Proceedings 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation
July 16-20, 2003, Kobe, Japan

Generating Optimal Behaviors of Mobile Robot using Behavior Network with Planning Capability*

Hyeun-Jeong Min, Kyung-Joong Kim and Sung-Bae Cho
Deptartment of Computer Science
Yonsei University
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
E-mail : solusea@candy.yonsei.ac.kr, uribyul@candy.yonsei.ac.kr, sbcho@csai.yonsei.ac.kr

Abstract

This paper presents an approach to the optimal method for behavior generation of mobile robot. Recently, a hybrid system that supplements gap between reactive and plan-based approaches by employing a reactive system for lower level control for fast reaction and a planner for higher level for generation of optimal sequence of behaviors has gained popularity. Behavior network structures may generate behaviors automatically through the internal links and external links with sensors and goals, and can be applied much into more complex problems. In this paper we propose an optimization method of behavior sequences for generating optimal behavior of mobile robot using behavior network with planning capabilities. Behavior network inspired by behavior selection of animals selects the next behavior that has the highest priority from the information acquired to sensors and goals. Globally optimal behavior sequences are able to be generated when all behaviors selected in this behavior network are processed in advance and the next behavior is selected among behaviors that approach the goals. We could notice that robot reaches the goals faster in behavior network with planning capability rather than behavior network only through the experiment using Khepera mobile robot simulator.

1 Introduction

Traditional artificial intelligence (AI) that aims to replicate human level intelligence in a machine form has concerned itself with complex systems in simple environments that have no noise and uncertainty and could be precisely modeled. In contrast to traditional plan-based systems that plan optimal sequences in predefined environment, behavior-based systems are able to react and perceive in complex, noisy and uncertain environments [1, 2]. Behavior-based systems embody the mapping between the conditions and actions of reactive systems for fast reaction, and may be computed without limitations of purely reactive systems. These systems perform computations on them in order to decide what effector action to take. The behaviors in behavior-based system are more powerful than reactive rule, condition-action pairs [3]. These behavior-based robot systems can operate robustly in the presence of uncertainty, but they have limitations that they are still designed by hand initially and it is difficult to apply in complex problem. For solving these limitations behavior network, generating behaviors through links of behavior automatically as well as by hand, has been exploited.

Corresponding to the correlation of traditional artificial intelligence that confers human level intelligence in machines and the ecology studying about behavioral habit of animals, the action selection mechanisms making appropriate behaviors in mobile robots using the behaviors of animals in nature are proposed by ecologists and robotics engineers [4, 5, 6]. The action selection mechanism based on ecology is the method to generate behaviors of machines artificially [6]. The behavior network, one of these action selection mechanisms, selects the best behavior in a certain environment through linkage of basic behaviors and the information of sensors and inner goals. This is a useful way to generate high-level behaviors efficiently by the interaction of basic behaviors. Behavior network may acquire the information from motivations or goals and sensors of outer spaces. It executes the behaviors selected by the highest activation value using inter-connection information of behaviors in module in addition to the information of motivations or goals, or outer spaces.

The flexible behaviors can be generated using hybrid architecture of combining elements of reactive behaviors and deliberative planning of behaviors in uncertain and noisy environment [7]. One of the techniques of planning is potential-field planning that escapes local minima by performing random walks, and another technique is a "roadmap" connecting randomly selected configurations in configuration space, the degrees of freedom of the robot [8, 9, 10]. Besides, it has made progress to study about hybrid of deliberate planning and reactive behaviors in unknown environments [11]. In this paper we extend behavior network with planning capability for the shortest behavior sequences or faster achievement than those in behavior network without planning. This approach may generate optimal sequences of behaviors up to goals. To combine planning capability we process all of candidate behaviors in each level of behavior search tree in advance whenever we select n behaviors in behavior network, and select the best behavior in the last level. Then we produce the optimal sequences suited among n behaviors selected. The experiment with Khepera shows that the method using behavior network with planning capability produces the optimized sequences of behaviors.

*This research was supported by Brain Science and Engineering Research Program sponsored by Korean Ministry of Science and Technology, and Korea Institute of Science and Technology.

0-7803-7866-0/03/$17.00 ©2003 IEEE

2 Background

2.1 Behavior Network

Figure 1 shows the structure of behavior network. As shown in this figure, behavior network has basic behaviors, the information of sensors and goals. The sensors are preconditions of each behaviors and the goals are effects to achieve.

Behavior network is composed of behaviors, internal links and external links. Each behavior has a set of preconditions, add list, delete list, activations and execution codes. Each behavior has preconditions, and must fulfill all preconditions to be executed. The preconditions are logical conditions about the environment required to be true in order to execute the behavior. Add list is a set of conditions that the behavior is likely to make true after the behavior is executed, and delete list is a set of conditions that the behavior is likely to get false after a behavior is executed [12].

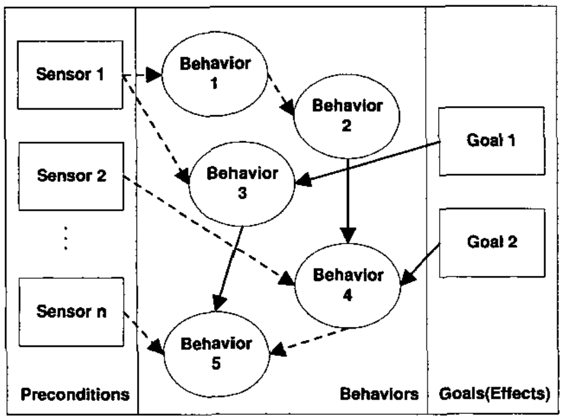

Figure 1: An example of behavior network. The dashed lines denote successor links or sensors and the solid lines denote predecessor links or goals.

The internal links are specified as follows:

- Predecessor link : If proposition X is false, proposition X is a precondition of behavior A, and proposition X is in the add list of behavior B, then the link from A to B is an active predecessor link.
- Successor link : If proposition X is false, proposition X is in the add list of behavior A, proposition X is a precondition of behavior B, and the behavior A is executable, then the link from A to B is an active successor link.

The external links are specified as follows:

- If proposition X about the environment is true, and proposition X is a precondition of behavior A, then there is an active link from the sensor of the proposition X to behavior A.
- If goal Y has an activation greater than zero, and goal Y is in the add list of behavior A, then there is an active link from the goal Y to behavior A.
- If goal Y has an activation greater than zero, and goal Y is in the delete list of behavior A, then there is an active link from the goal Y to behavior A.

2.2 Representation of Behavior Network

Competition of behaviors is the basic characteristics of behavior network. Each behavior attempts to get higher activation level than other behaviors from activation spreading in forward and backward directions. Among candidate behaviors, one that has the highest activation level is selected and has control of robot. Activation level a of behavior is calculated as follows.

- Forward propagation : Activation a is updated as the value added by environmental sensors that are precondition of the behavior. Precondition is the sensor that is likely to be true when the behavior is executed. n means the number of sensors. a_s is the activation level of the sensor.

$$\Delta a_1 = \sum_{i=1}^{n} f(a_{s_i})$$

$$f(a_{s_i}) = \begin{cases} \phi \times a_{s_i}, & s_i \in \text{precondition} \\ 0, & s_i \notin \text{precondition} \end{cases}$$

- Backward propagation : Activation a is updated as the value added by goals that are directly connected to the behavior. If execution of the behavior is desirable for the goal, there is positive goal-behavior link. On the other hands, there is negative goal-behavior link. n means the number of goals. a_g is the activation level of the goal.

$$\Delta a_2 = \sum_{i=1}^{n} f(a_{g_i})$$

$$f(a_{g_i}) = \begin{cases} \gamma \times a_{g_i}, & g_i \in \text{positive link} \\ -\delta \times a_{g_i}, & g_i \in \text{negative link} \end{cases}$$

- Internal spreading : Activation a is updated as the value added by other behaviors that are directly connected. If execution of behavior B is desirable for behavior A, predecessor link from A to B and successor link from B to A is active. If execution of behavior B is not desirable for behavior A, conflictor link from A to B is active. n is the number of behaviors. a_b is the activation level of the behavior.

$$\Delta a_3 = \sum_{i=1}^{n} f(a_{b_i})$$

$$f(a_{b_i}) = \begin{cases} a_{b_i}, & \text{predecessor link from } b_i \\ \phi/\gamma \times a_{b_i}, & \text{successor link from } b_i \\ -\delta/\gamma \times a_{b_i}, & \text{conflictor link from } b_i \\ 0, & \text{otherwise} \end{cases}$$

At the end, the activation of a is updated as follows.

$$a' = a + \Delta a_1 + \Delta a_2 + \Delta a_3$$

If the activation level a' is larger than threshold θ and precondition of the behavior is true, the behavior becomes a candidate for execution. Among candidate behaviors, the highest activation behavior is selected for execution. If there is no candidate behavior, threshold θ is reduced by 10 % and the activation update procedure is repeated until there are candidate behaviors.

3 Behavior Network with Planning

The behavior network selects behaviors to achieve the goals using the information of sensors, and planning makes behavior search tree and selects the optimal sequences of behaviors. First of all, we have to make the behavior network and select the optimal behaviors in generated behavior network. For generating the optimal sequences of behaviors we have to choose an appropriate level and process behaviors up to that level. We have to consider time complexity for selecting the level. If we suppose to select k levels in behavior network that can select maximum n behaviors, the time complexity is $O(nk)$. We have to define the k value considering this time complexity. After selecting k levels, we can generate behavior sequences from parent nodes of selected behavior node that gets near to goals. Through many repetition of this procedure up to goals we generate the optimal sequences of behavior globally.

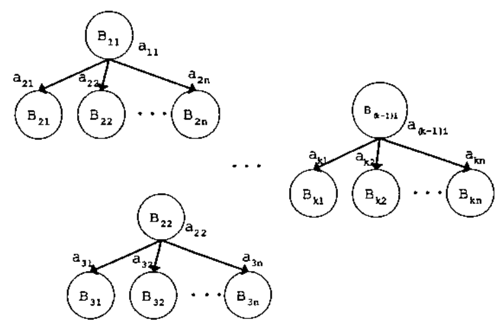

Figure 2: Behavior search tree selects maximum n behaviors and processes from root to the k-th level. We can acquire maximum n^k behavior sequences at the level.

Figure 2 shows the generation of behavior search tree that stores all the selected behaviors to level k. In this figure B_{ij} is the j-th behavior of the i-th level and a_{ij} is the activation value of B_{ij}. In level k, we select behavior that gets near to goals and can generate optimal behavior sequences through search in reverse direction. The function selecting optimal behavior in level k is define as follows.

$$B_{optimize} = \underset{c}{\operatorname{argmax}} \{f_{goal}(B_c) + a_{max}\}$$

In this function $a_{max} = max\{a_{ij} | i = k \text{ and } 0 < j \leq n\}$ and $f_{goal}(B_c)$ is the function to find the behavior such that preconditions are to be true among behaviors getting near to goals. In figure 2 the behavior sequences are $B_{11} \Rightarrow B_{22} \Rightarrow \cdots \Rightarrow B_{(k-1)i}$. For generating optimal behavior sequences we have to select behavior that gets near to goals among maximum n^k behaviors selected in level k of behavior network. We can find the optimal behavior sequences globally when the behaviors in behavior search tree are iterated up to the global goals.

Figure 3 shows the flowchart to find the globally optimal sequences of behaviors till the global goals are achieved. This flowchart shows that we can find optimal sequences through iteration of the procedure after making the behavior search tree. This procedure is terminated if mobile robot reaches the goals.

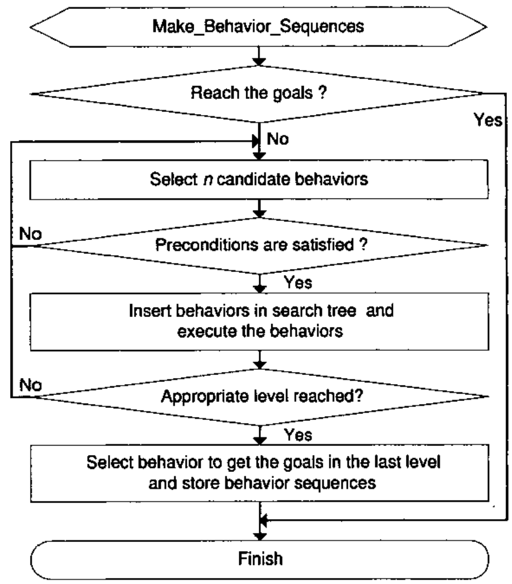

Figure 3: The flowchart of making the behavior sequences.

The selection of behaviors follows Maes's action selection mechanism using the activation value of behaviors in behavior network. Therefore, the selected behaviors have to satisfy the preconditions of that behavior as well as large activation value. The behaviors selected by these procedures must be put in the behavior search tree. Finally process the behaviors in the behavior search tree and find optimal behavior in the last level of behavior search tree. Behavior sequences are generated by linking parent behaviors (nodes) of optimal behavior in level k.

4 Experimental Results

For the experiment, we use the simulator of Khepera mobile robot with four behavior modules. We require that this robot should recharge the battery at low battery level to go around the environment. Recharging battery behavior is conducted only when robot is inside battery recharge area and battery of robot is less than a half. For this problem, we use the combination of several basic behaviors evolved on CAM-Brain and programmed under behavior network [13]. We first use planning for generating the optimal behavior sequences when Khepera mobile robot moves into battery recharge area. Then we compare the case of using the behavior network with planning capability. Khepera mobile robot optimizes behavior sequences applying information about sensors in practical use. Figure 4 shows the simulation environment. The light source is in the battery recharge area as shown in this figure.

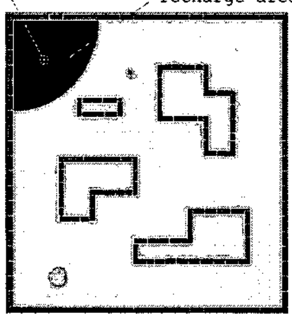

Figure 4: Experimental environment.

4.1 Behavior Network

There are four behaviors, five sensors and two goals, and behavior search tree is constructed by the behavior network planning. Four basic behaviors of Khepera mobile robot are defined as follows.

- Recharging Battery: When a robot is in battery recharge area, battery of mobile robot is recharged.
- Following Light: Mobile robot goes to stronger light. This behavior has to be operated to go to the battery recharge area because the light source exists in that area.
- Avoiding Obstacle: When the obstacles are detected around the robot, robot avoids them without bumping.
- Going Straight: If there is nothing around the robot, it goes ahead. This behavior takes it to move continuously without stopping.

We define 5 preconditions using the information from sensors of robot under behavior network, such as "In battery recharge area," "Obstacle is close," "Near battery recharge area," "Light is low," and "Nothing around robot." They are set as follows.

- "In battery recharge area": It is true if robot is in battery recharge area.
- "Near battery recharge area": It is true if the distance from robot to light source is less than 800.
- "Obstacle is close": It is true if the maximum value of distance sensors is larger than 700.
- "Light is low": It is true if the minimum value of light sensors is larger than pre-defined value.

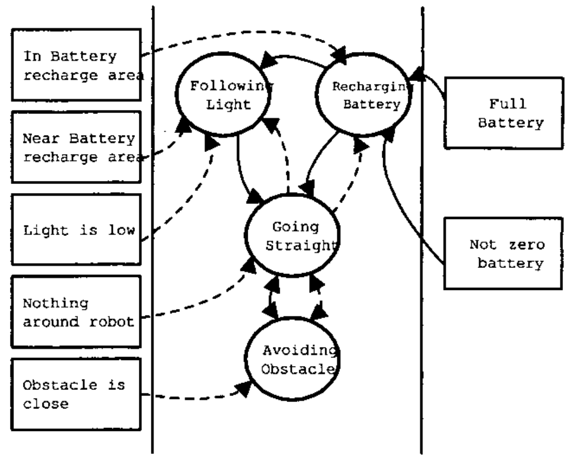

Figure 5: Behavior network of Khepera mobile robot in this experiment.

- "Nothing around robot": It is true if the maximum value of distance sensors is less than 700.

We set 2 goals such as "Full battery," and "Not zero battery." Because robot's battery decreases while robot moves, the robot attempts to maintain high battery value to operate long. They are set as follows.

- "Not zero battery" : It is true if battery is less than half of the maximum battery.
- "Full battery" : $c = \frac{m-n}{m}$
  c : value of "Full battery,"
  m : maximum battery,
  n : robot's battery

4.2 Behavior Search Tree

The behavior network for this experiment is composed of four basic behaviors, five sensors and 2 goals, and behavior search tree is constructed by behavior network with planning. The behavior search tree is satisfied as follows:

- If all 4 behaviors are selected in the n-th level, there are at maximum 4^n behaviors (nodes). This experiment selects 8 levels, and 2 behaviors satisfied preconditions with the greatest activation value in behavior network.
- Select behaviors of which preconditions are true and the activation value is greater than that of other behaviors. Then that behaviors are the children of the next node in search tree.
- If only one behavior is satisfied preconditions, the next node has one behavior. Therefore, there are maximum 2^8 behaviors in the last level of behavior search tree.

We make the behavior search tree as above explanation and then the optimal behavior sequences are selected by values of robot's light sensors. Because the mobile robot goes to light source for recharging the battery. Table 1 describes preconditions and add lists of the behaviors.

Table 1: Preconditions and add lists of behavior network.

	Preconditions	Add lists
Recharging Battery	In battery recharge area	Full Battery, Not zero battery
Following Light	Light is low, Near battery recharge area	In battery recharge area
Avoiding Obstacles	Obstacle is close	Nothing around robot
Going Straight	Nothing around robot	Obstacle is close, In battery recharge area, Near battery recharge area

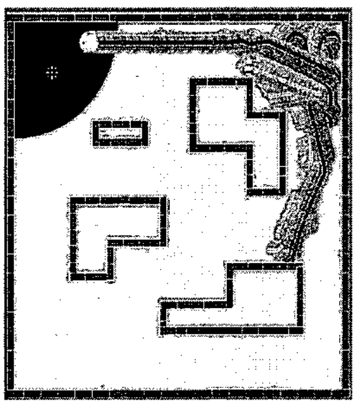

Figure 6: The simulation result using behavior network. Recharging battery behavior is executed only once.

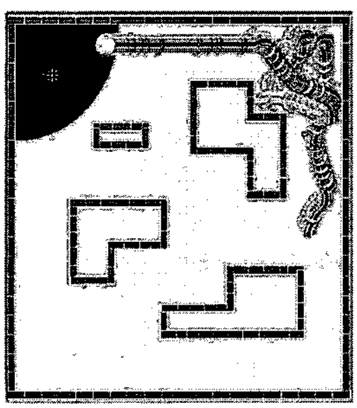

Figure 7: The simulation result using the behavior network with planning capability. Recharging battery behavior is executed only once.

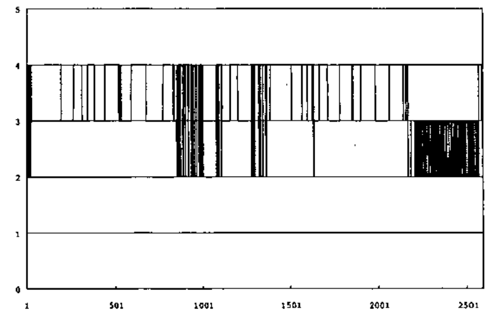

Figure 8: Behavior sequences of the simulation result using only behavior network. The x-axis is the number of behavior selections and y-axis shows behavior indexes.

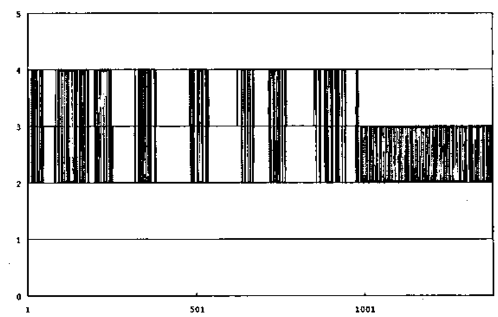

Figure 9: Behavior sequences of the simulation using behavior network with planning capability. The x-axis is the number of behavior selections and y-axis shows behavior indexes.

Each behavior has one or two preconditions. Figure 5 shows the relationship of behaviors, preconditions and goals in behavior network of this experiment. In this figure, the dashed lines are successor links or sensors and the solid lines are predecessor links or goals.

The data structure for this tree is defined as follows:

- BTree: The structure of behavior tree having level of tree, array, behavior and pointer of next node.
- B_COUNT : The number of candidate behaviors ($n = 2$ for this experiment).
- L_LEVEL : The level of behavior search tree ($k = 8$ for this experiment).

4.3 Results and Analysis

We compare behavior sequences in behavior network with behavior network with planning capability. We simulate under several configurations in environment. Figure 6 shows a simulation result of behavior network. Robot in this environment executes 2586 times of behaviors selection up to recharge battery. The behavior sequences defined such as 1 is "Recharging battery," 2 is "Following light," 3 is "Avoding obstacles," and 4 is

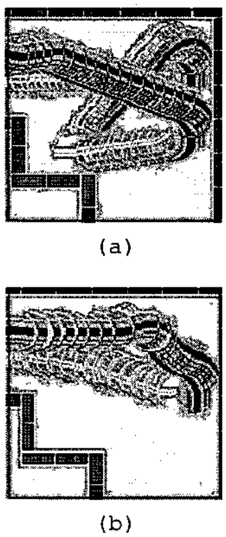

Figure 10: Robot's trajectory in behavior network (a) and in the behavior network with planning capability (b). The background of this experiment is in arbitrary area of the same environment.

"Going straight" on figure 6 are shown in figure 8. Figure 8 shows the simulation result in the behavior network with planning capability. Planning is up to 8 levels locally and the behavior of the greatest light value in each level 8 is selected. This occurs total 1376 times of behaviors selection up to reach the goal, recharge battery. The behavior sequences of figure 7 are shown in figure 9.

Now, we show the trajectory of Khepera mobile robot using the behavior network and the behavior network with planning capability respectively. Figure 10 shows the comparison about the trajectory of robot's behavior in the same local area. (a) is the simulation of only behavior network, and (b) is the simulation of the behavior network with planning capability. The behaviors are generated 670 times in behavior network, while the behaviors are generated 200 times in the behavior network with planning capability.

5 Conclusions and Future Works

This paper develops a framework for generating robust planning behavior in uncertain and noisy environments. We have analyzed the influence of planning in behavior network through the simulation of Khepera mobile robot in behavior network with planning capability. Traditional AI planning systems do one thing: they form plans, that is, optimal sequences of actions that lead to a specified goal state when executed in a specified environment. Behavior network makes behaviors with low levels of cognitive complexity into complex, noisy and uncertain environments. We have tried to find the optimal sequences of behaviors using the behavior network with planning capability.

We have compared the behavior sequences under only behavior network with under the behavior network with planning capability through the experiment. The result of the experiment shows that the sequence of behaviors under the connection of planning and behavior network are less than under only behavior network. We can find the optimal sequence of behaviors up to goals if we have well- defined level and optimal behaviors in local. The approach is general planning algorithms, and should thus serve to make and formulate planning more efficient in a more dynamic environment in the future.

References

[1] R. A. Brooks. Intelligence without representation. *Artificial Intelligence Journal*, 47:139–159, 1991.

[2] M. J. Mataric. *Interaction and Intelligent Behavior.* Ph.D.Thesis, 1994.

[3] M. J. Mataric. Behavior-based control: Main properties and implications. In *IEEE International Conference on Robotics and Automation*, pages 46–54, 1992.

[4] P. Maes. How to do the right thing. In *Connection Science*, pages 291–323, 1989.

[5] P. Maes. Situated agents can have goals. *Journal for Robotics and Autonomous Systems*, 6:49–70, 1990.

[6] P. Maes. Bottom-up mechanism for behavior selection in an artificial creature. In *Proceedings of the First International Conference on Simulation of Adaptive Behavior*, pages 238–246, 1991.

[7] R. C. Arkin and D. C. Mackenzie. Planning to behave: A hybrid deliberative/reactive robot control architecture for mobile manipulation. In *Proceedings of International Symposium on Robotics and Manufacturing*, pages 5–12, 1994.

[8] J. Barraquand L. Kavraki J. C. Latombe T. Y. Li R. Motwani and P. Raghavan. A random sampling scheme for path planning. In *Proceedings of the 7th International Symposium on Robotics Research*, pages 249–264, 1996.

[9] M. Piaggio and R. Zaccaria. Using roadmaps to classify regions of space for autonomous robot navigation. *Robotics and Autonomous Systems*, 25:209–217, 1998.

[10] J. Bruce and M. Veloso. Real-time randomized path planning for robot navigation. In *Proceedings of the 2002 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems EPFL, Lausanne, Switzerland*, pages 2383–2388, 2002.

[11] R. Onai S. Kurihara, S. Aoyagi and T. Sugawara. Adaptive selection of reactive/deliberate planning for a dynamic environment. *Robotics and Autonomous Systems*, 24:183–195, 1998.

[12] K.-J. Kim and S.-B. Cho. Robot action selection for higher behaviors with CAM-Brain modules. In *Proceedings of the 32nd International Symposium on Robotics*, pages 1623–1628, 2001.

[13] S. B. Cho and G. B. Song. Evolving CAM-Brain to control a mobile robot. *Applied Mathematics and Computation*, 111:147–162, 2000.