On the Dynamics of Robot Exploration Learning

Jun Tani and Yuya Sugita

Sony Computer Science Laboratory Inc.
Takanawa Muse Building, 3-14-13 Higashi-gotanda,
Shinagawa-ku,Tokyo, 141 JAPAN
email: tani@csl.sony.co.jp, http://www.csl.sony.co.jp/person/tani.html

Abstract. In this paper, the processes of exploration and of incremental learning in the robot navigation task are studied using the dynamical systems approach. A neural network model which performs the forward modeling, planning, consolidation learning and novelty rewarding is used for the robot experiments. Our experiments showed that the robot repeated a few variation of travel patterns in the beginning of the exploration, and later the robot explored more diversely in the workspace by combining and mutating the previously experienced patterns. Our analysis indicates that internal confusion due to immature learning plays the role of a catalyst in generating diverse action sequences. It is found that these diverse exploratory travels enable the robot to acquire the rational modeling of the environment in the end.

1 Introduction

One of the debates in behavior-based robotics is whether or not agents should possess higher-order cognitive functions such as internal modeling, planning and reasoning. Most researchers in behavior-based robotics have rejected the "representation and manipulation" framework since they consider that the representation cannot be grounded and that the mental manipulation of the representation cannot be situated adequately in the behavioral context of the robot in the real world environment. This argument seems to be valid if the agent's mental architecture employs the symbolist framework. One of the major difficulties in the symbolist framework is that the logical inference mechanism utilized in planning or reasoning assumes completely consistent model of the world. This presumption cannot be satisfied if the learning should be conducted dynamically as in animal and in human adaptation processes. It is, however, also true that the embodiment of higher-order cognitive functions is crucial if we attempt to reconstruct an intelligence at the human level in robots, since even two year-old human infants are said to possess primitive capabilities of modeling and planning within their adopted environment.

We consider that an alternative to the symbolist framework can be found in the dynamical systems approach [4, 1] in which the internal cognitive processes are considered to exist in tight coupling with the external environmental processes [1]. Our previous study in navigation learning demonstrated that a robot

using a recurrent neural net (RNN) is able to learn the "grammatical" structure hidden in the environment, as embedded in attractor dynamics with a fractal structure, from the experiences of sensory-motor interactions [8]. The forward dynamics [3] of the RNN generates a mental image of future behavior sequences driven by the acquired attractor dynamics. The crucial argument in that study is that the situatedness of the higher cognitive processes are explained on the basis of the entrainment of the internal dynamics by the environmental dynamics. However, a drawback of that study was that the learning was conducted in an off-line manner i.e. the navigation could be conducted only after complete learning of the environment.

In the current paper, we study the development of the interactive processes between learning and acting in the robot's exploration of its environment. By conducting real robot experiments, we focus on how the robot interacts with its environment and how it makes sense of the world by utilizing its limited experiences. Our experiment exhibits an interesting result: we find that the diverse exploratory behaviors are generated through taking advantage of the state of confusion in the internal modeling in the middle of the learning process. Our analysis, based on the dynamical systems scheme clarifies the underlying mechanism.

2 The Model

In this section we introduce a neural net model which enables the system to perform exploratory behavior, goal-directed planning and behavior-based learning. The neural net architecture employed has been built by combining pre-existing neural net schemes. In the learning process, both reinforcement learning and prediction learning are conducted [11]. Using reinforcement learning, the action-policies for better rewarding are reinforced, through which the most preferred action in the current state is selected. In prediction learning, the forward model [3] is adapted to extract the causality between the action and the sensation. In goal-directed planning, the inverse dynamics scheme [11, 3] is applied to the forward model in order to generate possible action sequences. In this planning process, the action policy adapted using reinforcement learning provides heuristics for searching for the better rewarded acion sequences. In the current formulation, rewards are given to the system based on the novelty which the system experiences for each exploration action [10, 6]. In other words, when the system cannot predict the next sensation in terms of the current action, the current action is rewarded. In addition, the prediction learning attempts to learn to predict how much prediction error it will make. By combining this novelty-rewarding scheme with the reinforcement learning and with the prediction learning schemes, the system tends to explore the workspace regions with which it is unfamiliar. As the novelty rewarding scheme continues to bring new experiences to the system, the system is forced to operate in a nonequilibrium state in which learning as well as acting cannot always be rationalized. The main purpose of this modeling is to investigate the possible interplay between exploration and learning when the system develops in a nonequilibrium dynamical manner.

2.1 The neural net architecture

A RNN architecture is employed in our model as shown in Fig 1. The RNN receives the current sensory input s_t, the current reward signal r_t, and the current action x_t. The RNN then outputs the prediction of the next sensory input $\hat{s}_{t+1}$, the reward signal $\hat{r}_{t+1}$, and its preference for the next action $\hat{x}_{t+1}$ which is expected to obtain the maximum reward in the future. For the novelty rewarding, the current normalized prediction error for the sensory inputs is used to evaluate the current novelty reward. It is noted that the reward is generated internally and we observe that the RNN learns to predict it (see section 3.2). The RNN has context units c_t in the input and output layers in order to account for the internal memory state (See Ref.[8] for more details of the role of context activation in navigation learning.)

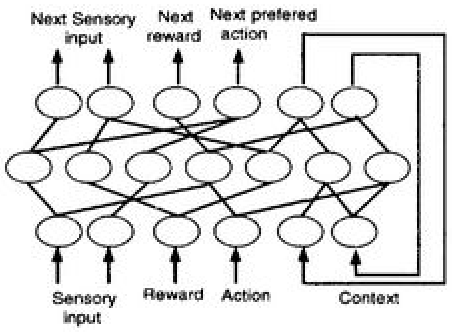

Fig. 1. The RNN architecture.

(A) Learning: The RNN learns to predict the next sensory inputs and the rewards corresponding to the current sensory inputs, the action selection and the internal state. This corresponds to the forward model learning. The preference for the next action is learned by a variant of the profit sharing method [2] in order to propagate the decayed reward signal backwards in time. This means that if the current action selection leads to an unpredictable experience, this action selection is reinforced. This corresponds to reinforcement learning. Both learning processes are executed in the RNN using the back-propagation through time (BPTT) algorithm.

(B) Planning: The objective of planning is to find the action plan $x*$ $|(x_0, x_1, ...x_\tau)$ which generates the path to maximize the future calmative rewards. The action sequence is dynamically computed by using contributions both from the forward model part and from the action policy part. Inverse dynamics [3] are applied to the forward model in order to obtain the update of

the action plan $\Delta x*$ for maximizing the calmative reward expected in the future sequence. We consider the following energy function by taking the negative of the calmative reward from the current time step to the terminal step τ:

$$Em(x*) = -\sum_{i=0}^{\tau} \alpha^i \hat{r}_{i+1} \qquad (1)$$

where α is the decay coefficient of the reward. The back-propagation through time (BPTT) algorithm [5] is used to compute the update to the action sequence which minimizes the energy assumed in the model part. In addition to this, the action policy influences the planning dynamics in that the difference between the preferred action and the planned action at each step is minimized. The update to the action at each future step is obtained by taking the sum of both parts of the contributions and adding a Gaussian noise η. The update to the action plan is therefore

$$\Delta x_i = \epsilon \cdot [\frac{-\delta Em(x*)}{\delta x_i} + kr \cdot (\hat{x}_i - x_i) + kn \cdot \eta] \qquad (2)$$

The Gaussian noise term is employed to prevent the plan dynamics being captured in a local minimum. The value of kn is changed in proportion to the value of Em. Therefore the plan search dynamics become stabilized when the energy is minimized; otherwise, it continues to be activated. Here, the reader is reminded that the contributions to the update from the forward model and from the action policy do not always agree with each other in the course of the exploration processes since the overall system dynamics are characterized by highly nonlinear and nonequilibrium dynamics.

(C) Incremental learning by consolidation: The robot learns what it experienced incrementally after each travel is terminated by using the so-called consolidation learning scheme [9] which has been developed as inspired by the biological observation of the memory consolidation [7] during sleep in mammals. In our system, a new episodic sequence experienced in the current travel is stored in the temporal memory. In the consolidation process, the RNN generates the imaginary sensory action sequence by rehearsing from the long term memory prelearned. This rehearsal can be performed by repeating "planning", as described in the previous section, without actually moving - as in dreaming. Then, the RNN is re-trained using both the new episodic sequence stored in the temporal memory and the rehearsed sequences generated from the pre-learned memory simultaneously. This combination of rehearsal and learning allows the memory system to be re-organized without suffering from some catastrophic interference between the novel experiences and the pre-learned memory.

3 Experiment

3.1 Task setting

A mobile robot as shown in Fig 2 (a) is used for the experiment. The robot is equipped with range sensors and a color vision camera on its head. Fig 2 (b)

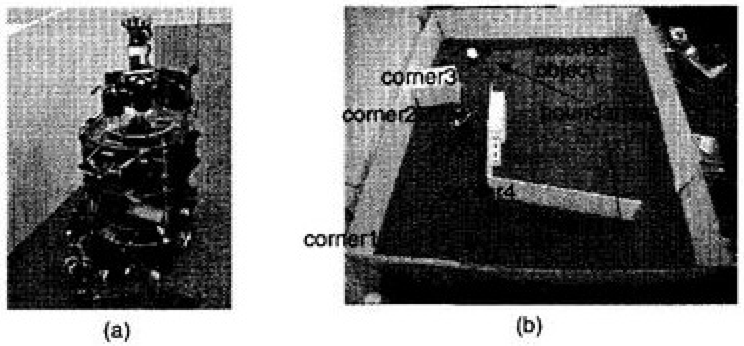

Fig. 2. (a) The robot employed in the experiment and (b) the adopted workspace.

shows the adopted workspace. The basic behavior of the robot is determined using a set of pre-programmed action modules for actions such as wall-following, wall-switching and colored-object-approach. The actions are switched between by the RNN using branching. Two cases of branching are considered: (a) the robot, after turning a corner, determines whether it will continue to follow the current wall on its left side or instead to leave the current wall and to move forward diagonally at 45 degrees to the right until it encounters another wall; and (b) the robot, after finding a colored object, determines whether to continue the wall-following or to approach the colored object. In this setting, the action can be represented by one bit of information which represents whether or not to branch. The RNN architecture receives two types of sensory input at each branch point. One is the travel vector which represents what distance and from which direction the robot has traveled since the previous branch. These values are measured by taking the sum and the difference between the left and right wheels's rotation angles. The other sensory input is the categorical output of the visual image obtained when the robot encounters a colored object. The robot plans its future action sequence dynamically while it travels and receives the sensory inputs at each branch encountered. The robot starts its exploration travel from a fixed home position and the exploration is terminated when the travel takes it outside a predefined boundary. (The home position and the boundary predefined in our experiment is shown in Fig 2 (b).) At the moment of termination, the RNN receives the termination sign in its sensory input and the robot is brought back to the home position manually. Following this, the consolidation process takes place in which the temporary stored sequence is learned using 10 rehearsal sequences. After the consolidation, exploration by the robot is resumed.

3.2 Results

The robot repeated the exploration travels 20 times in the experiment. This experiment was conducted three times under the same conditions. Fig 3 represents the average prediction error for each travel sequence in the three experimental cases. For all three cases, on average that the prediction error gradually decreases as the exploration is proceeds.

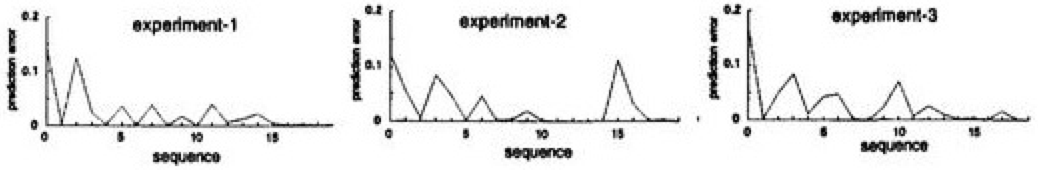

Fig. 3. The history of the prediction error for the three experiment cases.

It is interesting to observe the rehearsing during the consolidation learning since the contents of the rehearsing activities represent what the robot has learned so far. Fig 4 shows how the diversity of the rehearsed plans at each consolidation learning process change as the exploration proceeds. The lower graphs in the figure shows ID of all rehearsed plans generated during each consolidation learning period; the upper graph represent the corresponding predicted rewards of the plans generated. (The ID is assigned for each plan generated by encoding the bit pattern of the branching sequence, a maximum of 10 time steps in length, into numbers from 0 to 512.) It is observed that the diversity of plans is increased and that the predicted reward is decreased as the exploration trial is continued. We observed that the rehearsed plans are generated not just by repeating the

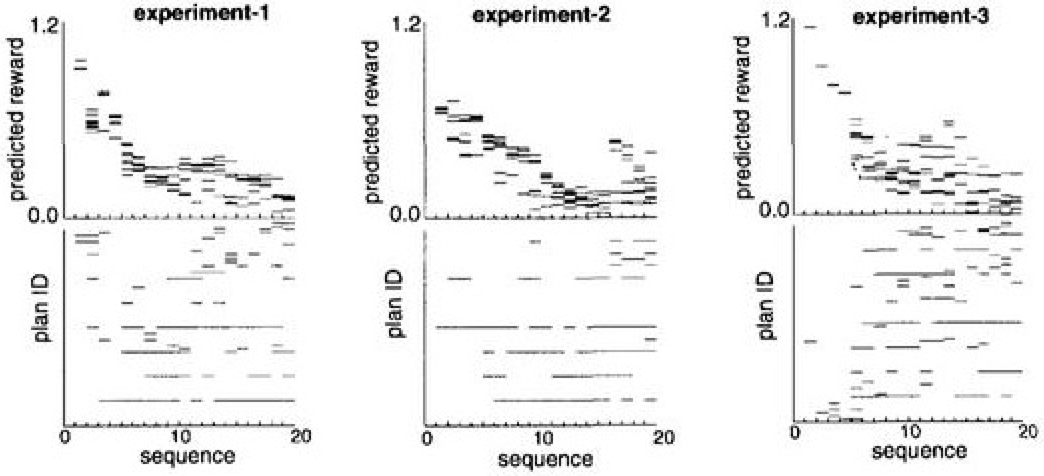

Fig. 4. Changes in the diversity of the rehearsed plans during the three exploration experiments.

sequences previously experienced but by combining previously experienced sequences into new ones. Since the rehearsing directly affects the re-organization of the learned contents, the diversity generated in rehearsing leads to the diversity in actual travel.

In the following, we examine how the diverse travel sequences are generated in the course of exploration. Fig 5 shows all 20 trajectories of the robot's travel observed in one experimental case (experiment-1). In the initial period of the exploration, the robot tends to repeat the same branching sequences. As is evident in Fig 5, the same trajectory is repeated for the first two travel sequences. For the third sequence, branching changes and a different trajectory is generated. This trajectory is repeated in the next two travel sequences. The trajectory in the sixth travel sequence seems to be generated by combining the two travel sequences previously experienced. We summarize that the novelty rewarding scheme causes the observed repetitions and variations in the travel. When the robot undergoes a previously unexperienced travel sequence, the branching sequence experienced is reinforced strongly because of its unpredictability. When the same trajectory is repeatedly generated through reinforcement, the sequence becomes predictable and is rewarded less. As a result, the probability of modifying the current travel is increased.

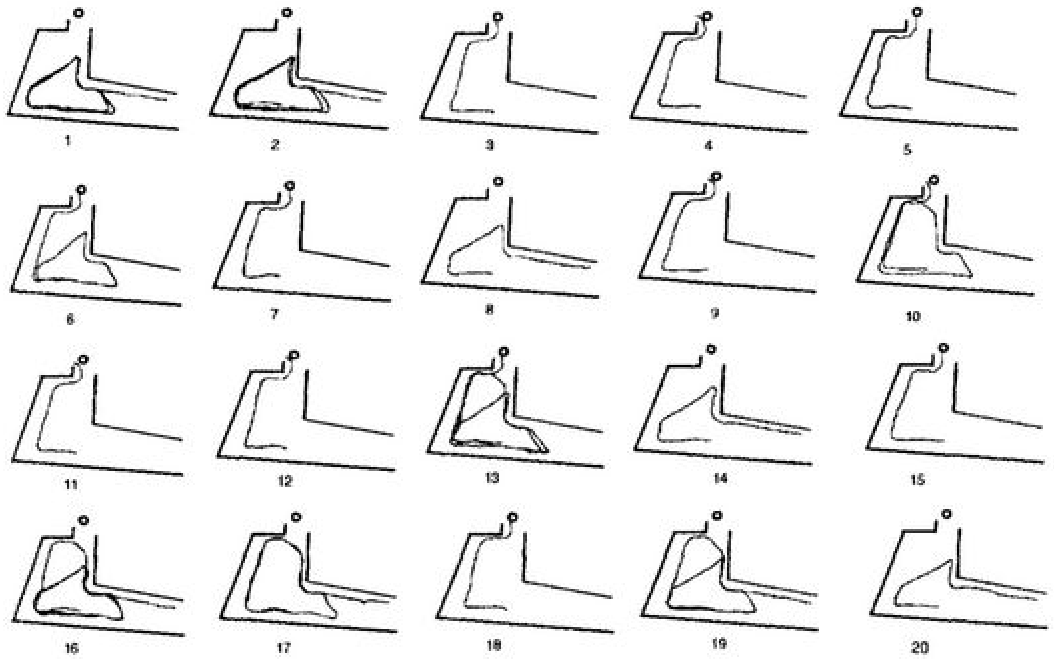

Fig. 5. The trajectories of the robot exploration travel for one experimental case. The travel sequence number is given.

An interesting question is how novel action sequences are generated in the planning process. What we found is that novel branching sequences are originated not merely by the noise term in the planning dynamics but also by the internal confusion caused by the incremental learning. This point is illustrated by considering an example seen in the 10th travel sequence. In this travel sequence, the robot, starting from the home position, continued to follow the wall after passing corner1, then it branched to another wall after passing corner2. This branching at corner2 is a novel experience for the robot. We investigated how this branching decision was generated by examining the recorded planning process. Fig 6 shows the actual planning processes which took place immediately before the branching was made at corner2. In Fig 6 (a) each column consisting of white and black squares represents a branching sequence plan at each time step of the planning process, where the black and white squares denote branching and non-branching, respectively. Fig 6 (b) indicates the predicted reward for the plan generated. At the beginning of the planning process, a plan of not branching twice is generated with a low predicted reward. This plan will repeat the 5th travel sequence if actually realized. At the end of the planning process, plans are generated such that branching actions are planned to occur repeatedly after passing corner2 with an expectation of a higher reward, even though such action sequences have never been experienced. It is noted that this type of plan was not observed when the robot approached the same corner in its earlier travels. Further examination showed that the lookahead prediction of the sensory sequences after branching at corner2 and at corner1 are mostly the same. This can be interpreted as meaning that the robot hypothesized that branching at any corner would lead to better chances for encountering novel experiences because it applied the situation after branching at corner1 to consider the situation at corner2. (Indeed, the travel will continue as long as branching is selected at approaching corners without terminating the travel by going out of the workspace

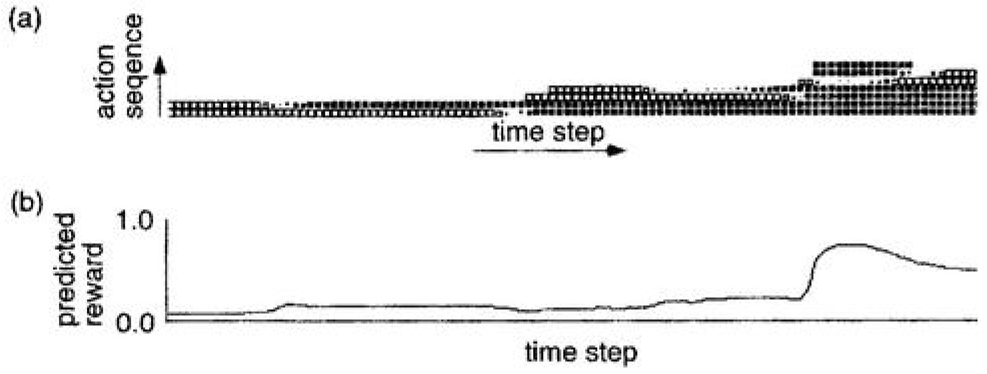

Fig. 6. (a) The time history of plan generation at the corner2, (b) the predicted reward of the corresponding plan.

boundary.) We conclude that the novel action of branching at corner2 results from the expectation of a higher reward which is falsely anticipated by means of fake memory generated in the course of consolidating immature experience. This phenomenon of the novel action trial being generated by fake memory and the internal confusion was seen frequently in the middle of the learning process.

Finally, we investigated how the internal modeling develops by examining the evolution of the RNN attractor. Fig 7 shows the attractor which appeared in the phase space of the RNN at different stages in experiment-1. The phase plots were drawn by iteratively activating the RNN in the closed-loop mode with inputs comprising 4000 steps of random branching action sequences. The generated sequence of the context units activation are plotted in the two dimensional phase space. In Fig 7, cluster structures consisting of multiple segments are clearly seen in the later periods of the exploration travel. Our examination clarified that this set of cluster segments represents the global attractor. Further analysis indicated that in the phase plots in Fig 7 (c) and Fig 7 (d) each segment corresponds uniquely to each branching position in the workspace and also that the graph structures are topologically equivalent between that of the state transition in the phase space and that of branching of the robot trajectories in the environment. In this condition, it is said that the "dynamical closure" is generated in the attractor since an equivalence of the closed graph structure is generated in the phase space. However, such structures were barely seen in the phase plots in Fig 7 (a) and Fig 7 (b). While the learning process is "immature", the shape of the attractor varies substantially after each learning and neural dynamics exhibits diverse trajectories in the phase space and the robot behaves as if it were confused. In the meanwhile, the attractor develops step by step as the diverse exploration repeated and finally the dynamical closure is organized in the internal neural dynamics.

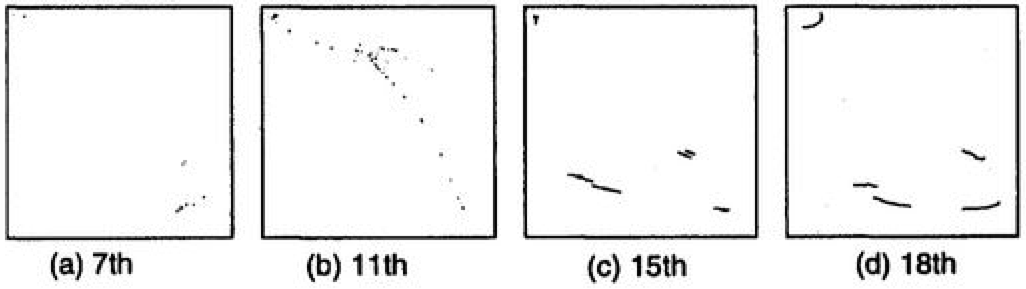
(a) 7th (b) 11th (c) 15th (d) 18th

Fig. 7. The RNN attractor appeared at a certain stage of the learning process in experiment-1. The learning stage is given at the base of each plot.

4 Discussion and Conclusion

In the experiments, it was shown that the robot learned incrementally about its workspace through exploration and that the robot was eventually successful in obtaining a rational model of the workspace. However, the emphasis in this study is on the observation of dynamical processes before the rational model is achieved. In the beginning, a few travel sequences are repeated and later some combinations of them are made. In the middle period, novel actions are frequently tried with a false expectation of the future consequences. The confusion due to the immaturity turns out to be beneficial since it acts as a catalyst for generating the diverse behavior required to explore the environment. Such diverse behavior enables the robot to acquire the rational model later.

Our experimental studies, however, are limited in a sense that (a) the robot is manually recovered when it goes out of the workspace boundary, (b) the environment is static. Our future study will address these problems.

References

1. R.D. Beer. A dynamical systems perspective on agent-environment interaction. *Artificial Intelligence*, Vol. 72, No. 1, pp. 173–215, 1995.
2. J.H. Holland and J.S. Reitman. Cognitive systems based on adaptive algorithms. In D.A. Watermann and F. Hayes-Roth, editors, *Pattern Directed Inference Systems*. New York: Academic Press, 1978.
3. M.I. Jordan and D.E. Rumelhart. Forward models: supervised learning with a distal teacher. *Cognitive Science*, Vol. 16, pp. 307–354, 1992.
4. J.B. Pollack. The induction of dynamical recognizers. *Machine Learning*, Vol. 7, pp. 227–252, 1991.
5. D.E. Rumelhart, G.E. Hinton, and R.J. Williams. Learning internal representations by error propagation. In D.E. Rumelhart and J.L. Mclelland, editors, *Parallel Distributed Processing*. Cambridge, MA: MIT Press, 1986.
6. J. Schmidhuber. A possiblity for implementing curiosity and boredom in model-building neural controllers. In J.A. Meyer and S.W. Wilson, editors, *From Animals to Animats: Proc. of the First International Conference on Simulation of Adaptive Behavior*, pp. 222–227. Cambridge, MA: MIT press, 1991.
7. L.R. Squire, N.J. Cohen, and L. Nadel. The medial temporal region and memory consolidation: A new hypothesis. In H. Weingartner and E. Parker, editors, *Memory consolidation*, pp. 185–210. Erlbaum, Hillsdale, N.J., 1984.
8. 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.
9. 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.
10. S.B. Thrun and Knut Moller. Active exploration in dynamic environments. In *in Proc. of NIPS 4*, pp. 531–538. 1990.
11. P. Werbos. A menu of designs for reinforcement learning over time. In W.T. Miller, R.S. Sutton, and P.J. Werbos, editors, *Neural Networks for Control*, pp. 67–95. MIT Press, Boston, MA, 1990.