Anticipatory Behavior in Adaptive Learning Systems, Springer-Verlag, pp.167-178, 2003.

Symbols and Dynamics in Embodied Cognition: Revisit a Robot Experiment

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

http://www.bdc.brain.riken.go.jp/ tani

Abstract

This paper introduces novel analyses that clarify why the dynamical systems approach is essential for studies of embodied cognition by revisiting author's prior robot experiment studies. Firstly, we argue that the symbol grounding problems as well as the "situatedness" problems should be the consequences of lacking a shared metric space for the interactions between the higher cognitive levels based on symbol systems and the lower sensory-motor levels based on analog dynamical systems. In our prior studies it was proposed to employ recurrent neural networks (RNNs) as adaptive dynamical systems for implementing the top-down cognitive processes by which it is expected that dense interactions can be made between the cognitive and the sensory-motor levels. Our mobile robot experiments in prior works showed that the acquired internal models embedded in the RNN is naturally situated to the physical environment by means of entrainment between the RNN and the environmental dynamics. In the current study, further analysis was conducted on the dynamical structures obtained in the experiments, which turned out to clarify the essential differences between the conventional symbol systems and its equivalence realized in the adaptive dynamical systems.

1 Introduction

We speculate that the problems of cognition commence with the robots' attempt to acquire internal models of the world in certain forms so that they can mentally simulate or plan their own behavior consequences. By this means, we may not consider purely reactive-type robots that base their actions on simple sensori-motor reflexes since those robots do not deal with any mental processes that employ internal models. When discussing internal models, it is important to consider how they can be grounded to the physical environments and how the mental processes manipulating them can be situated in the behavioral contexts. This question addresses one of the observation problems in cognition which asks us where the observer, dealing with the descriptions, is positioned. We examine these problems by revisiting our prior studies on robot navigation learning experiments [1]. We attempt to conduct further dynamical systems analysis for the results of this experiment. This analysis will clarify the essential differences between symbols in conventional symbol systems and those embedded in analog dynamical systems through learning processes.

In the traditional approach of the robot navigation problems, the robots are forced to acquire exact maps of the environment measured in the global coordinate systems. Such robots apparently use the external views to describe their environments, since the descriptions are made by assuming the global observation from the outside.

On the other hand, the recent approach based on landmark-based navigation [2, 3] does not assume any global observations of the environments. In this approach, the observer sits inside the robot and looks at the outside through the sensory device focusing on upcoming events or landmarks. The observer collects the sequences of landmark-types and tries to build chain representations of them in the form of finite state machines (FSM) as the topological map of the environment. Although it is true that this approach provides us with much more successful results in the navigation tasks compared to the global map strategies, the symbolic representation of the FSM can still cause the symbol grounding problems. The symbol grounding problem is a general problem, as discussed by Harnad [4]. The problem is that discrepancies occur between the objects in the physical environment and their symbolic representations in the system which cannot be resolved autonomously through the system's own operations.

Let us consider the situation where the robot navigates in a pre-learned environment by identifying the current position from trying to match the state transitions in the FSM. A problem can arise when the robot fails to recognize an oncoming landmark because of some noise. The robot will be lost because it has received an erroneous sensory input which is different from the one expected using the FSM. The FSM will simply halt upon receiving this illegal input[1].

Although some may argue that this problem can be resolved by further development of the categorization schemes for landmark recognition, we consider that this approach leaves the underlying problem unsolved. We believe that the underlying problem exists in the position of the observers who look over the symbolic representations and try to manipulate them. The observer here is external to the descriptions. As long as such external observers are allowed for the robots, the robots face the symbol grounding problems.

We have investigated this problem from the dynamical systems perspectives [5, 6]. We speculate that real number systems best represent the mental activities of robots. We expect that the nonlinear dynamics characterized by chaos or fractal may serve as a basis for the mental activities of robots, as the theories of symbolic dynamics [7, 8, 9] have shown that such nonlinear dynamics exhibits a certain linguistic complexity. When the internal dynamics, which describe the mental processes of the robot, and the environment dynamics are coupled together through the sensory-motor loop, those two dynamics would share the same metric space. We consider that the mental processes of the robots can be naturally situated to the environments as the coherence is achieved between those two dynamics interacting with each other in the same phase space. An important objective here is to unify the two separate entities for "descriptions" and their "manipulations" in the systems into one entity within the framework of the time-development of dynamical systems. We speculate that the internal observer [10, 11] finally appears in the cognitive processes of robots if this objective is accomplished. The next section reviews our embodied work on mobile robot learning of cognitive maps based on the dynamical systems approach.

[1]When an illegal input is received, the current state cannot be identified correctly. There could be an extra algorithm by which the current state can be estimated by means of the maximum likelihood. Such an extra algorithm, however, could generate another symbol grounding problem. The author will argue that there are intrinsic mechanisms to avoid these problems in the dynamical systems approach

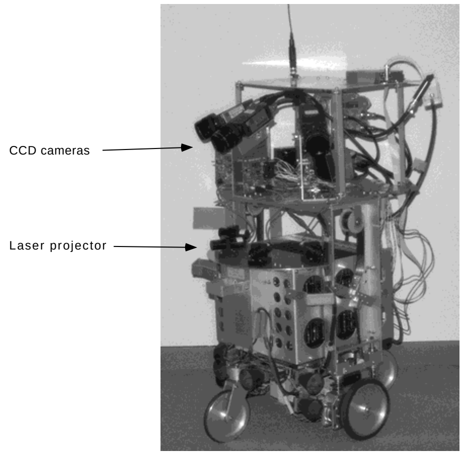
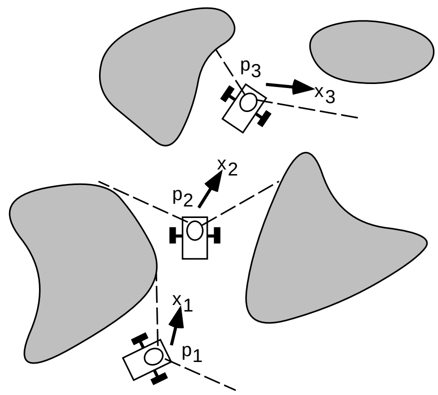


Figure 1: The *YAMABICO* mobile robot. It is equipped with a laser range sensor.

2 Formulation

Firstly we review our navigation scheme which is applied to the *YAMABICO* mobile robot (cf. Fig. 1). *YAMABICO* can obtain the range image by a laser range finder in real-time. In our formulation, maneuvering commands are generated as the output of a composite system consisting of two levels. The control level generates a collision-free, smooth trajectory using a variant of the potential field method i.e. the robot simply proceeds towards a particular potential hill in the range profile (direction toward an open space). The navigation level focuses on the topological changes in the range profile as the robot moves through a given workspace. The profile gradually changes until another local peak appears when the robot reaches a branching point. At this moment of branching the navigation level decides whether to transfer the focus to the new local peak or to stick with the current one. The navigation level functions only at branching points which appear in unconstructed environments. The importance here is that the navigation of the robot consists of the topological trajectories which are determined by the branching sequences. The control level is pre-programmed and the learning takes place only in the navigation level. Hereafter, our discussion focuses on how to learn and determine the branching sequences using neural learning schemes.

In the learning phase, the robot explores a given obstacle environment by randomly determining branching. Suppose that the robot comes to the



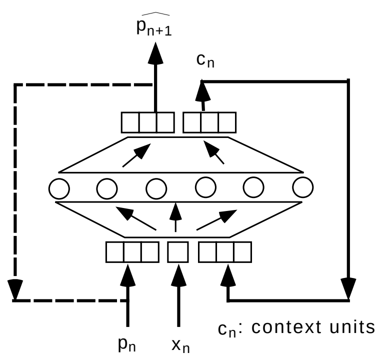


Figure 2: Sensory-motor sequence in branching and RNN architecture.

n*th* branch point with receiving sensory input (range image vector) p_n and randomly determine branching (0 or 1) as x_n, then it moves to the next branch point n+1*th* (see Fig 2.) Through the entire exploratory travel, the robot acquires the sensory-motor sequence of (p_i, x_i). Using this sample of the sensory-motor sequence, a recurrent neural net (RNN) is trained so that it can predict the next sensory input p_{n+1} in terms of the current sensory input p_n and the branching motor command x_n (see Fig 2). We employ the idea of the context re-entry by Jordan [6] which effectively adds internal memory to the network. The current context input c_n (a vector) is a copy of the context output in the previous time: by this means the context units remember the previous internal state. The navigation problem is an example of a so-called "hidden state problem"(or non-Markov problem) where a given sensory input does not always represent a unique situation/position of the robot. Therefore, the current situation/position is identifiable, not by the current sensory input only, but additionally the memory of the sensory-motor sequence stored during travel is necessary. The memory structure is self-organized through the learning process. We expect that the RNN can learn certain "grammatical" structure hidden in the obstacle environment as embedded in its intrinsic dynamical structure by utilizing the context re-entry. (As many have shown the capability of RNNs for grammar learning.) We employ the back-propagation through time algorithm [12] for the RNN learning.

Once the RNN is trained, it can conduct the following two types of mental processes. (1) The RNN can conduct lookahead prediction of the sensory sequences for arbitrary given motor programs (branching sequences) by the closed-loop forward computation. In this computation the sensory prediction outputs in the current step is copied to the sensory inputs in the next step as shown by a dashed loop in the left hand side of the RNN in Fig 2. In this way, lookahead prediction of the future sensory sequence can be recursively computed with a given branching sequence. (2) The RNN can conduct goal-directed planning. It can generate the motor programs (branching sequences) for the robot to reach a goal specified by the corresponding distal sensory image. The inverse dynamics of the RNN with the minimum travel distance criteria can determine an optimal motor program. Details of goal-directed planning are not shown here, but in [1].

3 Experiment

Here, we review a part of our experiments of lookahead prediction. The robot explored a given workspace and the RNN was trained with 193 samples of the sensory-motor sequence. After this learning, the robot is started to travel from arbitrary positions. The robot maneuvers following an arbitrary motor program (branching sequence) and it tries to predict the coming sensory input of the next branch using the sensory input and a given motor command at each current branch. (This is one-step lookahead prediction.) Fig 3 shows an example of the results. The upper part of the figure shows the measured trajectory of the robot. The lower part shows the comparison between the actual sensory sequence and the predicted one. The figure shows the nine steps of the branching sequence, where five units in the most left are the

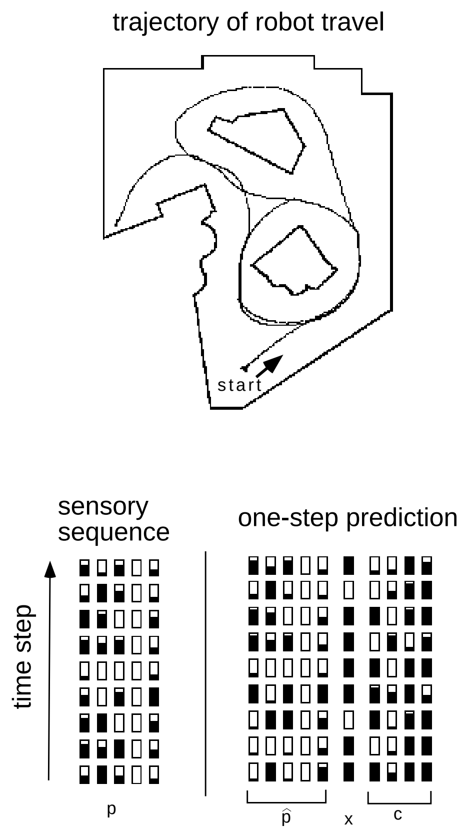


Figure 3: One-step prediction.

sensory input, the next five units are its prediction, the next one unit is the motor command (0 or 1 of branching), and the most right four units are the context units. Initially the robot cannot predict correctly. It, however, becomes able to predict correctly after the 4th step. Since the context units are randomly set initially, the prediction fails at the very beginning. However as the robot continues to travel, the sequence of the sensory input "entrain" the context activations into the normal state transition sequence, thereafter the RNN becomes able to predict correctly. We repeated this experiment with various initial settings (positions and motor programs), which showed that the robot always starts to predict correctly within 10 steps. Furthermore we found that although the context is easily lost when perturbed by large sensory noise (e.g. when the robot fails to detect a branch or receiving totally different values of the sensory inputs from the ones expected for some branching steps), the prediction can be always recovered as long as the robot continues to travel. This auto-recovery of the cognitive process is made in consequence that a sort of coherence is organized between the internal and the environmental dynamics in their interactions.

Once the robot is "situated" in the environment (i.e. the robot becomes able to conduct one-step predictions correctly as the context is recovered after the travel), the robot can conduct multiple steps of lookahead predictions from a branching point. An example of the comparison between a lookahead prediction and its outcome of the actual sensory sequence during the travel is shown in Fig. 4. In (a) the arrow denotes the branching point where the robot conducted a lookahead prediction using a motor program given by 1100111. The robot, after conducting the lookahead prediction, traveled following the motor program, generating an "eight-figure" trajectory, as shown. In (b) the left-hand side shows the sensory input sequence, while the right-hand side shows the lookahead sequence, the motor program and the context sequence. This sequence consists of eight branching steps (from the 0th to the 7th step) including the initial one in the "start" point. It can be seen that the lookahead for the sensory input agrees very well with the actual values. It is also observed that the context as well as the prediction of sensory input at the 0th and the 7th steps are almost the same. This indicates that the robot predicted its return to the initial position at the 7th step in its "mental" simulation. The robot actually returned back to the "start" point at the 7th step in its test travel. We repeated this experiments of lookahead prediction for various branching sequences, and found that the robot can predict the sensory sequences correctly for arbitrary motor programs unless severe noise

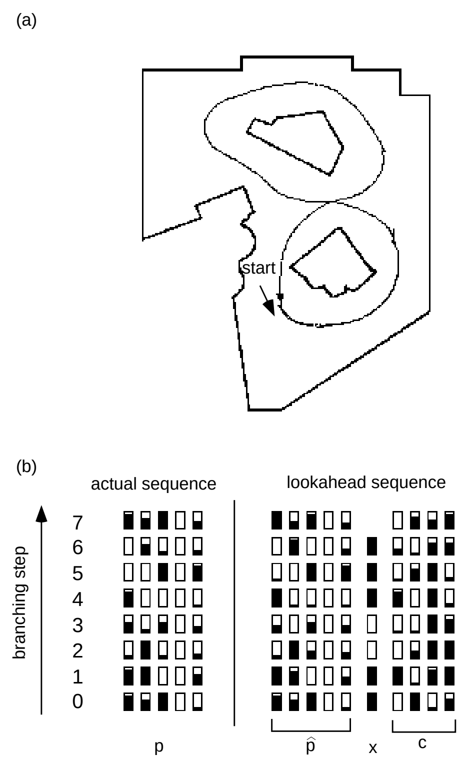


Figure 4: The robot conducted lookahead prediction for a motor program from a branching point.

affects the branching sequence. From this result, it can be assumed that the robot successfully learned to extract grammatical structure hidden in the obstacle workspace.

4 Analysis

It is assumed that there exists an essential dynamical structure which can generate the coherence between the internal and the environment system, as we have discussed. We conducted the phase space analysis of the obtained RNN in order to see such structure. Phase plots show shapes and structures of attractors (invariant sets) of dynamical systems. For the purpose of drawing the phase plot of the RNN trained, the re-entry loop is connected from the sensory output nodes to the sensory input nodes so that the RNN can conduct lookahead predictions for arbitrary length of motor command sequences. Then the RNN was activated for 2000 steps with feeding randomly generated branching sequences of $x*$. (Here, the RNN conducts mental simulations for the random branching sequences.) The state space trajectory of the context units was plotted using the activation sequences of two context units (we took a 2-D projection of the entire state space) excluding the first 100 step points, which are regarded as transient. The result is a one-dimensional like invariant set as shown in Fig. 5 (a). Our mathematical analysis shows that this invariant set is topologically transitive[2].

We also found that the invariant set is a global attractor since the plotting always converges into the same figure independent of the initial setting of the context values or the branching sequences used. More intuitively, the state starting from arbitrary points in the state space goes around the phase widely during the initial transient period. After convergence, the state transits only within the invariant set (among segments) shown in Fig. 5.

In this plot, one may interpret that the invariant set shows the boundary of rationality/cognition for the mental processes of the robot. When the RNN is perturbed by receiving noisy inputs, the state goes out of the invariant set where the rationality in terms of predictability is lost. However, as the RNN continues its dynamical iterations, the state always returns to the rational region, i.e. the invariant set, and the RNN is able to predict correctly again. This cognitive boundary is self-maintained solely from the

[2]This means that there are always finite step transition paths between any given two points in the invariant set

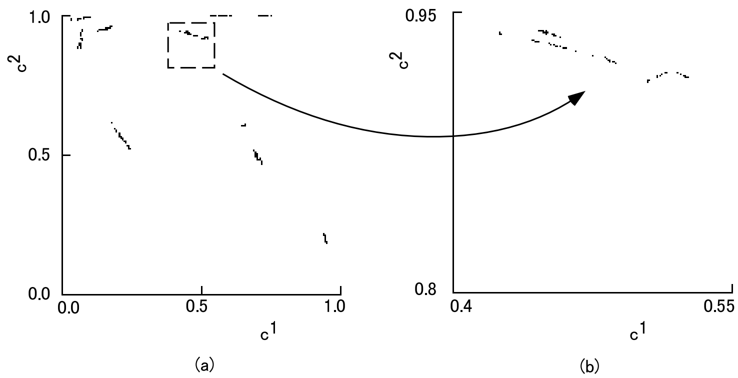


Figure 5: (a)The global attractor appeared in the RNN dynamics and (b) enlargement of a part of the attractor.

system's own dynamical iterations, as stated by Maturana and Varela [13]. Here, we see that an inherently robust mechanism of dynamic cognition is achieved by self-organizing the global attractor.

Our further analysis of this invariant set revealed the fact that each line segment corresponds to each identical branching point. Each segment has two ways of transitions depending on the binary branching. And each segment is accessible from all other segments within finite steps of state transitions (Remember that the invariant set is topologically transitive.). It was also found that each segment is not just a one-dimensional collection of points but it is organized as a Cantor set [14] where the history of past branching sequences is represented by the current position in this Cantor set. Mathematically speaking, if two branching sequences share an infinite number of steps of same branching sequences in the past, the current states of the two sequences will correspond to two points that are epsilon-neighbors on the Cantor set of the same segment. On the other hand, two points will be distanced from each other if their recent past sequences are different. (This is due to the dynamic characteristics of the RNN as an iterated function system. See [15] for the details.) An interesting point is that the RNN naturally takes a context-dependent representation in which the history of the robot travel is encoded tacitly. This idea is also related to Tsuda's [16] Cantor set coding of the episodic memory in the hippocampus.

What we see in the phase plot is the so-called dynamical closure which some might interpret as equivalent to an FSM representation. However, a segment shown in the phase plot is not equivalent to a node in an FSM since it maintains more rich information of the context in terms of the Cantor set coding. The "symbols" appeared in the dynamical systems scheme are not arbitrarily shaped tokens in Harnad's terminology [4], but they maintain a certain metric structure in a tacit manner.

5 Summary and Discussion

It can be said that the representation and manipulation capabilities of symbols have been the most significant power of Artificial Intelligence. However, cognitive robotics researchers found that such computational symbols cannot be grounded easily in the physical environment. Then, they attempted to employ pattern categorizers as ideal interfaces between the real world analog dynamical systems and the computational symbolic systems. However, such trials could not produce much successful solutions to the problems. The failure is due to the fact that those two systems cannot share the same metric space where they can interact densely with each other.

Our studies have shown an alternative approach based on the dynamical systems view. We proposed that a RNN, as an adaptive dynamical system, could be an alternative to symbol systems which can naturally interact with the physical real world by sharing the same metric space of analog dynamical systems. Our experiments with a real robot have shown some interesting results. Firstly, it was shown that the internal system, once perturbed by possible accidental events or noise, is naturally re-situated to the environment system by means of entrainment between the two systems. This sort of entrainment between the internal and environmental systems becomes possible because those two systems share the same metric space of analog dynamical systems in our scheme.

Secondly, the phase space analysis indicated that the dynamical system's iterations by the RNN can be equivalent to the symbol manipulations processes of FSMs or language as has also been indicated by Pollack [8], Kolen [15] and Elman [17] [3]. We, however, found genuine differences between the symbol systems embedded in the proposed dynamical system and those of the computational FSMs in the way they are constituted. In the FSM formulation, first nodes are allocated as discrete states and then transitions

[3]Their studies, however, could not articulate enough the advantage of RNNs as an alternative of symbol systems since their studies never addressed the embodied cognition of sensory-motor systems.

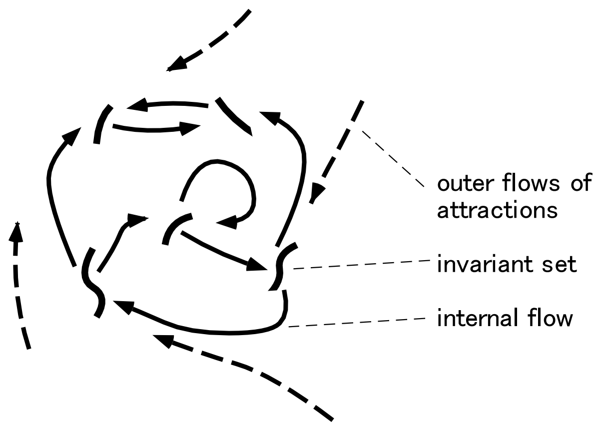


Figure 6: The invariant set, the outer flow attracted to the invariant set, and the internal flow going inside the invariant set are illustrated.

among them are defined. In our dynamical systems scheme using a RNN, the dynamical flow, which is represented as a local vector in the state space, is organized through the sensory-motor learning processes. As a result, a mechanism equivalent to the FSM becomes visible in the phase space. The dynamical flow includes both the outer flow attracted towards the invariant set and the internal flow going only within the invariant set (see Fig. 6 for the illustration).

A crucial point is that an equivalent function of the FSM can be generated only in the form of an attractor and that the outer flow of attraction is indispensable for its existence. It is this flow that explains the auto-recovery mechanism of the system from its perturbed states. Important here is that the auto-recovery mechanism is intrinsic to the dynamical systems scheme since the internal flow and the outer attraction flow are generated as inseparable units in the process of self-organizing the attractor. This suggests that the transient dynamics might be more crucial in cognitive systems than believed previously since once conflicts or gaps arise between the mental images and its reality they are resolved during such transient periods. Symbol systems, which support neither notions of attraction nor transient dynamics, just halt if conflicts occur unless extra exception-handling type mechanisms are initiated.

In the proposed dynamical systems scheme, the system itself neither sees the descriptions of an FSM nor involves their direct manipulations. When the internal system merely repeats its dynamical iterations, the emergent structure of the state transitions observed in the phase plots by an outside observer (like myself) may be perceived as if symbols actually existed and were manipulated internally. In fact, all what exists is the dynamical structure and the resultant dynamical flow in the system. The descriptions and manipulations appear to be an inseparable entity in the dynamical system. Since there are no observers dealing with the descriptions, we finally find the internal observer [10, 11] in our robot. Consequently, there are no descriptions or symbols to cause the symbol grounding problem from the view of this internal observer.

In the end, I would like to address the open problems related to this study. When the original experiments reviewed in this paper were completed about 8 years ago, I thought of two future directions. One direction was to study dynamic adaptation schemes in which the robot has to learn about open environments in an incremental way rather than off-line. The studies have been conducted [18, 19] and are continuing focusing on how coherent and incoherent phases autonomously appear during the dynamic changes of the internal attractors. We proposed that such open-dynamics characteristics might explain the momentary self-consciousness discussed in the phenomenology literatures. The other line of study is to consider articulation mechanisms of sensory-motor flows. The branching mechanism in the YAMABICO experiments was pre-programmed as described earlier. Then, we started to consider how "concepts" of branching or landmarks can be learned as articulated from the experiences of continuous sensory-motor flow in navigation tasks [20]. This question leads to further general questions of how behavior primitives could be self-organized[21] and how they can be combined to generate diverse behavior patterns[22]. However, I have to admit that these studies are still half-baked and there are many open problems left for future studies.

References

[1] J. Tani, "Model-Based Learning for Mobile Robot Navigation from the Dynamical Systems Perspective", *IEEE Trans. on SMC (B)*, vol. 26, no. 3, pp. 421–436, 1996.

[2] B. Kuipers, "A qualitative approach to robot exploration and map

learning", in *AAAI Workshop Spatial Reasoning and Multi-Sensor Fusion (Chicago)*, 1987, pp. 774–779.

[3] M. Mataric, "Integration of representation into goal-driven behavior-based robot", *IEEE Trans. Robotics and Automation*, vol. 8, no. 3, pp. 304–312, 1992.

[4] S. Harnad, "The symbol grounding problem", *Physica D*, vol. 42, pp. 335–346, 1990.

[5] R.D. Beer, "A dynamical systems perspective on agent-environment interaction", *Artificial Intelligence*, vol. 72, no. 1, pp. 173–215, 1995.

[6] M.I. Jordan and D.E. Rumelhart, "Forward models: supervised learning with a distal teacher", *Cognitive Science*, vol. 16, pp. 307–354, 1992.

[7] J.P. Crutchfield, "Inferring statistical complexity", *Phys Rev Lett*, vol. 63, pp. 105–108, 1989.

[8] J.B. Pollack, "The induction of dynamical recognizers", *Machine Learning*, vol. 7, pp. 227–252, 1991.

[9] J. Tani and N. Fukumura, "Embedding a Grammatical Description in Deterministic Chaos: an Experiment in Recurrent Neural Learning", *Biological Cybernetics*, vol. 72, pp. 365–370, 1995.

[10] K. Matsuno, *Physical Basis of Biology*, CRC Press, Boca Raton, FL., 1989.

[11] Y.P. Gunji and N. Konno, "Artificial Life with Autonomously Emerging Boundaries", *App. Math. Computation*, vol. 43, pp. 271–298, 1991.

[12] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning internal representations by error propagation", in *Parallel Distributed Processing*, D.E. Rumelhart and J.L. Mclelland, Eds. Cambridge, MA: MIT Press, 1986.

[13] H. Maturana and F. Varela, *Autopoiesis and cognition: the realization of the living*, D. Riedel Publishing, Boston, 1980.

[14] S. Wiggins, *Introduction to Applied Nonlinear Dynamical Systems and Chaos*, New York: Springer-Verlag, 1990.

[15] J.F. Kolen, *Exploring the computational capabilities of recurrent neural networks.*, PhD thesis, The Ohio State University, 1994.

[16] I. Tsuda, "Toward an interpretation of dynamic neural activity in terms of chaotic dynamical systems", *Behavioral and Brain Sciences*, vol. 24:5, pp. 793–848, 2001.

[17] J.L. Elman, "Finding structure in time", *Cognitive Science*, vol. 14, pp. 179–211, 1990.

[18] J. Tani, "An interpretation of the "self" from the dynamical systems perspective: a constructivist approach", *Journal of Consciousness Studies*, vol. 5, no. 5-6, pp. 516–42, 1998.

[19] J. Tani and J. Yamamoto, "On the dynamics of robot exploration learning", *Cognitive Systems Research*, vol. 3, pp. 459–470, 2002.

[20] J. Tani and S. Nolfi, "Learning to perceive the world as articulated: an approach for hierarchical learning in sensory-motor systems", *Neural Networks*, vol. 12, pp. 1131–1141, 1999.

[21] J. Tani and M. Ito, "Self-organization of behavior primitives as multiple attractor dynamics by the forwarding forward model network", *Trans On IEEE SMC-B*, 2003, in print.

[22] J. Tani, "Learning to generate articulated behavior through the bottom-Up and the top-down interaction processes", *Neural Networks*, vol. 16, pp. 11–23, 2003.