for the "LLRRL" sequence was adopted regardless of the other-robot's action sequences. Error bars indicate the standard deviation. Stars indicate a significant difference (***: $p < 0.001$) and "n.s." indicates no significant difference.

reaction times between the narrow and wide IS distribution conditions ($t(18) = 12.21$, $p < 0.001$; $t(18) = 9.34$, $p < 0.001$; respectively). The shortest mean reaction time obtained in the wide IS distribution condition indicates that when the network was placed in the corresponding initial state, the action generation of the self-robot was almost synchronized with that of the other-robot. On the other hand, the longest mean reaction time obtained in the wide IS distribution condition indicates that a long time was required to revise behavioral contexts when the self-robot's anticipation derived from the initial state failed.

These differences observed between the two learning conditions can be attributed to the different neural dynamic structures developed for these conditions. In the case of the probabilistic dynamic structure developed in the narrow IS distribution condition, the behavior of moving either to the left or to the right is determined simply by following the other-robot by means of a sensory reflex without any top-down prior based on initial states. Therefore, the difference in initial states did not affect the self-robot's behavior or its reaction times. In contrast, in the case of the deterministic dynamic structure developed to be sensitive to the initial state, the top-down prior at the branching point is too strong to be modified by the sensory input. Therefore, when corresponding initial states were given to the network, it worked positively toward the realization of cooperative interactions without any time delay. However, when the non-corresponding initial states were given, the self-robot required a long time to adapt to the unanticipated actions of the other-robot and sometimes adaptation was not achieved. To consider this problem, we examined the effect of introducing an additional neural mechanism of bottom-up recognition by using error information.

3.5 Interaction between Top-down Proactive Action Generation and Bottom-up Error Regression

As we learned from the experiment in the preceding subsection, when the top-down prior was too strong, a simple sensory reflex was insufficient for revising the internal neural dynamics. To solve this problem, we introduced the ERS into the trained network and reconducted an action generation test in the wide IS distribution condition. Figure 8A shows a schematic illustration of the open-loop generation with ERS.

Figure 8B shows an example of the time series of sensory predictions, variance predictions, prediction errors, SC states, and FC states obtained from the S-MTRNN with ERS with a wide IS distribution condition implemented on the self-robot. Clearly, the SC states in the gray areas change in a discontinuous manner. Modulating the higher-level SC states in this way by using ERS caused drastic changes in lower-level network activity, including the FC states and sensory predictions. Through these processes, the prediction errors were rapidly suppressed, and thus the self-robot was able to revise its behavioral context immediately after encountering unanticipated perceptual events.

To clarify the dynamic process of the regression by which the record of past states in the window can be overwritten and that of the prediction in which future plans can be modified by the regression, we extracted the states for time steps 175 to 265 from Fig. 8B. Figure 8C shows three sets of the states when the current time step (the window head) is 221, 224, and 227, corresponding to the "pre-modification," "modification," and "post-modification" phases, respectively. In the figure, the dynamically sliding windows are shown as gray frames. The figure shows the states in the past (left side of the window head) with solid lines, in the future (right side of the window head) with dashed lines, and at the present (the window head) with labels "Now" and specific time indexes indicating the boundary between the past and future states. It should be noted that although the range of the time step of each panel is the same (from 175 to 265), not only future predicted states but also the record of the past states at the same time step can differ from each other because immediate past states within the time window can be overwritten by revising the internal neural dynamics using ERS. Past states outside of the window are not affected by the regression dynamics and are constant.

From Fig. 8C, we can see that the past states were overwritten in the modification phase (center panels). In the pre-modification phase (left panels), the neural dynamics of the FC state and the predicted visuo-proprioceptive states corresponded to the action primitive labeled as "R". At this time, the prediction error of the visual input representing the horizontal position of the object (red line) has increased, meaning that there was a discrepancy between the prediction and actual sensory (visual) feedback. In the previous subsection, we observed that the prediction error at each branching point cannot be suppressed (see the wide IS distribution condition in Fig. 6) by only the received sensory inputs. On the other hand in the modification phase shown in Fig. 8C, the large prediction error was suppressed by the bottom-up recognition using ERS that revises the internal neural dynamics by back-propagating the precision-weighted prediction error to the higher-level network. During this process, the internal states of the SC units at the window tail were slightly modulated, and the modulation affected the lower-level dynamics of the FC units. By comparing the pre-modification and modification phases, we can confirm that the activity of the FC units was distinctly different. By revising the past SC and FC dynamics, the previously generated prediction (past) states were overwritten, and the future plan changed from moving the hand to the right to moving it to the left. At this time, the past states outside of the window were not affected by the recognition dynamics as mentioned before. In the absence of the prediction error, the modulation of the internal neural dynamics and overwriting of past states were not performed, and forward dynamics were smoothly generated, as seen in the post-modification phase (right panels).

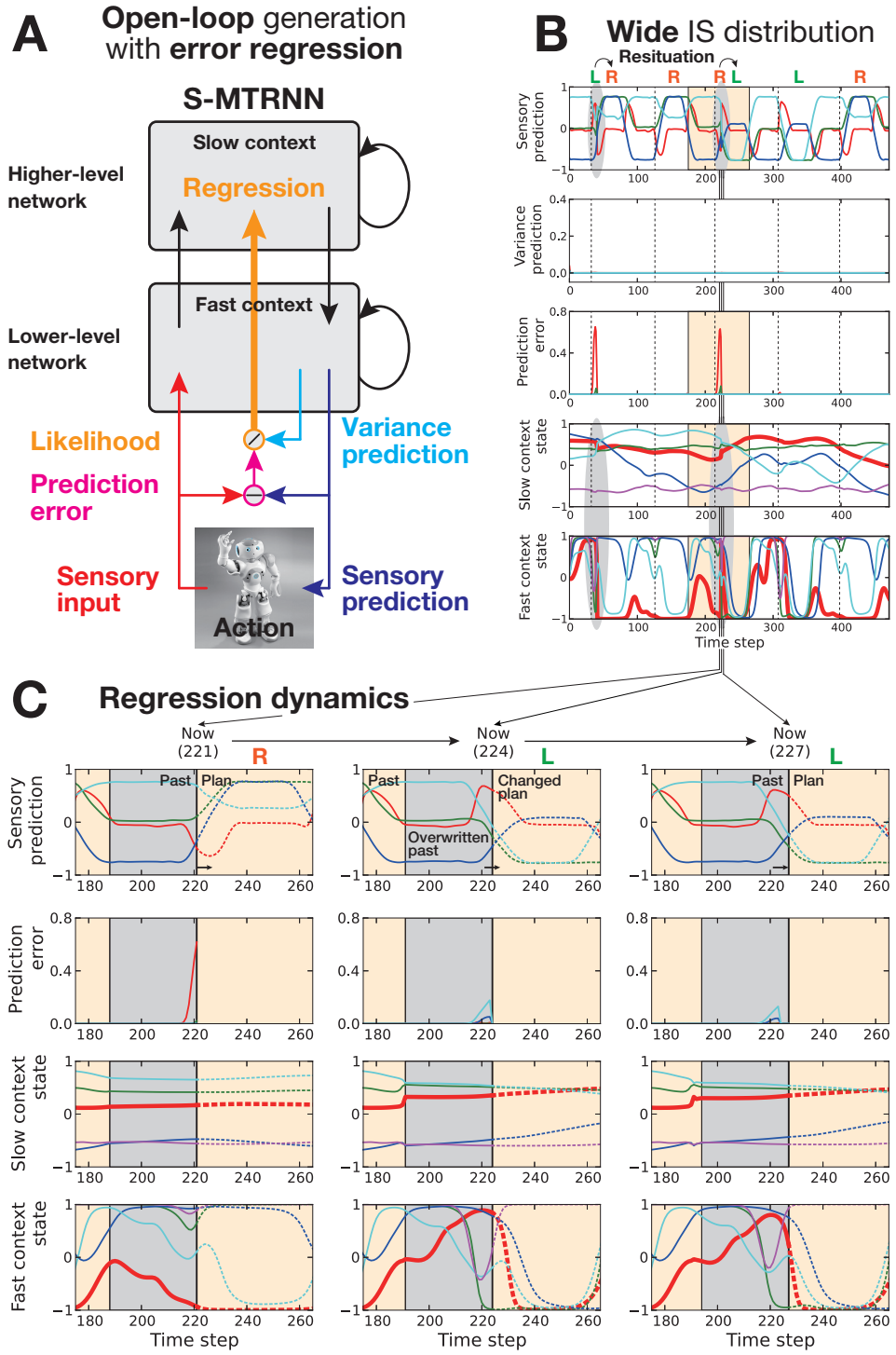

Figure 8. Open-loop generation scheme with error regression and time series obtained in the experiment. (A) Scheme for the open-loop generation with error regression. (B) Time series of one-step sensory predictions, variance predictions, prediction errors, SC states, and FC states during action generation with error regression for a robot using the network trained with the wide IS distribution. The format of these panels is the same as that in Fig. 6A and 6B. (C) Extracted states for time steps 175 to 265 corresponding to the states in the light orange area in B (here variance predictions are not shown). The current time steps labeled "Now" in the left hand, center, right hand panels are 221 (pre-modification phase), 224 (modification phase), and 227 (post-modification phase), respectively. Time windows are indicated by gray areas in each panel. The states, shown as solid lines, to the left of the window head correspond to past states at the current time; the states, shown as dashed lines, to the right of the window head correspond to future states at the current time. These past and future states can be overwritten and changed by the ERS at each time step. Note that the states shown in B correspond to the states at the current time step shown in C, which are the actual states before they are overwritten by the regression.

We now consider the effect of error regression on reaction times. We computed mean reaction times over the 10 sample networks trained with the wide IS distribution condition, each of which generated 32 sequences including all combinations of the five transitions of the two action primitives with and without the error regression mechanism. For computing the reaction times, the initial state optimized for the "LLRRL" sequence was adopted regardless of the other-robot's action sequences in the same manner as for the non-corresponding IS case shown in Fig. 7. As shown in Fig. 9, when the self-robot relied on only received sensory inputs (no error regression), the mean time for reaction to unanticipated external situations was 36.64 time steps (gray bar). By introducing the additional bottom-up recognition mechanism using ERS, the mean reaction time was reduced to just 8.22 time steps (red bar) with $SD = 2.75$. There is a significant difference between these mean reaction times ($t(18) = 8.54$, $p < 0.001$). This change shows that the internal contextual dynamics can be revised by means of interactions between the top-down intentional prediction and the bottom-up recognition of the actual behavior when deterministic predictive dynamics is used internally. The mean reaction time in the wide IS distribution condition with ERS is significantly shorter ($t(18) = 5.57$, $p < 0.001$) than that in the narrow IS distribution condition (17.80 time steps shown in Fig. 7). This difference can be attributed to the time steps required for each adaptation mechanism. In the narrow IS distribution condition, the adaptation to the other-robot's behavior is based on the received sensory (especially visual) inputs that gradually change the internal neural dynamics with the longer period than the other. In contrast, in the wide IS distribution condition with ERS, the adaptation is based on the forcible modification of the top-down prior which rapidly changes the internal neural dynamics in a discontinuous manner (see Fig. 8B) with the shorter period than the other. It is noted furthermore that this modification force becomes much larger in the case of the wide IS distribution because the error divided by the smaller variance estimated is used for the modification by means of BPTT algorithm.

4 Discussion and Conclusions

In this study, we hypothesized that different types of behavior generation, namely, sensory reflex and intentional proactive behavior generation, could be produced by a single neural mechanism depending on the learning condition. To test our hypothesis, we developed a novel hierarchical dynamic neural network model (S-MTRNN) and conducted robotics learning experiments in which one robot, called the self-robot, equipped with the S-MTRNN was required to interact cooperatively with another robot, called the other-robot. In the experiments, the other-robot generated action sequences which were observed by the self-robot as probabilistic transitions of action primitives. The self-robot acquired an internal or generative model that was able to generate predictions of visuo-proprioceptive states as well as their uncertainty levels (in terms of variances or inverse precisions) through its own actions and perceptual experiences. The experimental results demonstrated that sensory reflex behavior with a probabilistic prediction mechanism was developed when the initial precision characteristics of the higher-level network were not allowed in the learning process. In contrast, intentional proactive behavior with a deterministic prediction mechanism was developed when the initial precision was allowed. Furthermore, the results also demonstrated that each behavior generation scheme required different adaptation mechanisms, namely, simple sensory reflex and error regression, for revising the internal neural dynamics when the self-robot encountered unanticipated actions of the other-robot. In the following, we discuss the difference between the probabilistic model and the deterministic one by focusing on their learning capabilities and on their contributions to the development of different behavior generation schemes.

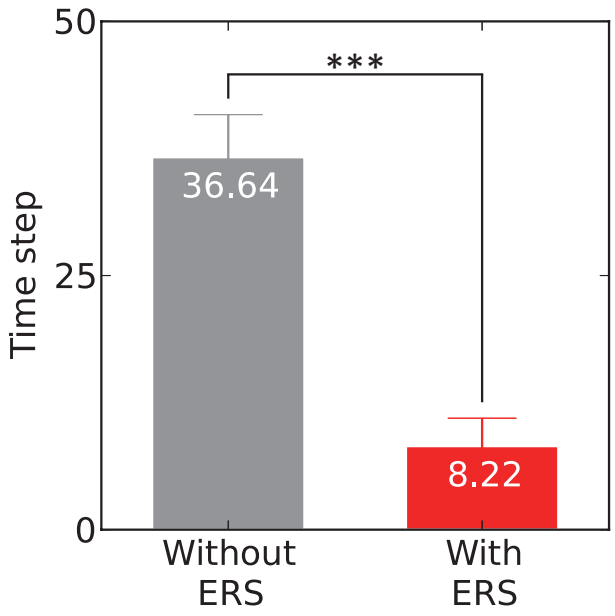

Figure 9. Reaction times of the self-robot during action generation with and without error regression scheme (ERS). Times are given for the results of the simulated self-robot's action generation using the network trained with the wide IS distribution condition. An arbitrarily selected initial state (values optimized for the "LLRRL" sequence) was given to the network regardless of the other-robot's action sequences. The bars and numbers in the graph correspond to mean reaction times over 10 trained sample networks, each of which generated 32 sequences including five transitions of the two action primitives. Error bars indicate the standard deviation. Stars indicate a significant difference (***: $p < 0.001$). The gray bar, for the case without ERS, corresponds to the rightmost bar in Fig. 7.

4.1 Learning of Perceptual Sequences Observed as Probabilistic Transitions

We demonstrated that the difference in the distribution of initial states of the higher-level network affected the learning of visuo-proprioceptive sequences observed as probabilistic transitions. When the S-MTRNN was trained with the narrow IS distribution condition, various combinatorial sequences were produced by stochastic dynamics with closed-loop generation in which self-generated noise with the estimated variances at each branching point determined the next primitive, as in Markov chains [51] (see Fig. 4A). On the other hand, when the network was trained with the wide IS distribution condition, all the learned sequences could be reproduced exactly by the top-down deterministic dynamics determined by the optimized initial states of the higher-level network [32, 41, 48] (see Fig. 4B).

The distinct models developed from our proposed S-MTRNN can be mechanized by means of a learning scheme using the maximization of a model likelihood in which sensory prediction error is divided by the predicted variance or weighted by the predicted precision. In a previous study using the conventional MTRNN model (without a variance prediction mechanism), Nishimoto et al. [48] demonstrated that the developmental process of the functional hierarchy emerged from the multiple timescale dynamics of the network and the learning scheme of prediction error minimization. During the process, a set of reusable primitives was first acquired in the lower-level network with fast dynamics. Then, sequential combinations of the primitives were acquired in the higher-level network with slow dynamics using initial precision characteristics to minimize prediction errors. By utilizing these mechanisms, Namikawa et al. [32] demonstrated that nondeterministic or probabilistic transition sequences can be embedded in deterministic chaotic dynamics, which were self-organized in the higher-level network, as pseudo-stochastic sequences. Our proposed S-MTRNN (which includes a variance prediction mechanism), however,

has another pathway to represent probabilistic event sequences. In short, the S-MTRNN can represent the probabilistic characteristics of event transitions by means of estimating the variance of the noise externally added to the output units. This means that if the network regards observed sequences as probabilistic transitions of primitives, the network ceases to minimize prediction error at a certain level and instead optimizes the variance. This has been identified by Friston [8, 24] as the mechanism of attention that controls the acquisition of prediction error or shapes precision-weighted prediction error. However, the uniqueness of the current study is that, if the network regards the same sequences as deterministic sequential combinations of the primitives, the network tries to minimize both the prediction error and variance by attributing the potential unpredictability to deterministic chaos generated by sensitivity to initial conditions. The importance is that in the former case a probabilistic model is developed and in the latter case a deterministic model is developed, even though the same network and the same training sequences were employed.

4.2 Sensory Reflex Behavior versus Intentional Proactive Behavior

In an actual action generation test, we confirmed that the difference in the modeling of observed perceptual events was essential for the development of behavior generation schemes. When we employed the S-MTRNN trained with the narrow IS distribution condition (probabilistic model), the self-robot generated sensory reflex behavior. On the other hand, when we employed the S-MTRNN trained with the wide IS distribution condition (deterministic model), the robot generated intentional proactive behavior. These results originating from the different parameter settings are in general agreement with simulations of active inference [10]. Both in our approach and in active inference, because action generation can be understood as fulfilling predictions (prior expectations) about proprioceptive states, the type of generative model induced by the parameter setting works as an essential factor for the development of behavior generation schemes.

During the generation of the sensory reflex behavior, the network predicted a large variance (small precision) and generated a small prediction error at each branching point (see Fig. 6A). These results indicate that the network predicted a neutral sensory state at branching points. Therefore, there is no difference between reaction times of the self-robot for the corresponding and non-corresponding IS cases (see the narrow IS distribution condition in Fig. 7). It is believed that the sensory reflex behavior resulted from the allowance of uncertainty for sensory predictions and the development of sensitive sensory structures at each branching point.

In contrast, during the generation of the intentional proactive behavior, the network tried to generate exact sensory predictions with almost zero variance. In this case, when the intention of the self-robot corresponded to that of the other-robot, cooperative interactions could be achieved smoothly without any time delay (see the corresponding IS case in the wide IS distribution condition in Fig. 7). However, when a discrepancy between their intentions occurred, erratic behavior by the self-robot was observed (see Fig. 6B) and long reaction times were required (see the non-corresponding IS case in the wide IS distribution condition in Fig. 7). In the previous subsection, the importance of the multiple timescale dynamics for developing a functional hierarchy was discussed. From the viewpoint of learning, the slowly changing higher-level dynamics are essential for reproducing combinatorial sequences in a deterministic manner. While at the same time, the slow dynamics that are not affected by sensory inputs directly have a negative effect on generating adaptive sensory reflex behavior. As an alternative mechanism to the sensory reflex, we proposed a novel ERS, which can be considered as an extension of the predictive coding schemes used in RNNPB [5, 7] and active inference [9–11], for revising the slow dynamics in a forcible manner by means of propagating precision-weighted prediction errors from the lower-level to the higher-level network. It should be noted that the aforemen-

tioned attention mechanism for controlling the acquisition of prediction error is utilized in this error regression process.

In an action generation test with ERS, we confirmed that the self-robot controlled by the S-MTRNN trained with the wide IS distribution was able to generate cooperative behavior when the robot encountered unanticipated actions by the other-robot. This adaptation was achieved because the ERS slightly modulated the neural activity of the higher-level network in order to maximize the model likelihood or to minimize precision-weighted prediction error. In Figs. 8B and 9, we can see the effect of the ERS on the revision of internal neural activity and on the reaction time of the self-robot, respectively. These results indicate the importance of the interaction between the top-down process for anticipating future states and the bottom-up process for recognizing perceptual reality during intentional proactive behavior generation [5,52]. This interactive process with ERS corresponds to the error monitoring process that might be mediated by the parietal cortex [53].

Different types of computational models have been employed to implement the robot control architectures for sensory reflex behavior [54, 55] and intentional proactive behavior [5, 32]. In contrast, in this study we have demonstrated that these different behavioral schemes can be produced by a single neural mechanism. Because the learned action sequences were simple and each behavioral scheme was separately developed depending on the learning condition in this study, our next step is to consider these aspects as detailed in the next subsection.

4.3 Limitations and Future Work

Several issues remain to be examined in future studies. In this paper, we have focused on the learning of visuo-proprioceptive sequences observed as probabilistic transitions, under the two distinct learning conditions of the narrow and wide IS distributions. In a set of visuo-proprioceptive sequences used for the training, all the primitives were clean and concatenated with equal probability. In other words, we have not considered a situation where each primitive includes fluctuations [43] or a situation where a certain statistical bias (e.g. more movements to the left than those to the right) is given for a set of sequences [32]. Through the experiments, we have demonstrated the robustness of the proposed model to the uncertain situational changes (probabilistic transitions of perceptual events) by means of sensory reflex behavior in the narrow IS distribution condition and intentional proactive behavior with ERS in the wide IS distribution condition. We should also consider the aforementioned situations for further evaluation of the proposed schemes in future study. In the context of the IS distribution, we have not employed a distribution with an intermediate variance between the two conditions or a mixture distribution. These considerations might be important to discuss the adaptive modulation between sensory reflex behavior and intentional proactive behavior, each of which was separately developed depending on the two distinct learning conditions in this study. Future work should therefore include follow-up work designed to evaluate these cases.

Another issue is the learning method. Our proposed S-MTRNN was trained in an offline manner using the gradient ascent method with BPTT. The time constants of the FC and SC units were manually set for the network training. Although we have a guideline to set these parameters as described in Section 2.4, the current scheme requires some amounts of trial and error for tuning them. Future study should therefore consider a scheme of automatic optimization of the time constants by utilizing the BPTT scheme. One might assume that the usage of the offline learning method limits the utility of our proposed scheme for more practical applications. We consider that the offline learning corresponds to the consolidation learning [56, 57] that enables cognitive agents to consolidate perceptual experiences into a long-term memory. In addition to the scheme of the offline learning and online adaptive behavior

generation demonstrated in the present study, the aspect of one-shot learning or online learning should also be considered.

In future studies, we will consider two essential problems. First, the current experiment did not consider the possibility that robots can change the world by acting on it. We have described how our proposed S-MTRNN can learn to generate and recognize visuo-proprioceptive sequences under the principle of model likelihood maximization which is formally equivalent to the principle of free energy minimization (because the free energy is an upper bound on the negative logarithm of model likelihood). In particular, during action generation, the likelihood can be maximized by changing both the prediction state and the sensory signals that shape prediction error. However, in the present study, because the object was held by the other-robot, the self-robot was unable to change its visual input in response to its predictions. However, by changing the experimental setup, active sensory sampling can be conducted by the self-robot. For example, when the robot encounters an unanticipated situation, it can change the prediction to fit the received sensory signals and change the sensory signals to fit the prediction for minimizing prediction error. As noted in [8–11], action is the only way to change sensory signals for error minimization, and thus active sampling should be considered.

The other issue is the consideration of bidirectionality in cooperative interactions. In the present study for simplicity, only the self-robot was required to change its actions and the other-robot's behavior was automatically controlled to generate fixed action sequences that were not affected by the behavior of the self-robot. That is, the interaction was unidirectional not bidirectional. However, bidirectionality in social interactions is essential for understanding the mechanism of turn-taking behavior [58] and the acquisition of "nested" internal models in which an internal model includes itself [59,60]. Therefore, in future studies, we plan to examine cases of mutual interactions by implementing the proposed model on two interacting robots. This will allow us to investigate autonomous mechanisms for the formation and manipulation of communicative interactions between cognitive agents.

Appendix A Gradients

The gradient $\frac{\partial \ln L_{\text{out}}}{\partial \boldsymbol{\theta}_{\text{share}}}$ and $\frac{\partial \ln L_{\text{all}}}{\partial \boldsymbol{\theta}_{\text{init}}}$ for each learnable parameter can be obtained [43] by the conventional back-propagation through time (BPTT) method [47]:

$$\frac{\partial \ln L_{\text{out}}}{\partial w_{ij}} = \begin{cases} \dfrac{1}{\tau_i} \displaystyle\sum_{s \in I_{\text{S}}} \sum_{t=1}^{T^{(s)}} x_{t,j}^{(s)} \frac{\partial \ln L_{\text{out}}}{\partial u_{t,i}^{(s)}} & (i \in I_{\text{FC}} \wedge j \in I_{\text{I}}), \\ \dfrac{1}{\tau_i} \displaystyle\sum_{s \in I_{\text{S}}} \sum_{t=1}^{T^{(s)}} c_{t-1,j}^{(s)} \frac{\partial \ln L_{\text{out}}}{\partial u_{t,i}^{(s)}} & (i \in I_{\text{FC}} \cup I_{\text{SC}} \wedge j \in I_{\text{FC}} \cup I_{\text{SC}}), \\ \displaystyle\sum_{s \in I_{\text{S}}} \sum_{t=1}^{T^{(s)}} c_{t,j}^{(s)} \frac{\partial \ln L_{\text{out}}}{\partial u_{t,i}^{(s)}} & (i \in I_{\text{O}} \cup I_{\text{V}} \wedge j \in I_{\text{FC}}), \end{cases} \tag{10}$$

$$\frac{\partial \ln L_{\text{out}}}{\partial b_i} = \begin{cases} \dfrac{1}{\tau_i} \displaystyle\sum_{s \in I_{\text{S}}} \sum_{t=1}^{T^{(s)}} \frac{\partial \ln L_{\text{out}}}{\partial u_{t,i}^{(s)}} & (i \in I_{\text{FC}} \cup I_{\text{SC}}), \\ \displaystyle\sum_{s \in I_{\text{S}}} \sum_{t=1}^{T^{(s)}} \frac{\partial \ln L_{\text{out}}}{\partial u_{t,i}^{(s)}} & (i \in I_{\text{O}} \cup I_{\text{V}}), \end{cases} \tag{11}$$

$$\frac{\partial \ln L_{\rm out}}{\partial u_{t,i}^{(s)}} = \begin{cases} \left\{1-(c_{t,i}^{(s)})^2\right\}\left\{\displaystyle\sum_{k\in I_{\rm FC}\cup I_{\rm SC}} \frac{w_{ki}}{\tau_k}\frac{\partial \ln L_{\rm out}}{\partial u_{t+1,k}^{(s)}} + \sum_{k\in I_{\rm O}\cup I_{\rm V}} w_{ki}\frac{\partial \ln L_{\rm out}}{\partial u_{t,k}^{(s)}}\right\} \\ \quad + \left(1-\dfrac{1}{\tau_i}\right)\dfrac{\partial \ln L_{\rm out}}{\partial u_{t+1,i}^{(s)}} & (0\le t \wedge i\in I_{\rm FC}\cup I_{\rm SC}), \\ -\dfrac{y_{t,i}^{(s)}-\hat{y}_{t,i}^{(s)}}{v_{t,i}^{(s)}}\left\{1-(y_{t,i}^{(s)})^2\right\} & (1\le t \wedge i\in I_{\rm O}), \\ -\dfrac{1}{2}+\dfrac{(y_{t,i}^{(s)}-\hat{y}_{t,i}^{(s)})^2}{2v_{t,i}^{(s)}} & (1\le t \wedge i\in I_{\rm V}), \end{cases} \tag{12}$$

$$\frac{\partial \ln L_{\rm all}}{\partial u_{0,i}^{(s)}} = \frac{\partial \ln L_{\rm out}}{\partial u_{0,i}^{(s)}} - \frac{1}{\sigma_{\rm IS}^2}(u_{0,i}^{(s)}-u_i) \qquad (i\in I_{\rm SC}), \tag{13}$$

$$\frac{\partial \ln L_{\rm all}}{\partial u_i} = \sum_{s\in I_S}\frac{1}{\sigma_{\rm IS}^2}(u_{0,i}^{(s)}-u_i) \qquad (i\in I_{\rm SC}). \tag{14}$$

Appendix B Acceleration of Network Training

To accelerate network training, we employed the adaptive learning rate scheme [32, 38]. In this scheme, the learning rate α is also optimized during the learning process based on the change of a total error before and after updating parameters in the following way:

1. For each learning step n, updated parameters are tentatively computed by (7) and (8) using the learning rate α ($\alpha_{\rm share}$ or $\alpha_{\rm init}$), and the total error rate r defined by

$$r = \frac{\displaystyle\sum_{s\in I_{\rm S}} E^{(s)}(\boldsymbol{\theta}'_{\rm share}(n), \boldsymbol{\theta}'_{\rm init}(n))}{\displaystyle\sum_{s\in I_{\rm S}} E^{(s)}(\boldsymbol{\theta}_{\rm share}(n), \boldsymbol{\theta}_{\rm init}(n))}, \tag{15}$$

$$E^{(s)}(\boldsymbol{\theta}_{\rm share}, \boldsymbol{\theta}_{\rm init}) = \sum_{t=1}^{T^{(s)}}\sum_{i\in I_{\rm O}}(y_{t,i}^{(s)}(\boldsymbol{\theta}_{\rm share}, \boldsymbol{\theta}_{\rm init}) - \hat{y}_{t,i}^{(s)})^2, \tag{16}$$

   where $(\boldsymbol{\theta}_{\rm share}(n), \boldsymbol{\theta}_{\rm init}(n))$ and $(\boldsymbol{\theta}'_{\rm share}(n), \boldsymbol{\theta}'_{\rm init}(n))$ are the current parameters and the tentatively updated parameters, respectively, is also computed.

2. If $r_{\rm th} < r$, then α is replaced with $\alpha_{\rm dec}\alpha$, and the procedure returns to step (1) without updating the current parameters $(\boldsymbol{\theta}_{\rm share}(n), \boldsymbol{\theta}_{\rm init}(n))$. Otherwise the procedure moves to step (3)

3. If $r < 1$, then α is replaced with $\alpha_{\rm inc}\alpha$. The current parameters $(\boldsymbol{\theta}_{\rm share}(n), \boldsymbol{\theta}_{\rm init}(n))$ are updated with $(\boldsymbol{\theta}'_{\rm share}(n), \boldsymbol{\theta}'_{\rm init}(n))$ and the procedure moves to the next learning step $n+1$.

In the present study, we used $r_{\rm th} = 1.1$, $\alpha_{\rm dec} = 0.7$, and $\alpha_{\rm inc} = 1.05$, which were determined by reference to [32, 38]. We put the upper limit of 1000 for this iteration process at each learning step.

Appendix C Reaction Time Measurement

To measure reaction times, we conducted numerical simulations instead of actual cooperative interactions by reutilizing the recorded 32 pattern visuo-proprioceptive sequences used in the learning phase. During generation phases, the network received sensory input values derived from recorded data for both the two-dimensional object position and the automatically controlled two-dimensional head joint angles, and from its own predictions of four-dimensional arm joint angles. In the experiments for generating forward prediction dynamics, we computed differences between the arm movement state predicted by the trained network and the state recorded in the sequence data at each time step. We measured reaction times by counting time steps before the computed differences became sufficiently small. More specifically, we computed the sum of the mean squared errors of predicted four-dimensional arm joint angle state $E_t^{(s)}$ at each time step t for each sequence s:

$$E_t^{(s)} = \frac{1}{2} \sum_{i \in I_{\mathrm{A}}} (y_{t,i}^{(s)} - \hat{y}_{t,i}^{(s)})^2, \tag{17}$$

where $I_{\mathrm{A}} \subset I_{\mathrm{O}}$ is the index set for the output units provided for the prediction of arm joint angles. After computing the above values, we counted the time steps from when the object was moved by the other-robot (branching point) until $E_t^{(s)}$ became less than a threshold $E_{\mathrm{th}} = 0.01$ (starting point of cooperative behavior), which was empirically determined. These computations were conducted for each generation condition, as shown in Figs. 7 and 9, by using 10 sample networks with differently randomized initial parameters trained with each learning condition. The values shown in the figures are mean values over the 10 trained sample networks, each of which generated 32 sequences including all combinations of the five transitions of the two action primitives (160 branches).

References

[1] Daniel M Wolpert, Zoubin Ghahramani, and Michael I Jordan. An internal model for sensorimotor integration. *Science*, 269(5232):1880–1882, 1995.

[2] Daniel M Wolpert, R Chris Miall, and Mitsuo Kawato. Internal models in the cerebellum. *Trends in Cognitive Sciences*, 2(9):338–347, 1998.

[3] R P Rao and D H Ballard. Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1):79–87, 1999.

[4] Andy Clark. Whatever next? Predictive brains, situated agents, and the future of cognitive science. *The Behavioral and Brain Sciences*, 36(3):181–204, 2013.

[5] Jun Tani. Learning to generate articulated behavior through the bottom-up and top-down interaction processes. *Neural Networks*, 16(1):11–23, 2003.

[6] Masato Ito and Jun Tani. On-line imitative interaction with a humanoid robot using a dynamic neural network model of a mirror system. *Adaptive Behavior*, 12(2):93–115, 2004.

[7] Masato Ito, Kuniaki Noda, Yukiko Hoshino, and Jun Tani. Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model. *Neural Networks*, 19(3):323–337, 2006.

[8] Karl Friston. The free-energy principle: A rough guide to the brain? *Trends in Cognitive Sciences*, 13(7):293–301, 2009.

[9] Karl J Friston, Jean Daunizeau, and Stefan J Kiebel. Reinforcement learning or active inference? *PloS One*, 4(7):e6421, 2009.

[10] Karl J Friston, Jean Daunizeau, James Kilner, and Stefan J Kiebel. Action and behavior: A free-energy formulation. *Biological Cybernetics*, 102(3):227–260, 2010.

[11] Karl Friston, Jérémie Mattout, and James Kilner. Action understanding and active inference. *Biological Cybernetics*, 104(1-2):137–160, 2011.

[12] Jakob Hohwy. *The Predicrive Mind*. Oxford University Press, Oxford, 2013.

[13] Michael I. Jordan. Attractor dynamics and parallelism in a connectionist sequential machine. In *Proceedings of the 8th Annual Conference of the Cognitive Science Society*, pages 531–546, Amherst, MA, August 1986.

[14] Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280, 1989.

[15] Jeffrey L. Elman. Finding structure in time. *Cognitive Science*, 14(2):179–211, 1990.

[16] Jordan B. Pollack. The induction of dynamical recognizers. *Machine Learning*, 7(2-3):227–252, 1991.

[17] Mark M Churchland, John P Cunningham, Matthew T Kaufman, Stephen I Ryu, and Krishna V Shenoy. Cortical preparatory activity: Representation of movement or first cog in a dynamical machine? *Neuron*, 68(3):387–400, 2010.

[18] R. Nishimoto, J. Namikawa, and J. Tani. Learning multiple goal-directed actions through self-organization of a dynamic neural network model: A humanoid robot experiment. *Adaptive Behavior*, 16(2-3):166–181, 2008.

[19] P Dayan, G E Hinton, R M Neal, and R S Zemel. The Helmholtz machine. *Neural Computation*, 7(5):889–904, 1995.

[20] David C Knill and Alexandre Pouget. The Bayesian brain: The role of uncertainty in neural coding and computation. *Trends in Neurosciences*, 27(12):712–719, 2004.

[21] Karl Friston. A theory of cortical responses. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 360(1456):815–836, 2005.

[22] Karl Friston, James Kilner, and Lee Harrison. A free energy principle for the brain. *Journal of Physiology*, 100(1-3):70–87, 2006.

[23] Jun Tani. Self-organization and compositionality in cognitive brains: A neurorobotics study. *Proceedings of the IEEE*, 102(4):586–605, April 2014.

[24] Harriet Feldman and Karl J Friston. Attention, uncertainty, and free-energy. *Frontiers in Human Neuroscience*, 4(215):1–23, 2010.

[25] Jakob Hohwy. Attention and conscious perception in the hypothesis testing brain. *Frontiers in Psychology*, 3(96):1–14, 2012.

[26] Hanneke E M den Ouden, Peter Kok, and Floris P de Lange. How prediction errors shape perception, attention, and motivation. *Frontiers in Psychology*, 3(548):1–12, 2012.

[27] Eleanor J. Gibson and Anne D. Pick. *An Ecological Approach to Perceptual Learning and Development*. Oxford University Press, New York, NY, 2000.

[28] Karl Friston. Hierarchical models in the brain. *PLoS Computational Biology*, 4(11):e1000211, 2008.

[29] Marcel Brass and Patrick Haggard. The what, when, whether model of intentional action. *The Neuroscientist*, 14(4):319–325, 2008.

[30] Brian J White, Dirk Kerzel, and Karl R Gegenfurtner. Visually guided movements to color targets. *Experimental Brain Research*, 175(1):110–126, 2006.

[31] Andrew E Welchman, James Stanley, Malte R Schomers, R Chris Miall, and Heinrich H Bülthoff. The quick and the dead: When reaction beats intention. *Proceedings of the Royal Society B: Biological Sciences*, 277(1688):1667–1674, 2010.

[32] Jun Namikawa, Ryunosuke Nishimoto, and Jun Tani. A neurodynamic account of spontaneous behaviour. *PLoS Computational Biology*, 7(10):e1002221, 2011.

[33] John R. Searle. *Intentionality: An Essay in the Philosophy of Mind*. Cambridge University Press, New York, NY, 1983.

[34] Nathaniel D Daw, Yael Niv, and Peter Dayan. Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature Neuroscience*, 8(12):1704–1711, 2005.

[35] Mehdi Khamassi and Mark D Humphries. Integrating cortico-limbic-basal ganglia architectures for learning model-based and model-free navigation strategies. *Frontiers in Behavioral Neuroscience*, 6(79):1–19, 2012.

[36] Erwan Renaudo, Benoît Girard, Raja Chatila, and Mehdi Khamassi. Design of a control architecture for habit learning in robots. In Nathan F. Leopra, Anna Mura, Holger G Krapp, Paul F M J Verschure, and Tony J. Prescott, editors, *Biomimetic and Biohybrid Systems*, pages 249–260. Springer International Publishing AG, Cham (ZG), Switzerland, 2014.

[37] Jun Tani and Naohiro Fukumura. Embedding a grammatical description in deterministic chaos: An experiment in recurrent neural learning. *Biological Cybernetics*, 72(4):365–370, 1995.

[38] Jun Namikawa and Jun Tani. Learning to imitate stochastic time series in a compositional way by chaos. *Neural Networks*, 23(5):625–638, 2010.

[39] J M Fuster. The prefrontal cortex–an update: Time is of the essence. *Neuron*, 30(2):319–333, 2001.

[40] Matthew M Botvinick. Hierarchical models of behavior and prefrontal function. *Trends in Cognitive Sciences*, 12(5):201–208, 2008.

[41] Yuichi Yamashita and Jun Tani. Emergence of functional hierarchy in a multiple timescale neural network model: A humanoid robot experiment. *PLoS Computational Biology*, 4(11):e1000220, 2008.

[42] Jun Namikawa, Ryunosuke Nishimoto, Hiroaki Arie, and Jun Tani. Synthetic approach to understanding meta-level cognition of predictability in generating cooperative behavior. In Yoko Ymaguchi, editor, *Advances in Cognitive Neurodynamics (III)*, pages 615–621. Springer Netherlands, 2013.

[43] Shingo Murata, Jun Namikawa, Hiroaki Arie, Shigeki Sugano, and Jun Tani. Learning to reproduce fluctuating time series by inferring their time-dependent stochastic properties: Application in robot learning via tutoring. *IEEE Trans. Autonomous Mental Development*, 5(4):298–310, December 2013.

[44] Boon Hwa Tan, Huajin Tang, Rui Yan, and Jun Tani. Flexible and robust robotic arm design and skill learning by using recurrent neural networks. In *Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 522–529, Chicago, IL, September 2014.

[45] Stefan J. Kiebel, Jean Daunizeau, and Karl J. Friston. A hierarchy of time-scales and the brain. *PLoS Computational Biology*, 4(11):e1000209, 2008.

[46] Uri Hasson, Janice Chen, and Christopher J. Honey. Hierarchical process memory: Memory as an integral component of information processing. *Trends in Cognitive Sciences*, 19(6):304–313, 2015.

[47] David E. Rumelhart, G. E. Hinton, and Ronald J. Williams. Learning internal representations by error propagation. In David E. Rumelhart and D. McClelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, pages 318–362. The MIT Press, Cambridge, MA, 1986.

[48] Ryunosuke Nishimoto and Jun Tani. Development of hierarchical structures for actions and motor imagery: A constructivist view from synthetic neuro-robotics study. *Psychological Research*, 73(4):545–558, 2009.

[49] Yuichi Yamashita and Jun Tani. Spontaneous prediction error generation in schizophrenia. *PloS One*, 7(5):e37843, 2012.

[50] Tom Ziemke, Dan-Anders Jirenhed, and Germund Hesslow. Internal simulation of perception: A minimal neuro-robotic model. *Neurocomputing*, 68:85–104, 2005.

[51] Leonard E. Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The Annals of Mathematical Statistics*, 37(6):1554–1563, 1966.

[52] Jun Tani. Autonomy of self at criticality: The perspective from synthetic neuro-robotics. *Adaptive Behavior*, 17(5):421–443, 2009.

[53] Michel Desmurget, Karen T Reilly, Nathalie Richard, Alexandru Szathmari, Carmine Mottolese, and Angela Sirigu. Movement intention after parietal cortex stimulation in humans. *Science*, 324(5928):811–813, 2009.

[54] Rolf Pfeifer and Christian Scheier. Sensory—motor coordination: The metaphor and beyond. *Robotics and Autonomous Systems*, 20(2-4):157–178, 1997.

[55] J. Leitner, M. Frank, A. Fosrter, and J. Schmidhuber. Reactive reaching and grasping on a humanoid towards closing the action-perception loop on the iCub. In *Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics*, pages 102–109, Vienna, September 2014.

[56] J L McClelland, B L McNaughton, and R C O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457, 1995.

[57] Lynn Nadel and Morris Moscovitch. Memory consolidation, retrograde amnesia and the hippocampal complex. *Current Opinion in Neurobiology*, 7(2):217–227, 1997.

[58] Takashi Ikegami and Hiroyuki Iizuka. Turn-taking interaction as a cooperative and co-creative process. *Infant Behavior & Development*, 30(2):278–288, 2007.

[59] Makoto Taiji and Takashi Ikegami. Dynamics of internal models in game players. *Physica D: Nonlinear Phenomena*, 134(2):253–266, 1999.

[60] Wako Yoshida, Ben Seymour, Karl J Friston, and Raymond J Dolan. Neural mechanisms of belief inference during cooperative games. *The Journal of Neuroscience*, 30(32):10744–10751, 2010.