Journal of Consciousness Studies, Vol.11, No.9, pp.5-24, 2004.

The Dynamical Systems Accounts for Phenomenology of Immanent Time: An Interpretation by Revisiting a Robotics Synthetic Study

Jun Tani

Brain Science Institute, RIKEN

2-1 Hirosawa, Wako-shi, Saitama, 351-0198 Japan

Tel +81-48-467-6467, FAX +81-48-467-7248

E-mail tani@brain.riken.go.jp

Accepted for publication in Journal of Consciousness Studies.

Abstract

This paper discusses possible correspondences between the dynamical systems characteristics observed in our previously proposed cognitive model and phenomenological accounts of the immanent time considered by Edmund Husserl. Our simulation experiments in the anticiparatory learning of robot showed that encountering sensory-motor flow can be learned as segmented into chunks of reusable primitives with accompanying dynamic shifting between coherences and incoherences in local modules. It is considered that the sense of the objective time might appear when the continuous sensory-motor flow input to the robot is reconstructed into compositional memory structures through the articulation processes described.

1 Introduction

When a person behaves and generates continuous sensory-motor flow resultantly, he may not remember the whole behavior as a continuous flow like a video tape, but rather as a segmented sequence. For example, when I attempt to recollect how I turned on the TV in front of me just minutes ago, I only remember it as a linear sequence of combining behavioral events as like – I first sat down on the couch, then grasped the TV remote control found on the side table, and then pushed the power-on button while pointing toward the TV.

In studies of motor systems, some (Arbib, 1981; Feldman, 1980) proposed that various complex behavior sequences can be generated by flexibly combining a set of reusable behavior scheme which have been acquired in advance. In robotics applications, it was reported that a complex task of objects handling by a robot arm can be achieved by segmenting behavior sequences by using a set of behavior scheme that are prepared by programming(Kuniyoshi, Inaba, & Inoue, 1994). If the idea of the segmentation is essential, a question is that how such a set of behavior scheme can be learned from iterative sensory-motor experiences.

While we investigated possible neuronal mechanisms for the segmentation of sensory-motor flow we noticed that this problem is also related to an essential phenomenological question by Husserl (1964): how objective time could be constituted out of the subjective flow of temporality. Although the temporality is experienced as as a part of flow in the deep level of phenomenology, it does as temporal objects and events in the shallow level. Here, it is noted that objective time by Husserl (1964) does not mean time modeled in physics, but does for the one phenomenologically experienced when recalling rather objectively our own episodes. As illustrated previously in the TV example, in recalling past experiences, a temporal image of the past can be reactivated as a linear sequence of discrete events, instead of as a replay of the original continuous flow of our impression. After all, the flow itself cannot be consciously manipulable unless it is somehow segmented into a set of identifiable objects. Hence, we have to investigate by what sort of mechanism the flow can be reconstructed into consciously manipulatable structures.

We consider how this mechanism could be described by using adequate synthetic approaches especially using the dynamical systems approach (Kelso, 1988; Beer, 1995; Gelder, 1998). The dynamical systems approach attempts to describe the underlying mechanism of cognition in macro-scopic views by using the dynamical systems language (Beer, 1995) such as phase transitions, attractor, coherence, entrainment, etc. There have been some lines of related research that attempt to describe phenomenological observations. Varela (1999) proposed that nonlinear dynamics theory can be used as the formal descriptive tool for the phenomenon. By using the phenomenon of the spontaneous flipping of a Necker-cube as an example, he explained that the dynamic properties of intermittent chaos, which is characterized by its spontaneous shifts between static and rapid transition modes, could explain the paradox of continuous, yet segmented time perception. Tani (1998) also attempted to explain a possible dynamical system's structure behind the phenomenon of momentary self by conducting experiments with a real autonomous robot. In this research it is claimed that the "self" emerges momentarily when the coupled dynamics between the internal neural network and the environment shifts from coherent to incoherent dynamics. When everything proceeds as anticipated in the coherent phase, there is no distinction between the self and the environment in the coupled dynamics. However, the self can be perceived as separate from the environment when something goes wrong, in conflict with the system's anticipation, leading to the incoherent phase. It is argued that the open dynamics, which causes autonomous fluctuations between the coherent and the incoherent phases, may represent the structure of the "self". This corresponds to William James's saying that the stream of consciousness is segmented as like a bird making of an alternation of flights and perchings(James, 1890).

In this paper, first our proposed neural network model (Tani & Nolfi, 1998; ?), that is aimed at achieving hierarchical modular learning of sensory-motor flow, is reviewed. Then, the paper will review an experiment using a simulated robot(Tani & Nolfi, 1998, 1999) where it is shown how a mobile robot can learn to recognize the sensory-motor flow as segmented in multiple levels as the internal structure in the network is self-organized. In the end, with focusing on the results obtained in our synthetic studies we will discuss the dynamical systems account for phenomenology of immanent time.

2 Model

This section reviews the neural network proposed by Tani and Nolfi (1998, 1999)and also explain how the model is applied to a specific robot learning task.

2.1 Anticiparatory learning by a mobile robot

Before going to detailed descriptions of our models, it is better to explain the objectives of sensory-motor learning in our experimental tasks. We focus on the navigation capability of a mobile robot that moves around certain environment. The simulated mobile robot is equipped with a set of range sensors by which distances to its surrounding environment can be measured in multiple directions. The robot is assumed to maneuver in the workspace by changing its motion direction while its motion speed is kept as a constant. Figure 1 shows an example of robot travel in the employed

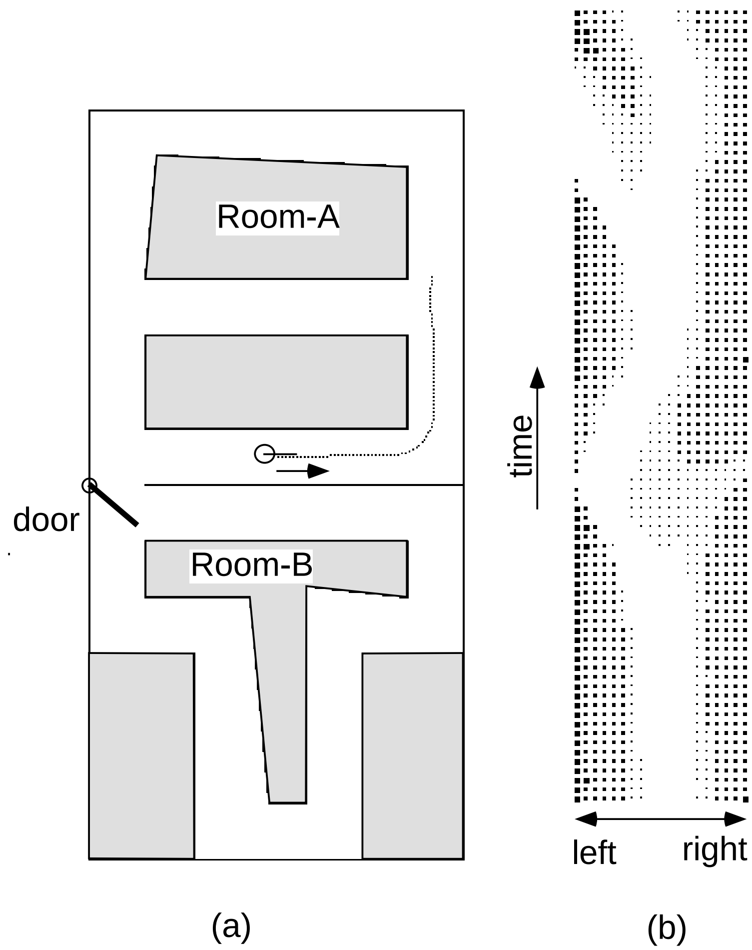

Figure 1: (a) Simulation workspace consisting of two rooms connected by a door. (b) the time development of the simulated range image while the robot traveled. Redrawing with permission, from (Tani & Nolfi, 1999) copy right (1999) Elsevier Science.

task environment and its corresponding sensory-motor flow. The goal in this setting is that the robot becomes able to anticipate coming sensory-motor flow based on the current situation/context of the robot through iterative learning. This can be done by utilizing the ideas of the forward model (Uno, Kawato, & Suzuki, 1989; Jordan & Rumelhart, 1992) in which the sensory state in the next step is anticipated with the sensory state and the motor commands given in the current step. In our experiment setting it is assumed that collision avoidance maneuvering controller is prepared as an innate function. We employed a variant of the potential method (Khatib, 1986) by which the robot tends to move toward the direction of open space sensed by the range sensors. This controller works as independent of the neural network functions described in the following subsections. The Anticiparatory learning takes a place in a given environment based on this pre-implemented collision-free maneuvering control scheme. What we assume is that certain internal model of the environment will be self-organized in the the forward model structure as the robot repeatedly travels around the same environment. After learning the network should become able to predict encountering sensory-motor inputs while it travels in the same environment. . The central issue, however, in this experiment is to see how the internal structures of segmenting continuous sensory-motor flow appear in the course of this prediction learning.

2.2 Neural network model

The following three sections explain the basic functions, the architecture and the computation algorithm for the neural network model employed in the experiments. Our proposal is to use multiple module RNNs, each of which competes to become an expert at predicting the sensory-motor flow for a specific behavior. This idea is inspired by the mixture of experts first expounded by Jacobs and Jordan (1991). The experts achieve their status through learning processes. For example, one module RNN would win in predicting the sensory-motor flow while traveling around a corner; another would win while following a straight wall. The switching between the winning RNN modules actually corresponds to the temporal segmentation of the sensory-motor flow. The essential point in this scenario is that the segmentations take place by means of pronounced changes in the observed dynamical structure in the sensory-motor flow, rather than just in temporal differences in the sensory-motor state. These highly pronounced changes correspond to switching between the dynamical functions, each of which is embedded in an RNN through having learned the specific sensory-motor flow. One might ask how each RNN can choose to learn its corresponding sensory-motor flow. The speciality of each module is determined during the processes of on-line learning. The competition between the modules during the simultaneous processes of recognition and learning result in generating their specialties. The next section will introduce a new architecture called the mixture of RNN experts which has been extended from the original idea of the mixture of experts (Jacobs & Jordan, 1991).

2.3 Architecture

Figure 2 shows the proposed architecture for the mixture of RNN experts (MRE) which is used for the prediction-learning of the sensory-motor flow.

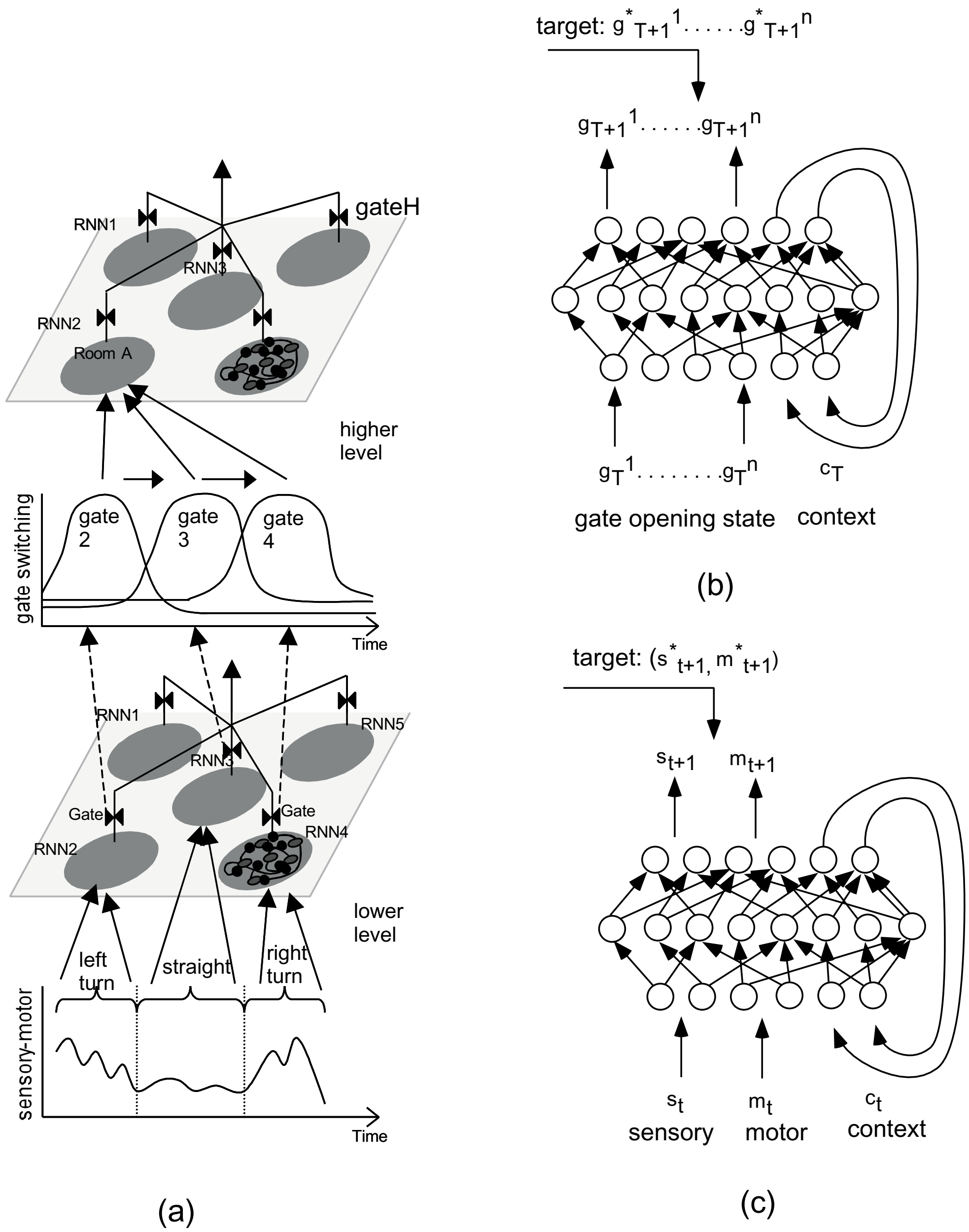

Figure 2: The complete architecture of the mixture of RNN experts for anticiparatory learning. (a) In hierarchical learning architecture, a gate of specific module corresponding to current sensory-motor is opened by turns in the lower level, then a specific module in the higher level is activated by receiving this gate opening sequence pattern, (b) RNN module for learning the gate opening dynamics in the higher level, and (c) details of each RNN module for learning the sensory-motor flow in the lower level.

Fig. 2(a) shows a hierarchical architecture consisting of two levels; more levels are possible in general. Each RNN module in the lower level receives the sensory-motor inputs, $X_t : (s_t, m_t)$, and outputs the prediction of the sensory-motor inputs at a time Δt afterwards in the form $X_{t+1} : (s_{t+1}, m_{t+1})$, as shown in Fig. 2(b). The RNN can also conduct look-ahead prediction for multiple steps without receiving the inputs but feeding back its prediction outputs in the previous step(Tani, 1996). This enables the robot to perform simulations by the inner world of a Hesslow (2002)'s sense. An essence of the RNN is the so called context loop in which the state of the context output units are fed-back to that in the context input units. By utilizing this mechanism, the prediction of the RNN is performed in a context dependent way which means that the sensory-motor values at each step is anticipated utilizing not just the sensory-motor values at the previous step but also its past history. It is well known that by using the back-propagation through time (BPTT) algorithm (Rumelhart, Hinton, & Williams, 1986) the activation patterns of context units at each step are self-organized such that necessary information received in the past can be retained in the internal state memory(Jordan, 1986; Elman, 1990; Tani & Fukumura, 1995). This memory is dynamical one which is inherent to nonlinear dynamical systems of multiple dimensions. Although these memory effects are retained during certain period depending on the conditions of the tarining, they generally decay as time goes by. The possible significance of this context memory will be discussed in the later section in the context of the retention in Husserl phenomenology.

The total output of the network is obtained from the weighted average of each output with its associated value of gate opening at the time g_t^i for all modules. The gate opening is computed dynamically with time using the prediction errors of each module, which are obtained from the difference between the prediction (s_{t+1}, m_{t+1}) and the outcome (s^*_{t+1}, m^*_{t+1}). The gate opens more if its module produces a relatively lower prediction error than the other modules. By having the winner-take-all dynamics among modules, the module with the lowest error over a suitable time interval becomes the winner. The original work on the mixture of experts (Jacobs & Jordan, 1991) used a gating network which selected the module with the closest correspondence to the target outputs. In our architecture, without using a gating network the module is activated autonomously as the result of dynamical competition between all modules over some time interval, utilizing on-line monitoring of the prediction errors. The winning module changes from one module to another as the profile of the sensory-motor flow changes with time. The learning in each module is accelerated if its gate opens more. By using this selective learning scheme, the expertise is developed intensively at each module.

In the proposed on-line learning scheme, the winner-take-all dynamics among modules and the learning in those modules are proceeded as tightly coupled. Therefore, in the early stage in the robot travel before the expertises are fully developed, the module activations tend to be merely ambiguous.

The higher level network learns the gate opening dynamics of the lower level network. More specifically, each RNN module in the higher level samples the gate opening state of the lower level in the current time step $G_T : (g_T^1, g_T^2 .... g_T^n)$ and makes a prediction for the next time step G_{T+1}, as shown in Fig. 2(c). T denotes the time step in the higher level; the higher level sampling interval ΔT is much larger than that in the lower level. The modules in the higher level compete for gate opening g'^i_T, in the same way as shown for the lower level, and the resultant gate opening can be sent to yet higher levels in a recursive manner. The higher level network observes the lower level activities by means of receiving its gate opening dynamics while the lower level network receives the sensory-motor flow. In this manner, the signal is "bottom-up" as abstracted from one level to the next.

2.4 Algorithm

This subsection describes the mathematical formulae for the proposed scheme of the MRE. Suppose a single level network consists of n RNN modules, where x_t^i, y_{t+1}^i, y_{t+1}^* and g_t^i are the inputs, the outputs, the target outputs for teaching and the gate opening of the i-th module RNN, respectively. x_t and y_{t+1} correspond to the sensory-motor state or the gate opening state depending on the levels of the network.

The "soft-max" activation function is used to represent the i-th gate opening g_t^i given by:

$$g_t^i = \frac{e^{s_t^i}}{\sum_{j=1}^{n} e^{s_t^j}} \tag{1}$$

where s_t^i is the current internal value of the i-th gate opening. The total output of the network is y_{t+1}, given by:

$$y_{t+1} = \sum_{i=1}^{n} g_t^i \cdot y_{t+1}^i \tag{2}$$

We define the following likelihood function L which is maximized for prediction learning: it has been obtained by modifying the original definition of Jacobs and Jordan (1991).

$$ln\, L = ln \sum_{i=1}^{n} g_t^i \cdot e^{\frac{-1}{2\sigma^2}\|y_{t+1}^* - y_{t+1}^i\|^2} \tag{3}$$

σ denotes a scaling parameter.

Both the weight of each RNN and the gate opening are updated simultaneously such that the likelihood function is maximized. This point is essential for the on-line learning scheme. In order to obtain the update rules for these two processes, we consider the partial derivatives of the logarithm of the likelihood function with respect to the internal value s^i and with respect to the output of the i-th RNN y^i given by:

$$\begin{aligned} \frac{\partial ln\, L}{\partial s_t^i} &= g(i \mid x_t, y_{t+1}^*) - g_t^i & (4) \\ \frac{\partial ln\, L}{\partial y_t^i} &= g(i \mid x_t, y_{t+1}^*) \frac{(y_{t+1}^* - y_{t+1}^i)}{\sigma^2} & (5) \end{aligned}$$

where $g(i \mid x_t, y_{t+1}^*)$ is the *a posteriori* probability that the i-th module RNN generated the target vector y_{t+1}^*, in terms of x_t. Explicitly, this is given by

$$g(i \mid x_t, y_{t+1}^*) = \frac{g_t^i \cdot e^{\frac{-1}{2\sigma^2}\|y_{t+1}^* - y_{t+1}^i\|^2}}{\sum_{j=1}^n g_t^j \cdot e^{\frac{-1}{2\sigma^2}\|y_{t+1}^* - y_{t+1}^j\|^2}} \tag{6}$$

where $\|y_{t+1}^* - y_{t+1}^j\|^2$ represents the square of the error of the current prediction. Eq. (4) denotes the direction of update for the internal gate opening value s_t^i. The differentiation of $ln\, L$ with respect to y_{t+1}^i involves the error term $y_{t+1}^* - y_{t+1}^i$ weighted by the *a posteriori* probability associated with the i-th module RNN as shown in Eq. (5). Thus the connective weights of the RNN are adjusted to correct the error between the output of the i-th RNN and the global target vector, but only in proportion to the *a posteriori* probabilities. By this means, the individual RNN which is the expert for the on-going input sequence tends to learn exclusively. The error distributed to each module RNN is:

$$error_{t+1}^i = g(i \mid x_t, y_{t+1}^*) \cdot (y_{t+1}^* - y_{t+1}^i) \tag{7}$$

The details of the derivation of Eq. (4) to Eq. (7) are given in Ref. (Jacobs & Jordan, 1991).

Upon obtaining the mathematical formulae, the actual update of the gate opening and the connective weights for each RNN are computed through the use of the back-propagation through time (BPTT) algorithm (Rumelhart et al., 1986). In this computation, the sequence of the sensory-motor inputs as well as the gate internal states for the previous l steps are stored temporally in the window of short term memory. When new sensory-motor inputs are received, their prediction errors are back-propagated in the window memory. Then, the sequence of l steps of the gate internal states in the window as well as the connective weights for each RNN module are updated. When the update is finished, the window memory is shifted one step forward in order to process the sensory-motor inputs of the next time step. The update for s_k^i, which is the ith gate internal state in the kth step in the window memory, is obtained as being:

$$\Delta s_k^i = \epsilon_g \cdot \frac{\partial ln\, L}{\partial s_k^i} - \eta_g \cdot (s_k^i - s_{k-1}^i) \qquad (8)$$

The first term in the right-hand side of the equation represents the direction of the update obtained in Eq. (4); the second term represents the damping term which suppresses abrupt changes in the gate opening; ϵ_g and η_g are parameters. This update is computed in the forward direction in the window memory from k=1 to k=l.

This dynamic gate opening scheme using the window steps was necessary in order to avoid sudden opening and closing of the gates at every step. Eq. (8) makes the gate opening profile smooth along the window steps. At the same time, this treatment makes the gate opening sluggish. When new sensory-motor values are input and stored in the top of the window memory at $t = tc$, the corresponding gate openings are set as neutral. As the time goes by, the sensory-motor values at $t = tc$ shift back in the window memory and the gate openings of $t = tc$ gradually change by following the dynamics defined in Eq. (8). It takes a certain delay time until the winner-take-all dynamic for the gate opening with processing the currently encountered sensory-motor inputs converges. The same scheme is employed for computing the gate opening in the higher level. The lower level gate opening values of d steps delay in the window memory is sampled in every ΔT and they are fed into the higher level sequentially.

The error obtained from Eq. (7) is back-propagated (Rumelhart et al., 1986) through the window memory for each RNN; the update of the connective weights is obtained by means of the steepest descent method utilizing parameters for the learning rate ϵ and for the momentum α.

Before concluding the modeling section, it is noted that recently some other neural network scheme have been proposed for the purpose of segmenting sensory-motor flow into chunks. Ziemke and Thieme (2003) proposed so-called the extended sequential cascade network in which the network connectivity is adapted by using the genetic algorithm, Tani (2003) proposed so-called the RNN with parametric bias, and Bakker, Linaker, and Schmidhuber (2002) applies the scheme of the short-term and long-term memory architecture (Hichreiter & Schmidhuber, 1997) to the sensory-motor learning. These three neural network scheme are common in a sense that multiple behavior primitives are distributedly represented in a single network with a parameter vector. The mapping between the parameter vector and behavior primitives are self-organized through adaptation processes. A specific behavior primitive of learned can be activated by switching the values of the parameter vector. Linaker and Niklasson (2000) showed that a simple vector quantization network based on change detection can perform well on this purpose. The author assumes that those models could also explain some mechanisms of time perception as well as the model proposed by the author. It is also noted that Wolpert and Kawato (1998) independently proposed a modular network for pairs of forward and inverse models as inspired by cerebellum anatomy. Their model, however, did not address the issues of segmentation and level structures.

3 Simulation Experiments

3.1 The environment

The scheme proposed above was examined through the simulation of the mobile robot navigation learning as have been described in the earlier section. For our simulations, we adopted two different rooms, namely Room A and Room B connected by a door, as shown in Figure 1 (a). In this workspace, the robot travels around one room three times, then enters the other room going through the newly opened door and travels around the other room three times. (When the door is opened, the robot autonomously enters into a new room by just conducting the collision-free maneuvering.) The direction of these round travels are the same for all times. The on-line learning experiment was conducted while the robot moved between rooms for a total of 5 room encounters without stoping. The entire travel of the robot in this simulation took about 2100 Δt steps. The robot trajectory repeats the same one at each round travel in the same room since no noise is assumed in the collision-free maneuvering and the characteristics of the controller is constant independent of the neural learning.

The lower level network, which consists of 5 RNN modules each of which has 6 inputs, 6 outputs, 4 hidden units and 2 context units, learns to predict the sensory-motor state in the next step. The higher level network, which consists of 5 RNN modules each of which has 5 inputs, 5 outputs, 4 hidden units and 2 context units, learns to predict the gate opening state in the lower level network in the next step. Other parameter settings for the networks are $\epsilon = 0.002$, $\alpha = 0.9$, $\epsilon_g = 0.007$, $\eta_g = 0.02$, $l = 80$, $d = 20$, $nepochs$=10. These settings are the same for both levels. The sampling interval in the higher level is 10 times longer than that in the lower level ($\Delta T = 10 \cdot \Delta t$). We observed how modules become self-organized in a hierarchical manner by looking at the gate opening dynamics taking place during the prediction learning of the two levels.

3.2 Results

We recorded gate opening dynamics in the window memory both in the lower and in the higher levels during the entire learning process. First, let us consider the gate opening processes in the lower level network. Figure 3 shows the time development of each gate opening state, appeared in the window memory of 20 steps delay, and of the motor input in the lower level for three different periods.

Figure 4 illustrates when and which module wins in the lower level network along the course travelled for each of the three different periods. Fig. 3 (a) shows the profiles for the period from step 130 to step 300 while the robot travelled in Room A for the first time. It can be seen that gate4 and gate3 open in turn as the profile of the motor command changes. In Fig. 4 (a), it is seen that the opening of gate4 corresponds to following a straight wall, while the opening of gate3 corresponds to both a left turn at a corner and to passing a T-junction. Fig. 3 (b) shows the profiles for the period from step 380 to step 550, when the robot experienced Room B for the first time. One can see that gate4, gate2 and gate3 open in turn. Fig. 4 (b) shows that these opening events corresponded to following a straight wall, making a right turn at a corner and making a left turn at a corner, respectively. Fig. 3 (c) shows the profiles for the period from step 820 to step 990, when the robot travelled around Room A for the second time. A remarkable finding is that the gate opening dynamics for this period differ from those observed during the first encounter with Room A. From Fig. 4 (c), one can see that the opening of gate3, which corresponded to both making a left turn at a corner and passing a T-junction in the previous encounter, now corresponds only to making a left turn at a corner, and that the opening of gate1 now corresponds to passing a T-junction. After this period, the learning processes in the network appeared to have stabilized and no further dramatic changes in the correspondence of the gate openings were found. By the end of the simulation, four types of sensory-motor primitives were generated using 4 RNN modules out of the 5 modules available to the lower level network. It is considered that those 4 primitives of left or right turn at at corner, passing a T-junction and following a straight wall are generated because they appear repeatedly in a stable way in their sensory-motor flow during the learning travel. An important observation is that the process of generating primitives is totally dynamic in the sense that the correspondence between the RNN modules and their associated behavior is not static during the on-line learning process. Next, we describe the gate opening dynamics in the higher level network. Figure 5 shows the opening of the 5 gates, appeared in the window memory of 20 steps delay, for the whole period of on-line learning. (The step number in

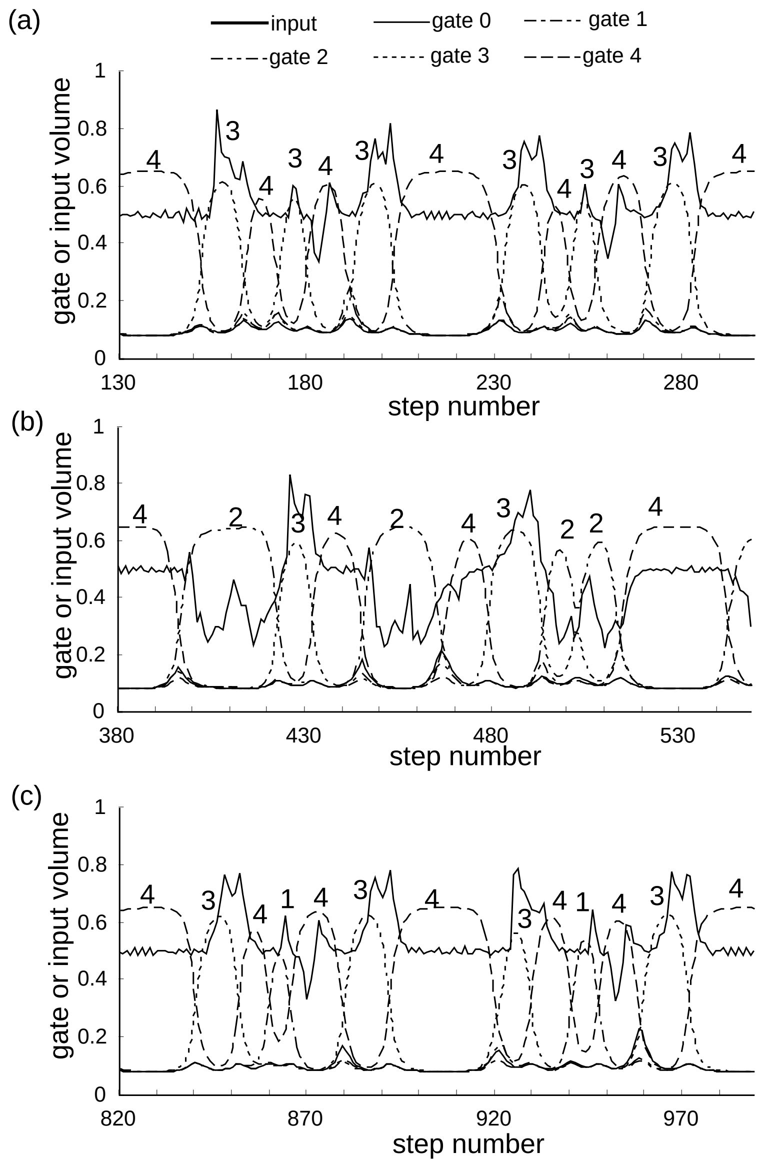

Figure 3: Time development of the opening of 5 gates and of a motor input in the lower level network for three different periods. The number near the data denotes the current winning gate. Redrawing with permission, from (Tani & Nolfi, 1999) Elsevier Science.

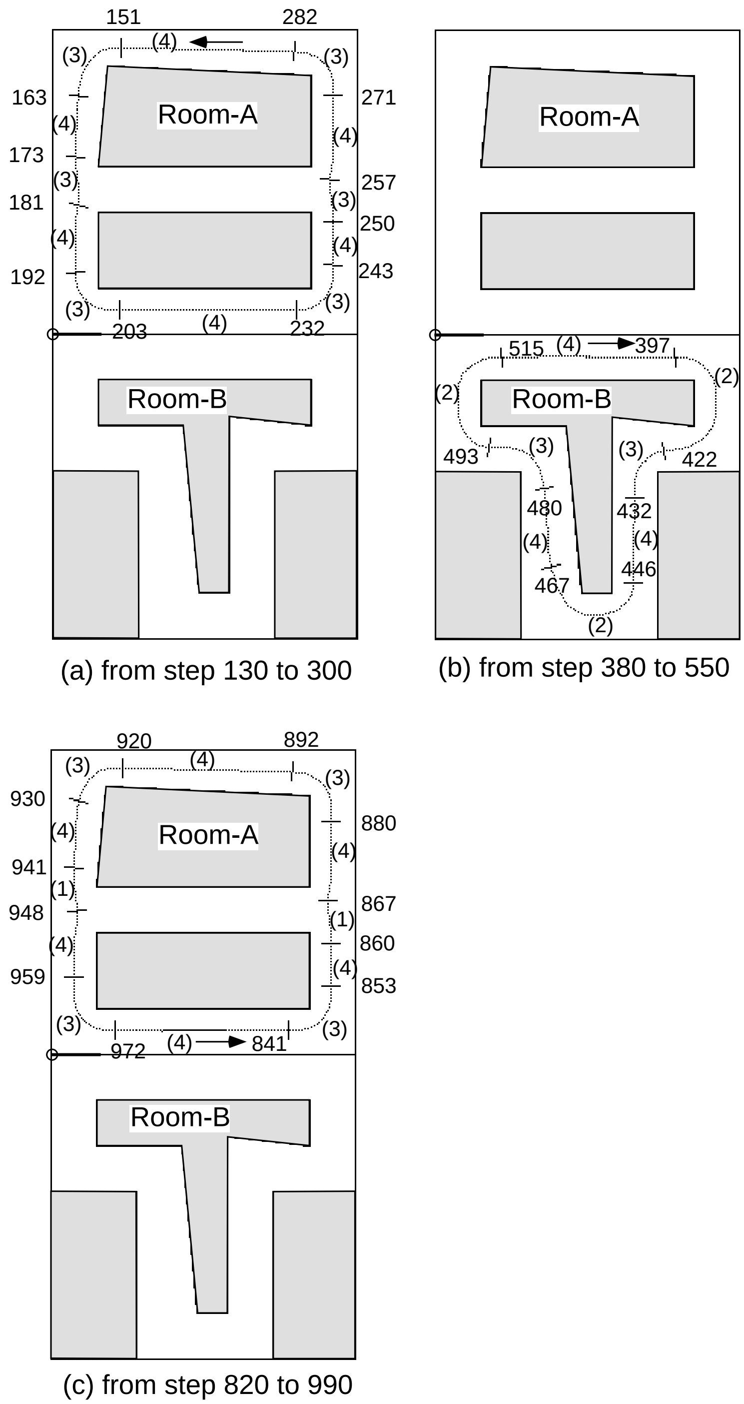

Figure 4: Along the trajectory of the robot travel indicated by the dotted lines, the number in parentheses indicates which module wins and the number without parentheses denotes the step number when the module switching took place. Redrawing with permission, from (Tani & Nolfi, 1999) copy right (1999) Elsevier Science.

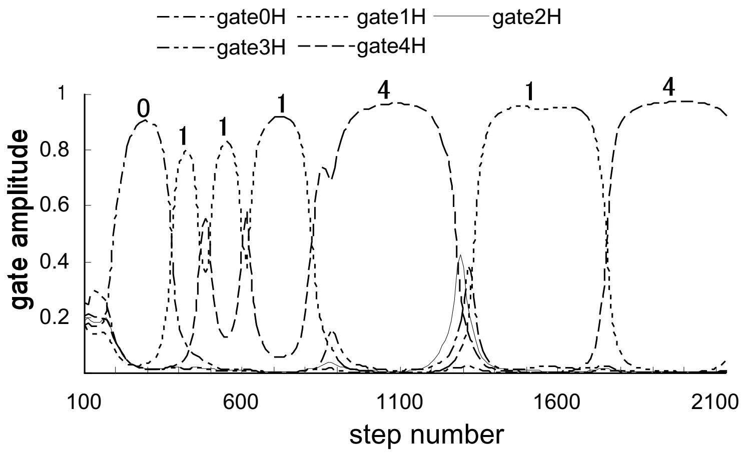

Figure 5: Gate opening dynamics in the higher level network during the whole process of learning. Redrawing with permission, from (Tani & Nolfi, 1999) copy right (1999) Elsevier Science.

this graph denotes the sensory-motor step number in the lower level, for clarity.) One can see that the stable switching of the gate opening between gate4H and gate1H takes place after 800 steps. This switching actually corresponds to the movement between rooms during the travel, where the open state of gate4H and gate1H correspond to travel in Room A and in Room B, respectively. We observe that gate0H opened only in the beginning while the robot traveled in Room A for the first time. The dynamic replacement of module0H by module4H for the representation of Room A evidently took place because the module representation in the lower level network also changed, as noted above. It is readily understood that the dynamics in higher level network can be stabilized only after stabilization occurs in the lower level network.

From so far we have obtained results in the simulation experiments, it can be concluded that the proposed MRE architecture was successful in learning about the environment in a hierarchical way through the sensory-motor interactions of the robot. The lower level network learned to anticipate the row profile of the sensory-motor flow by embedding the sensory-motor flow of specific behavior such as going straight, left or right turning at corner and passing through T-junction into corresponding modules. The higher level network did likewise for the sequences of segmented behavior by generating the forwad models for different rooms.

This learning experiment was repeated for five times with setting of different initial conditions including the starting position of the robot in either Room A or Room B and with different random initial connective weights of the networks. By looking at structures self-organized in the higher level network in these five experiments, equivalent module structures to those in the previous results, representing Room A and Room B, were found in four cases out of the five. Following this, we observed the lower level structures for these four cases and found that equivalent module structures to the previous result appeared in three cases, while the structures were different in one case. In the case where we did not observe clear module structure corresponding to two separate rooms in the higher level, it was observed that the lower level structures continued to change gradually which prevented the higher level structures from stabilizing. The stability in the higher level depends substantially on that in the lower level. These results reveal that the self-organization processes do not always arrive at one optimal solution. They can generate unstable and non-optimal structures by chance.

4 Dynamical systems analysis

In order to understand how the switching of modules takes place dynamically as corresponding to room entering, we examined the time series of prediction outputs by each RNN module in the higher level network. Figure 6 shows the time series of prediction outputs and the corresponding error for each RNN module in the higher level network recorded from step 1400 to step 2000 during which period the robot moved from Room B to Room A. In Fig. 6 (a), the upper five rows represent the sequence of prediction outputs by the five RNN modules. The five squares aligned vertically in each row represent the values predicted for the five coming inputs (the sampling of the five gate openings in the lower level network) by their size. The largest square area corresponds to the value of 1.0 and zero area corresponds to a value of 0.0. The bottom row represents the sequence of five inputs. The robot moved from Room B to Room A at around step 1770 (denoted by a dashed line in the figure). Fig. 6 (b) shows the time series of the prediction error for each module. By looking at Fig. 6 (b), it is observed that the prediction error by RNN 1 remains the lowest among the ones by other RNNs until step 1770. During this period, it is seen, from Fig. 6 (a), that the output sequence by RNN 1 keeps coherence with the input sequence to a certain extent; RNN 0, RNN 2 and RNN 3 are not activated at all, while RNN 4 is activated but is incoherent with the input sequence. On the other hand, the output sequence by RNN 4 becomes coherent with the input sequence after step 1770 with showing the lowest error among others while the output sequence of RNN 1 loses coherence with the input. This switching of the winning RNN modules takes place rather quickly within several iteration steps of the RNNs.

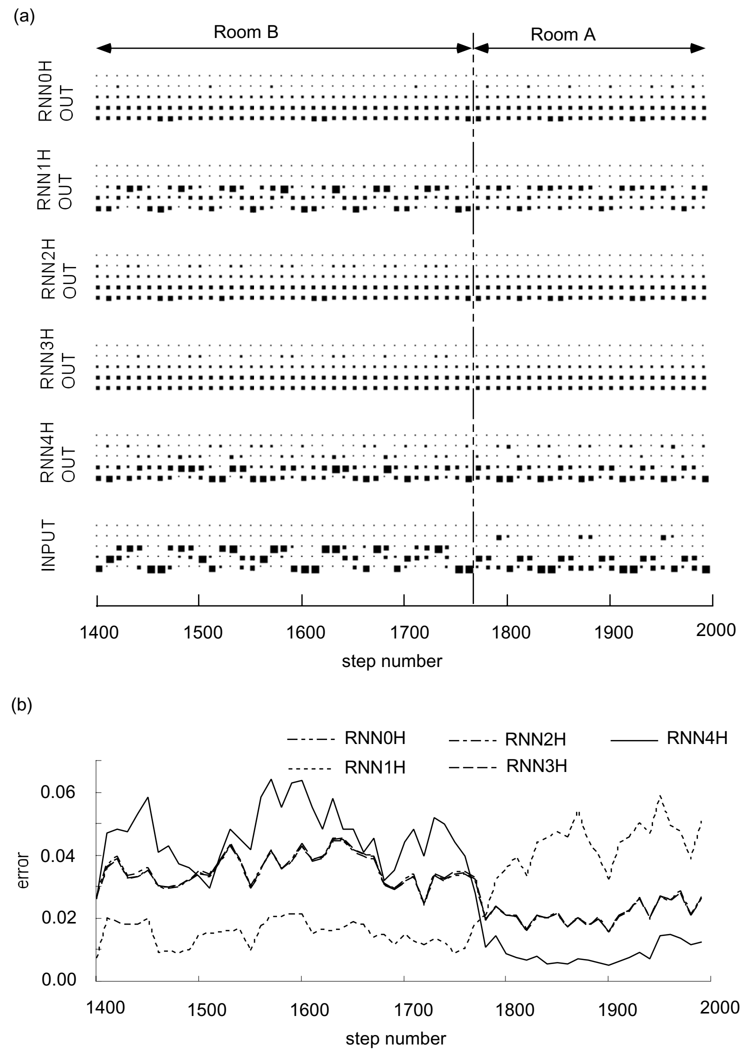

Figure 6: (a) shows the sequence of inputs (shown in the bottom row) and the corresponding prediction outputs by each RNN module (shown in the upper five rows) in the higher level network. (b) indicates the time dependence of the prediction error by each RNN module (the plots of RNN 0, 2 and 3 happen to be superimposed). Redrawing with permission, from (Tani & Nolfi, 1999) copy right (1999) Elsevier Science.

Here, we need to contemplate further why the memories for traveling around Room A and Room B are organized as separated chunks rather than a merged one. This is because a single RNN in the higher level cannot learn to predict the room transition events by the door opening while it can predict perfectly the input sequences of traveling around in a single room. It is difficult for the RNN to learn to predict this transition event since it takes place only once in three times traveling around the rooms. Therefore, each RNN tends to learn an event sequence encountering during its round travel in a specific room as a chunk of a periodic pattern by ignoring rather infrequent events of the room transitions. It is highly assumed that a single RNN can learn a whole sequence as a single chunk if the door opens always since the transition events become deterministic ones. The same can be applied to the way of memory organization in the lower level. If the behavior of left turn at corner always follows that of going straight corridor, those two are memorized as one chunk. Those two in our experimental setting is separated because whether left turn at corner or right turn at corner follows after going straight corridor cannot be predicted. (Since both combinations are possible depending on situations in the same environment.) The message here is that unpredictability which arises from compositional nature in the event sequences initiates segmentation in the sequence that is physically realized by incoherence between the internal dynamics and the external inputs. Since the whole robot travel is organized in a compositional way in the current simulation, those compositional units for reconstructing the input sequences during the travel is generated as chunks both in the lower and the higher level. It is noted that each chunk could be segmented into further fractions if each sequence is too long or more complex to learn by a single RNN. However, generations of such fractions can be avoided by adequately adjusting network size of the RNN. Theoretically speaking, if the learning targets are truly compositional and size of the RNNs are allocated enough, the memory structures should be self-organized solely based on the underlying compositionality.

5 Correspondences to phenomenological time

Finally, we attempt to make correspondences between our dynamical systems account and Husserl's phenomenological ideas of immanent time.

Husserl (1964) introduced the famous idea of "retention" and "protention" for explaining this paradoxical nature of "nowness". He used an example of hearing a sound phrase such as "Do Mi So" for explaining the idea. When we hear the note "Mi", we would still perceive a lingering impression of "Do", and at the same time we would anticipate hearing the next note of "So". The former is called retention and the latter protention. These terms are used to designate the experienced sense of the immediate past and the immediate future. They are a part of automatic processes and cannot be controlled consciously. Husserl believed that the subjective experience of "nowness" is extended to include fringes both in the experienced sense of the past and the future in terms of retention and protension. This description of retention and protention in the so-called pre-empirical level by Husserl seems to directly correspond to what each RNN is performing. The prediction of the RNN is performed by retaining the past flow in a context dependent way as have been described in the previous section. This self-organized contextual flow of the RNN's forward dynamics could account for phenomena of retention.

After coming to understand Husserl's idea of "nowness" in terms of retention and protention, the following question arises. Where is the "nowness" bounded? Husserl believes that the immediate past does not belong to a *representational* conscious memory, but just to an impression. Yet how could the immediate past, experienced just as an impression, slip into a distant past which can be retrieved through a conscious memory operation? What kind of mechanism qualitatively changes an experience from just an impression to an episodic conscious retrieval event? Furthermore, Husserl's goal was to explain the emergence of objective time from the pre-empirical level of retention and protention (Husserl, 1964). Husserl seems to feel that the sense of objective time would emerge as a natural consequence of organizing each experience into one, consistent linear sequence. But, what is the underline mechanisms for this?

The idea of articulation could be a key to answering these questions. Our main idea is that the "nowness" can be bounded where the flow of experience is segmented. The sequential notes of "Do Mi So" constitute a chunk within which a perfect coherence is organized in the coupling between the neural dynamics and the sound stimulus flow. Within the chunk, everything proceeds smoothly, automatically, and unconsciously. However, when we hear a next phrase of "Re Fa La" after "Do Mi So", a temporal incoherence emerges in the transition between the two phrases since this second phrase is not necessarily predictable from the first one. (Here, it is assumed that "Re Fa La" and "Do Mi So" are frequently heard phrases.)

In our neural network model, the winner module is switched from one to another when the external sensory-motor flow cannot be matched with the internal flow of the anticipation. It is noted that this matching is actually conducted in the window of short-term memory, as have been described previously. When the coherence is broken between the sensory-motor flow and that of anticipated in the window memory, the flow is segmented into chunks. Those segmented chunks are no more just parts of the flow, but the events that are identified, by an activated module, as one of the sensory-motor categories. This identification process takes a certain time period because of delays in the convergence of the winner-take-all dynamics among the modules, as have been explained in the previous section. This might explain the phenomenological observation that the flow of the immediate past is experienced just as an impression, which later becomes a consciously retrieval object after segmented.

Then after, the higher level RNN learns the sequences of the identified events and becomes able to regenerate them. In the memory retrieval, however, the sensory-motor flow can be reconstructed only in an abstract way since the flow is now represented by combining a set of behavior units. Although such ways of reconstructions could provide compositionality as well as generalization in representing the sensory-motor flow while they might lose subtle differences or uniqueness in each instance of experience. Consequently, it is presumed that the sense of objective time may appear when the experience of the sensory-motor flow is reconstructed in a compositional form while loosing its peculiarity.

An interesting suggestion drawn from our proposed model is that the time perception might deal with hierarchical structures. Indeed, this can be explained phenomenologically. For example, let us assume that we repeatedly hear "Re Fa La" followed by "Do Mi So" as a sequence. In such a case, we can imagine generating a new chunk in the higher level which ties these two phrases into a familiar sequence. Therefore, when we hear the phrase of "Re Fa La", we would have retention of "Do Mi So". In this situation, a question is whether the nowness is bounded inside of "Do Mi So", or if it is extended to the newly tied chunk of "Do Mi So" and "Re Fa La". Let us suppose a situation in which we hear a phrase "Ti Re So" instead of "Re Fa La" after "Do Mi So". We would then feel a sense of incompatibility in this new phrase which was not anticipated after "Do Mi So", and we would say that "now" I hear a strange phrase. However, it would be different if we heard a phrase like "Re Re Fa" in which the second note was generated by mistake. The sense of incompatibility comes from the note level in this situation and we would say that "now" I hear a strange note. The point of interest is that the sense of nowness can be directed to different levels depending on the level at which coherence is broken. And the underlying mechanism of this phenomena can be explained by the proposed model in which the sensory-motor flow input to the system is reconstructed by employing the multiple levels of representations. Goguen (2004) considers a similar ideas in his studies of musical qualia. He proposed that the structure of consciousness in musical experiences are hierarchically organized by their saliency, with emotional tone determined by their resonance with protention. He also considered that the anticipation plays an essential roles where its errors generates segments that are actually musical qualia. He suggested to use dynamic neural network models like our models for musical protention. Poppel (1997) proposed a hierarchical model of temporal perception in which system states of 30 ms and integration intervals of 3 s, together with a memory store, provide an explanatory neuro-cognitive machinery for differential subjective duration. The possible relations between his model and ours should be investigated in the future.

Husserl considered a further deeper level that is named the absolute flow level (Husserl, 1964). In this level neither retention nor protention yet appears. Only flow exists there. The flow is continuous and fluent but it can be stagnant sometimes. Husserl seems to believe that this stagnancy initiates certain retentional dynamics that leads to conscious perception of time in the end.

The author has been considering that this absolute flow level might be related to so-called the pure dynamics level in the author's cognitive robotics studies. When we conduct robotics experiments of either simulations or physical ones, we observe the system dynamics in terms of sensory-motor flows or neural activations as on-line. By this on-line observation, various impressions are obtained as like, now the system proceeds smoothly, it falls into stagnant, or it changes dramatically in all sudden. This level of observations are primitive but the purest since they have not been articulated yet using any a priori assumptions or knowledge. The observations in this pure dynamics level is crucial since they often provide important intuitions that could correspond to experiences in the absolute flow level of phenomenology. In the next level, analysis are conducted for the state trajectories recorded during the experiments by using dynamical systems language such as convergence, divergence, attractors, coherence, incoherence, etc. The impressions obtained in the previous level is now described rationally with using the dynamical system terms. Varela (1999) introduced the idea of intermittent chaos in order to account for the absolute flow that can shift from fluent to stagnant intermittently. Tani (1998) also discussed open dynamics in a similar fashion. Finally, the obtained dynamical systems descriptions should be further examined in the cognitive level. If particular dynamical systems phenomena appear, they should be examined in terms of various cognitive constraints that act on the dynamical systems as boundary conditions or initial conditions. By going through these levels of examinations, the intuitive impressions obtained in robotics experiments might be shaped up to concrete theories accounting for both of phenomenology and cognition.

In summary, the current paper related the robotics experiments of the sensory-motor articulations in multiple levels to the phenomenological discussions of time perception by Husserl. It was shown that the sensory-motor flow is segmented into sequences of identified events through the dynamic competition process among the expert modules. Those sequences are further segmented into larger time-scale chunks in the higher level. These experiments demonstrated the developments from the pure dynamics level where frequent transitions between fluency and stagnancy of the system flow was observed to the cognitive level where the flow is reconstructed in a compositional structure. The author considers that this development might account for the phenomenological question of how objective time emerges out of subjective time if the human brain is organized along functional principles similar to those discussed in the current paper.

Acknowledgment
I thank anonymous reviewers for their useful suggestions in improving the paper. I also thank Dr. Stefano Nolfi, Dr. Takashi Ikegami, Dr. Joseph Goguen and late Dr. Francisco Varela for fruitful discussions.

References

Arbib, M. (1981). Perceptual structures and distributed motor control. In *Handbook of Physiology: The Nervous System, II. Motor Control* (pp. 1448–1480). Cambridge, MA: MIT Press.

Bakker, B., Linaker, F., & Schmidhuber, J. (2002). Reinforcement Learning in Partially Observable Robot Domains Using Unsupervised Event Extraction. In *Proc. International Conference on Intelligent Robots and Systems 2002.*

Beer, R. (1995). A dynamical systems perspective on agent-environment interaction. *Artificial Intelligence*, *72*(1), 173–215.

Elman, J. (1990). Finding structure in time. *Cognitive Science*, *14*, 179–211.

Feldman, A. (1980). Superposition of motor programs, I. Rhythmic forearm movements in man. *Neuroscience*, *5*, 81–90.

Gelder, T. van. (1998). The dynamical hypothesis in cognitive science. *Behavior and Brain Sciences.* (in press)

Goguen, J. A. (2004). *Musical qualia, context, time and emotion.* (submitted to Journal of Consciousness Studies, special issue on art, brain and consciousness.)

Hesslow, G. (2002). Conscious thought as simulation of behaviour and perception. *Trends in Cog. Sci.*, *6*(6), 242-247.

Hichreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9(8)*, 1735–1780.

Husserl, E. (1964). *The phenomenology of internal time consciousness, trans. J.S. Churchill.* Bloomington, IN: Indiana University Press.

Jacobs, R., & Jordan, M. (1991). Adaptive mixtures of local experts. *Neural Computation*, *3*(1), 79–87.

James, W. (1890). *The principles of psychology.* Dover Publ. (reprinted 1950).

Jordan, M. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. In *Proc. of eighth annual conference of cognitive science society* (pp. 531–546). Hillsdale, NJ: Erlbaum.

Jordan, M., & Rumelhart, D. (1992). Forward models: supervised learning with a distal teacher. *Cognitive Science*, *16*, 307–354.

Kelso, S. S. . S. (1988). Dynamic Pattern Generation in Behavioral and Neural Systems. *Science*, *239*, 1513-1519.

Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. *The Int. J. of Robotics Research*, *5*(1), 90–98.

Kuniyoshi, Y., Inaba, M., & Inoue, H. (1994). Learning by Watching: Extracting Reusable Task Knowledge from Visual Observation of Human Performance. *IEEE. Trans. on Robotics and Automation*, *10*(6), 799–822.

Linaker, F., & Niklasson, L. (2000). Time series segmentation using an adaptive resource allocating vector quantization network based on change detection. In *Proc. International Joint Conference on Neural Networks 2000.*

Poppel, E. (1997). A hierarchical model of temporal perception. *Trends in Cognitive Science*, *1*(2), 56–61.

Rumelhart, D., Hinton, G., & Williams, R. (1986). Learning internal representations by error propagation. In D. Rumelhart & J. Mclelland (Eds.), *Parallel distributed processing* (pp. 318–362). Cambridge, MA: MIT Press.

Tani, J. (1996). Model-Based Learning for Mobile Robot Navigation from the Dynamical Systems Perspective. *IEEE Trans. on SMC (B)*, *26*(3), 421–436.

Tani, J. (1998). An interpretation of the "self" from the dynamical systems perspective: a constructivist approach. *Journal of Consciousness Studies*, *5*(5-6), 516–42.

Tani, J. (2003). Learning to generate articulated behavior through the bottom-up and the top-down interaction process. *Neural Networks*, *16*, 11–23.

Tani, J., & Fukumura, N. (1995). Embedding a Grammatical Description in Deterministic Chaos: an Experiment in Recurrent Neural Learning. *Biological Cybernetics*, *72*, 365–370.

Tani, J., & Nolfi, S. (1998). Learning to perceive the world as articulated: an approach for hierarchical learning in sensory-motor systems. In R. Pfeifer, B. Blumberg, J. Meyer, & S. Wilson (Eds.), *From animals to animats 5.* Cambridge, MA: MIT Press. (later published in Neural Networks, vol12, pp1131–1141, 1999)

Tani, J., & Nolfi, S. (1999). Learning to perceive the world as articulated: an approach for hierarchical learning in sensory-motor systems. *Neural Networks*, *12*, 1131–1141.

Uno, Y., Kawato, M., & Suzuki, R. (1989). Formation and control of optimal trajectory in human multijoint arm movement. *Biological Cybernetics*, *61*, 73–85.

Varela, F. (1999). Present-Time Consciousness. *Journal of Consciousness Studies*, *6*(2-3), 111–140.

Wolpert, D., & Kawato, M. (1998). Multiple paired forward and inverse models for motor control. *Neural Networks*, *11*, 1317–1329.

Ziemke, T., & Thieme, M. (2003). Neuromodulation of Reactive Sensorimotor Mappings as Short-Term Memory Mechanism in Delayed Response Tasks. *Adaptive Behavior*, *10*(3/4), 185-199.