BN+BN: Behavior Network with Bayesian Network for Intelligent Agent

Kyung-Joong Kim and Sung-Bae Cho

Dept. of Computer Science, Yonsei University
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
uribyul@candy.yonsei.ac.kr
sbcho@cs.yonsei.ac.kr

Abstract. In the philosophy of behavior-based robotics, design of complex behavior needs the interaction of basic behaviors that are easily implemented. Action selection mechanism selects the most appropriate behavior among them to achieve goals of robot. Usually, robot might have one or more goals that conflict each other and needs a mechanism to coordinate them. Bayesian network represents the dependencies among variables with directed acyclic graph and infers posterior probability using prior knowledge. This paper proposes a method to improve behavior network, action selection mechanism that uses the graph of behaviors, goals and sensors with activation spreading, using goal inference mechanism of Bayesian network learned automatically. Experimental results on Khepera mobile robot show that the proposed method can generate more appropriate behaviors.

1 Introduction

There are many Action Selection Mechanisms (ASMs) to combine behaviors for generating high-level behaviors including spreading activation network, subsumption architecture and hierarchical ASM [1]. The ASM is essential in behavior-based robotics because it selects appropriate one among candidate behaviors and coordinates them. Usually, ASMs cannot insert goals into the model in explicit or implicit manner. Behavior network, one of ASM, can contain goals of robot in implicit manner and propagates activation of behaviors in two directions (forward and backward) through the network for dynamic selection [2].

Behavior network can have many goals that need to be active in different environments and status. User can insert prior knowledge of goal activation into behavior network in the design stage of behavior network but it is difficult to capture and represent the knowledge. There are some computational methodologies to represent knowledge into graph model with inference capability such as Bayesian network, fuzzy concept network and fuzzy cognitive map [3,4,5]. Above all, Bayesian network has been used practically to infer goals of software users in Microsoft Excel [6].

T.D. Gedeon and L.C.C. Fung (Eds.): AI 2003, LNAI 2903, pp. 979-991, 2003.
© Springer-Verlag Berlin Heidelberg 2003

In previous work [7], we proposed a method to combine behavior modules evolved on CAM-Brain [8] using behavior network. In this paper, we attempt to apply Bayesian network to represent prior knowledge about goal activation in behavior network and infer posterior probability with observed variables. Bayesian network estimates the importance of goals with some observed sensor data. Structure of Bayesian network can be learned from the data that are collected with some different approaches such as conditional independence tests [9], scoring-based optimization [10] and hybrid of the two approaches [11]. In this paper, scoring-based method is adopted to construct Bayesian network because manual construction of network needs much effort and is impossible with observed raw sensor data.

In general, there are three kinds of agent architectures: deliberation, reactive agent and hybrid architectures. In imitation of the cognitive architecture of the human's brain, a new type of agent architecture is proposed and a robot soccer team underlying this framework is established [12]. To model complex systems, software agents need to combine cognitive abilities to reason about complex situations, and reactive abilities to meet hard deadlines. Guessoum proposed an operational hybrid agent model which mixes well known paradigms (objects, actors, production rules and ATN) and real-time performances [13]. There are some works related to hybrid agent architectures like combination of reactive control and deliberative planning [14,15,16].

Experiments are conducted on Khepera mobile robot simulator with four basic behaviors. Bayesian network structure is learned using the data that are collected from the experimental environments. Experimental results show that the proposed method of behavior network with Bayesian network is promising.

2 Behavior Network

Competition of behaviors is the basic characteristics of behavior network. Each behavior gets higher activation level than other behaviors from forward and backward activation spreading. 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. Precondition is the sensor that is likely to be true when the behavior is executed. Add list is a set of conditions that are likely to be true by the execution of behavior and delete list is a set of conditions that are likely to be false by the execution of behavior. Figure 1 is a typical example of behavior network.

Forward propagation: Activation a is updated as the value added by environmental sensors that are precondition of the behavior. n means the number of sensors, and a_s is the activation level of the sensor.

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

$$f(a_{s_i}) = \begin{cases} \phi \times a_{s_i} & s_i \in \text{precondition} \\ 0 & s_i \notin \text{precondition} \end{cases} \tag{2}$$

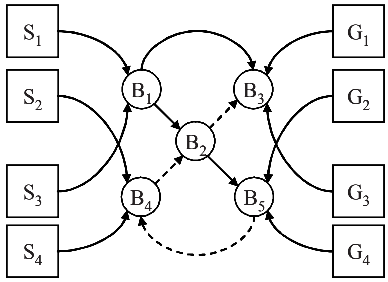


Fig. 1. An example of behavior network (S: sensors, B: behavior, G: goal). Solid line among behaviors represents predecessor link and dashed line represents successor link

Backward propagation: Activation a is updated as the value by goals that are directly connected to the behavior. If the execution of the behavior is desirable for the goal, positive goal-behavior link get active between them. Otherwise, negative goal-behavior link get active between them. n means the number of goals, and a_g is the activation level of the goal.

$$\Delta a_2 = \sum_{i=1}^{n} f(a_{g_i}) \tag{3}$$

$$f(a_{g_i}) = \begin{cases} \gamma \times a_{g_i} & g_i \in \text{positive link} \\ 0 & g_i \notin \text{negative link} \end{cases} \tag{4}$$

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

$$\Delta a_3 = \sum_{i=1}^{n} f(a_{b_i}) \tag{5}$$

$$f(a_{b_i}) = \begin{cases} a_{b_i} & \text{predecessor link from } b_i \\ \phi / \gamma \times a_{b_i} & \text{successor link from } b_i \\ -\delta / \gamma \times a_{b_i} & \text{conflictor link from } b_i \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

Finally, the activation of a is updated as follows.

$$a' = a + \Delta a_1 + \Delta a_2 + \Delta a_3 \tag{7}$$

If the activation level a' is larger than threshold θ and the precondition of the behavior is true, the behavior becomes a candidate to be selected. Among candidate behaviors, the highest activation behavior is chosen. Threshold θ is reduced by 10% and the activation update procedure is performed until there are candidate behaviors.

3 Behavior Selection with Bayesian Network

Behavior-based approach is to realize intelligence without any explicit representation. This property makes robot react immediately to unexpected situation such as "navigation on unknown planet." Robot does not have complex internal representation to process input signal. For higher behaviors, it is desirable to combine the reactive behaviors using some extra mechanisms like ASM.

Adding learning, inference and planning capabilities can improve ASM. Designing ASM is not an easy task because there are many variables to consider and knowledge about environment is not enough. Learning algorithm can determine the structure of ASM automatically and change a part of structure adaptively to the environments. Inference module uses computational model such as Bayesian network, fuzzy concept network and fuzzy cognitive map to represent prior knowledge and estimates unknown variables. ASM is not adequate to insert knowledge for inference and cannot select behaviors properly when the problem contains uncertainty. Planning optimizes the sequence of behaviors for solving the task.

In this paper, we focus on inference mechanism for ASM. Bayesian network is used to infer goals of behavior network that has one or more goals. Robot has several sensors that are represented as real value. Behavior-based robot uses only the sensor information to infer the status of environment. It is impossible to construct Bayesian network manually using the information of sensors. Therefore, Bayesian network structure is determined by learning from data that are collected from the environment.

3.1 Bayesian Network

Consider a domain U of n discrete variables $x_1,\dots,x_n$, where each x_i has a finite number of states. A Bayesian network for U represents a joint probability distribution over U by encoding (1) assertions of conditional independence and (2) a collection of probability distributions. Specifically, a Bayesian network B can be selected as the pair (B_s, Θ), where B_s is the structure of the network, and Θ is a set of parameters that encode local probability distributions [17]. Figure 2 shows an example of Bayesian network that has six variables.

The joint probability for any desired assignment of values $<y_1, \dots, y_n>$ to the tuple of network variables $<Y_1, \dots, Y_n>$ can be computed by the following equation:

$$P(y_1,\dots,y_n) = \prod_{i=1}^{n} P(y_i \mid Parents(Y_i)) \tag{8}$$

where $Parents(Y_i)$ denotes the set of immediate predecessors of Y_i in the network.

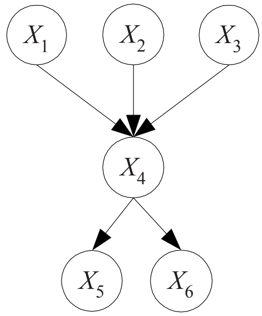


Fig. 2. An example of Bayesian network

3.2 Bayesian Network Learning

In this paper, we focus on learning structure of Bayesian network from data. The problem of learning a Bayesian network can be informally stated as: Given a training set $D=\{u_1,\ldots,u_N\}$ of instances of U, find a network B that best matches D. This optimization process is implemented in practice by using heuristic search techniques to find the best candidate over the space of possible networks. Scoring function assesses each Bayesian network using Bayesian formalism or minimum description length (MDL). Algorithm B based on greedy search is a representative method to find the structure of Bayesian network using scoring function [10]. Algorithm B is as follows.

Step 1: Initialize the network with no arc.
Step 2: Select arc($A\rightarrow B$) that has maximum increase of scoring function when the arc is inserted.
Step 3: Insert $A\rightarrow B$ into directed acyclic graph (DAG).
Step 4: Detect and remove cycles and remove it.
Step 5: Repeat 2-4 until there is no improvement or no arc inserted.

Usually, each Bayesian network is represented as a matrix and each element of the matrix represents link between nodes. Detection of cycle is easily conducted by checking links between ancestors and descendants of a node.

3.3 Behavior Network with Inference

Behavior network has one or more goals to be achieved in the environments. Coordination of their activation cannot be fixed in design stage of behavior network because there is uncertainty. Bayesian network is adopted to infer activation of goal from some observations from the environments. The structure of Bayesian network is automatically learned from the data that are collected from wandering. Observed variables are sensor information that can be collected from the robot sensors including distance, light and velocity. From these data, it is possible to estimate unknown variables including area information, emergency level and cooperation with other agents. Equation (3) is modified as follows:

$$\Delta a_2 = \sum_{i=1}^{n} \sum_{j=1}^{m} f(a_{g_i}) \times r_{i,j} \times P(v_j \mid observation) \tag{9}$$

where m is the number of unknown variables, and $r_{i,j}$ is the relevance of goal i with respect to variable j. This value is determined manually. $P(v_j \mid observation)$ is calculated using Bayesian network and equation (8).

4 Experimental Results

Khepera was originally designed for research and education in the framework of a Swiss Research Priority Program (see Figure 3). It allows confrontation to the real world of algorithms developed in simulation for trajectory execution, obstacle avoidance, pre-processing of sensory information, and hypothesis test on behavior processing. Khepera robot has two wheels. Eight infrared proximity sensors are placed around the robot.

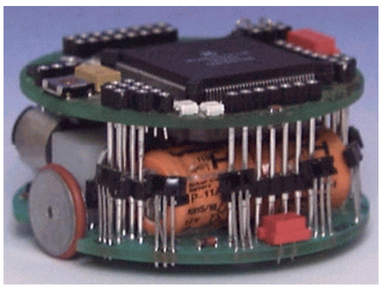
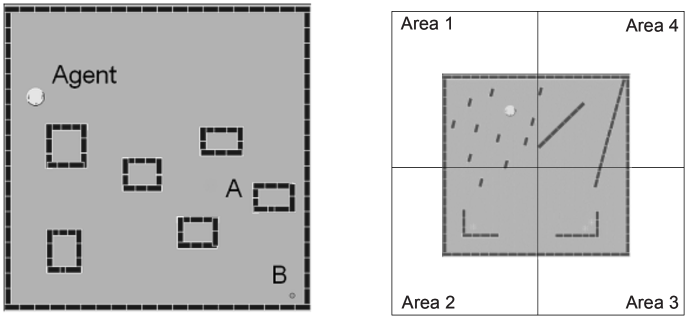


Fig. 3. Mobile robot, Khepera

Two different experimental environments are used (Figure 4). Environment-I (E-I) is designed to test the proposed method with manually constructed Bayesian network. Environment-II (E-II) is for automatically constructed Bayesian network. In E-I, there are two points (A, B) where robot must pass. Light source is positioned in A. Robot can detect position A with the level of light sensor. Problem of E-I is to detect robot's unexpected stop and make appropriate change of goal. In the design stage of behavior network for E-I, it is difficult to incorporate of how to manage this situation. Problem of E-II is different with one of E-I. Robot must pass light sources in Area 2 and Area 3.



(a) (b)

Fig. 4. Two different experimental environments. (a) Environment-I (b) Environment-II

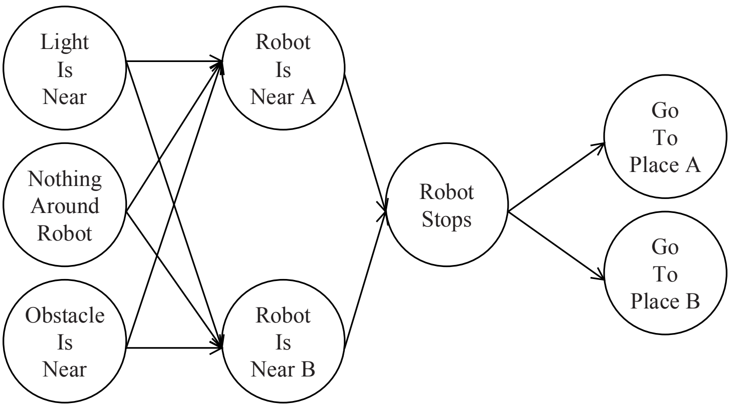
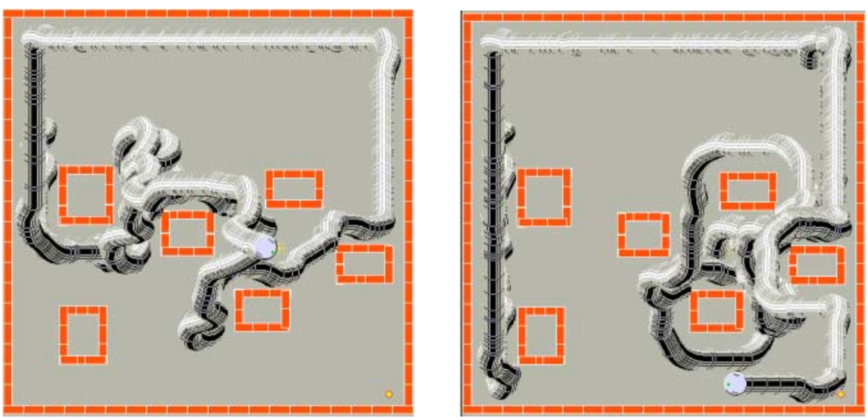


Fig. 5. Bayesian network that is manually constructed



(a) (b)

Fig. 6. Comparison with behavior network and the proposed method. (a) behavior network (b) the proposed method

4.1 Experiment I

Prior knowledge of this environment I is that robot frequently stops because it considers light source as obstacle. Also robot stops in position B by the small obstacle that is not easy to detect. Goal of agent is to go A and B sequentially (A and B are noted in Figure 4-(a)). Bayesian network is designed manually in a very simple form. The position of robot is estimated from the sensor information including light sensors and distance sensors. Figure 5 shows this Bayesian network and figure 6 shows a comparison with behavior network and the proposed method. The proposed method shows that robot can pass two positions without stopping but the behavior network without Bayesian network makes the robot stop at A.

4.2 Experiment II

There are four different areas in E-II (Area1, Area2, Area3, and Area4). Area1 is a start position that has many small obstacles. Area2 contains two light sources and Area3 contains one light source. Area 4 has simple obstacles. If robot can recognize area using observed information, behavior network can generate more appropriate behavior sequences. Robot uses three behaviors evolved on CAM-Brain [7]. They are "Avoiding obstacles," "Going straight," and "Following light."

Following light: The robot goes to stronger light. It must operate this module to go to the light source.
Avoiding obstacle: If there is an obstacle around the robot, it avoids obstacles without bumping against it.
Going straight: If there is nothing around the robot, it goes ahead. This module makes it to move continuously without stopping.

Figure 7 shows the behavior network for experiment II. There are two different links among behaviors. Predecessor link is represented with solid line and successor link is represented with dashed line. There are five different environmental sensors that can be inferred from original raw sensor data. They are defined as follows.

Obstacle is near: If the distance sensor is less than 1000, it becomes true.
Nothing around robot: If the distance sensor is less than 1000, it becomes true.
Light level I: If the light sensor is less than 400, it becomes true.
Light level II: If $400 <$ the light sensor < 450, it becomes true.
No light source: If the light sensor is larger than 450, it becomes true.

Light sensor has the value ranged from 50 to 500 and 50 means that light source is near. Distance sensor has the value ranged from 0 to 1024 where 1024 means that the obstacle is very near.

Three goals are defined as follows.

Minimizing bumping A: If the distance sensor is larger than 1020, it becomes true.
Minimizing bumping B: If the distance sensor is less than 800, it becomes true.
Going to light source: If the light sensor is larger than 450, it becomes true.

If the robot is in Area1, the value of distance sensor is large and needs to avoid bumping. If the robot is in Area4, the value of distance sensor is small and needs to go straight. If the light sensor has lower value, robot will propagate signals to “Following light.”

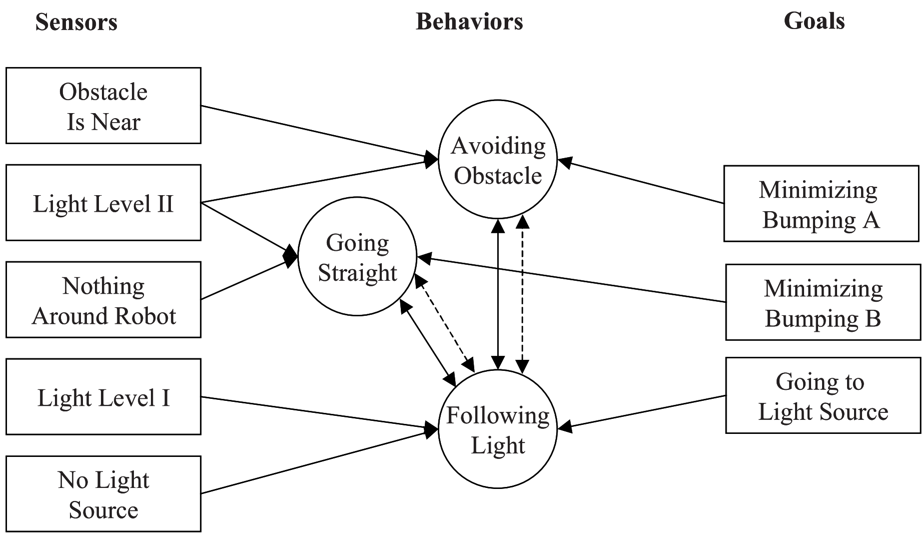
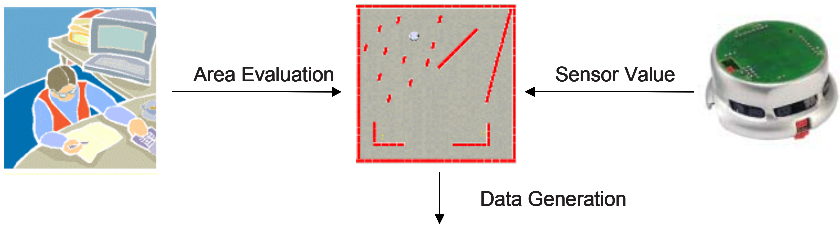


Fig. 7. Behavior network for experiment II



Distance1	Distance 2	Distance 3	Light 1	Light 2	Area
500.0	400.0	1022.0	499.0	200.0	1
498.0	300.0	1020.0	450.0	220.0	2

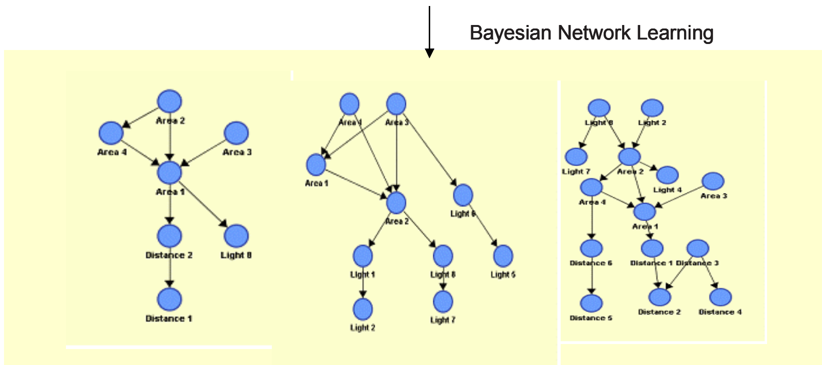


Fig. 8. Data collection for Bayesian network learning

Figure 8 shows the procedure of collecting data from the experimental environments. Robot moves randomly and user evaluates robot's position as one of “Area1,” “Area2,” “Area3,” and “Area4.” Khepera robot has 8 distance sensors and 8 light sensors of real value. Training data for Bayesian network have 20 variables (16 sensor values and 4 area information). If robot is in Area1, the variable for Area1 is 1.0 and other area variables are 0.0.

Bayesian network is learned using algorithm the *B* based on Tabu search [10]. Figure 9 shows the structure of Bayesian network that determines the position of robot. Bayesian network consists of 14 nodes (four nodes are area variables and ten nodes are related with distance and light sensors). We expected that light information was very important because light source can be used as criterion to classify bottom area with top area. Area2 and Area3 can be classified with the strength of light because Area2 has two light sources but area3 has one light source. The learned Bayesian network represents the information well. Two light sensor nodes directly link to Area2.

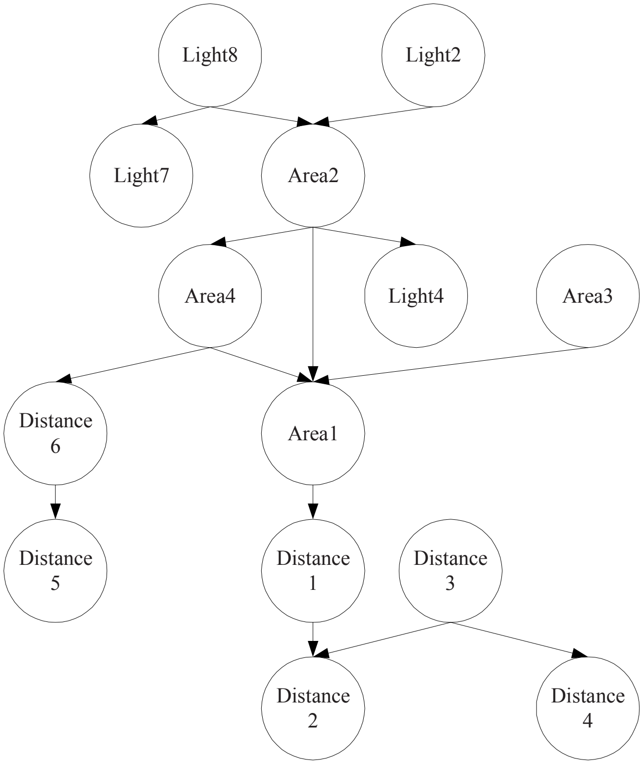
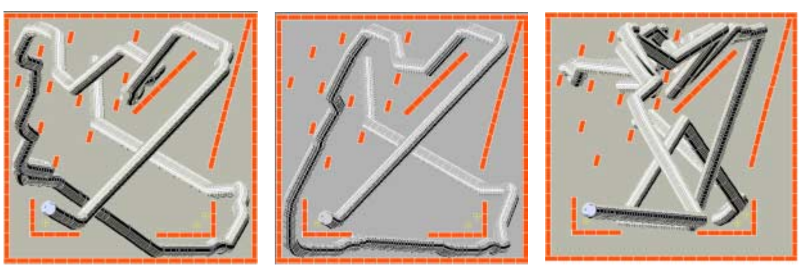


Fig. 9. Bayesian network that are learned from the collected data

Figure 10 shows experimental results in E-II. In (a,b), robot navigates the area with behavior sequence that is determined using the behavior network. The network selects one behavior at a time and executes it. In (c), robot navigates the area with the combination of behavior network and Bayesian network learned. Bayesian network determines the conditional probability of area1, area2, area3, and area4 with observed sensors. Unlike the robots in (a) and (b), robot passes light source in (c). Goal of robot is to visit four areas without exception and if there is a light source, it must go to pass the source.



(a) (b) (c)

Fig. 10. Experimental results with Bayesian network learned. (a,b) Behavior network without Bayesian network (c) The proposed method

5 Conclusions

In this paper, we have proposed a method for improving behavior network with Bayesian network that is learned from data. Behavior network can implicitly insert goals into the model but has no mechanism to coordinate goals without conflict. In this reason, some computational methodology to support inference with prior knowledge is needed. There are many inference models but Bayesian network is widely used and can provide sound mathematical foundations for inference and learning. To reduce user's difficulty in design of Bayesian network, the learning from data is adopted. Fuzzy concept network and fuzzy cognitive map can be candidates for inference of behavior network but there is no learning algorithm of structure for them. The different experiments show that the proposed method can perform better than behavior network without Bayesian network. As a future work, we will apply the model to more complex and realistic problems like avoiding movable obstacle.

Acknowledgement

This research was supported by Biometrics Engineering Research Center, and Brain Science and Engineering Research Program sponsored by Korean Ministry of Science and Technology.

References

[1] Pirjanian, P.: Behavior coordination mechanisms-state-of-the-art. Tech-report IRIS-99-375, Institute for Robotics and Intelligent Systems, School of University of Southern California, Oct (1999)

[2] Maes, P.: How to do the right thing. Connection Science, vol. 1, no. 3 (1989) 291-323

[3] Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann Publishers (1998)

[4] Chen, S.-M. and Wang, J.-Y.: Document retrieval using knowledge-based fuzzy information retrieval techniques. IEEE Transactions on Systems, Man and Cybernetics, vol. 25, no. 5 (1995) 793-803

[5] Stylios, C.D. and Groumpos, P.P.: Fuzzy cognitive maps: A model for intelligent supervisory control systems. Computers in Industry, vol. 39, no. 3 (1999) 229-238

[6] Horvitz, E., Breese, J., Heckerman, D., Hovel, D. and Rommelse, K.: The lumiere project: Bayesian user modeling for inferring the goals and needs of software users. Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (1998) 256-265
[7] Kim, K.J. and Cho, S.B.: Dynamic selection of evolved neural controllers for higher behaviors of mobile robot. IEEE International Symposium on Computational Intelligence in Robotics Automation, Banff, Canada (2001) 467-472
[8] Garis, H. de and Korkin, M.: The CAM-Brain Machine (CBM): An FPGA based tool for evolving a 75 million neuron artificial brain to control a lifesized kitten robot. Journal of Autonomous Robots, vol. 10, no. 3 (2001) 236-249
[9] Pearl, J. and Verma, T. S.: Equivalence and synthesis of causal models. Proceedings of the 6th Conference on Uncertainty in Artificial Intelligence (1990) 220-227
[10] Campos, L. M. de, Fernandez-Luna, J.M., Gamez, J. A. and Puerta, J. M.: Ant colony optimization for learning Bayesian networks. International Journal of Approximate Reasoning, vol. 31 (2002) 291-311
[11] Singh, M. and Valtorta, M.: Construction of Bayesian network structures from data: A brief survey and an efficient algorithm. International Journal of Approximate Reasoning, vol. 12 (1995) 111-131
[12] Shi, L., Zhen, Y. and Zengqi-Sun.: A new agent architecture for RoboCup tournament: Cognitive architecture. Proceedings of the 3rd World Congress on Intelligent Control and Automation, vol. 1 (2000) 199-202
[13] Guessoum, Z.: A hybrid agent model: A reactive and cognitive behavior. The International Symposium on Autonomous Decentralized Systems (1997) 25-32
[14] Lyons, D.M., and Hendriks, A.J.: Planning as incremental adaptation of a reactive system. Robotics and Autonomous Systems, vol. 14, no. 4 (1995) 255-288
[15] Murphy, R. R., Hughes, K., Marzilli, A., and Noll, E.: Integrating explicit path planning with reactive control of mobile robots using Trulla. Robotics and Autonomous Systems, vol. 27, no. 4 (1999) 225-245
[16] Aguirre, E., and Gonzalez, A.: Fuzzy behaviors for mobile robot navigation: Design, coordination and fusion. International Journal of Approximate Reasoning, vol. 25, no. 3 (2000) 255-289
[17] Chickering, D. M., Heckerman, D. and Meek, C.: A Bayesian approach to learning Bayesian networks with local structure. Microsoft Technical Report MSR-TR-97-7, 1997.