Object tracking algorithm of Swarm Robot System for using Polygon based Q-learning and parallel SVM

Snag-Wook Seo, Hyun-Chang Yang and Kwee-Bo Sim*

School of Electrical and Electronic Engineering, Chung-Ang University
221, Heukseok-Dong, Dongjak-Gu, Seoul, 156-756, Korea
Tel : +82-2-820-5319, Fax : +82-2-817-0553, E-mail : kbsim@cau.ac.kr

Abstract

This paper presents the polygon-based Q-leaning and Parallel SVM algorithm for object search with multiple robots. We organized an experimental environment with one hundred mobile robots, two hundred obstacles, and ten objects. Then we sent the robots to a hallway, where some obstacles were lying about, to search for a hidden object. In experiment, we used four different control methods: a random search, a fusion model with Distance-based action making (DBAM) and Area-based action making (ABAM) process to determine the next action of the robots, and hexagon-based Q-learning, and dodecagon-based Q-learning and parallel SVM algorithm to enhance the fusion model with Distance-based action making (DBAM) and Area-based action making (ABAM) process. In this paper, the result show that dodecagon-based Q-learning and parallel SVM algorithm is better than the other algorithm to tracking for object.

Key words : DBAM, ABAM , Parallel SVM, Polygon, Q-learning

1. Introduction

Nowadays, robots are performing human's work in dangerous field, such as rescue jobs at fire-destroyed building or at gas contaminated sites; information retrieval from deep seas or from space; and weather analysis at extremely cold areas like Antarctica. Sometimes, multiple robots are especially needed to penetrate into hard-to-access areas, such as underground insect nests, to collect more reliable and solid data

Multiple robot control has received much attention since it offers a new flexible and vigorous way to control multiple agents. For instance, Parker used the heuristic approach algorithm for multiple robots and applied it to cleaning tasks [1]. Ogasawara employed distributed autonomous robotic systems to control multiple robots transporting a large object [2]. However, the greater the dependency on communication in a system is, the more difficult a system hierarchy becomes. Therefore, this study proposes a fusion model with distance-based action making (DBAM) and area-based action making (ABAM) process for instinctive intelligence similar to bee behavior in an apiary. This in turn, is incorporated with Dodecagon-based Q-learning and parallel SVM, which is learned intelligence and helps multiple robots to navigate, avoid collision, and search using their own trajectories.

Reinforcement learning through exploring its environment actively enables an agent to determine what the following action should be. During the exploration of an uncertain state space followed with a reward, the agent learns what to do by continuum of its state history and appropriate propagation of rewards through the state space [3]. This research focused on Q-learning as a reinforcement learning technique because Q-learning is a simple way to solve Markovian action problems with incomplete information. In addition, an agent can map state-action pairs onto expected returns based on the action-value function Q [4]. In addition to this simplicity, Q-learning can be adapted to the real world. For example, state space can be harmonized with the physical space of the real world. An action can be regarded as a physical robot maneuver. This paper proposes that the Dodecagon-based Q-learning and parallel SVM can enhance fusion model with distance-based action making (DBAM) and area-based action making (ABAM) process so that the learning process can be better adapted to real world situations.

The organization of this paper is as follows. Section 2 introduces an action making process. Section 3 presents Dodecagon-based Q-learning and parallel SVM adaptation. Experimental results from the application of four different searching methods to find a target object are presented in Section 4. Section 5 presents conclusions are.

Manuscript received Jul. 30. 2008; revised Sep. 5. 2008.
* Corresponding author
This research was supported by the Development of Social Secure Robot using Group Technologies of Growth Dynamics Technology Development Project by Ministry of Knowledge Economy, Republic of Korea..

2. Action Making Process

2.1 Two action making process

Both Distance-based action making (DBAM) and Area-based action making (ABAM) process are widely used for determining next action of a robot. In the DBAM process is referred to as DBAM, a robot can recognize its surroundings by the distance between itself and an obstacle. But, in the case of ABAM, a robot uses the circumferential areas for recognizing its surroundings. The key to the ABAM process is that it removes uncertainty regarding its surroundings. It is similar to the behavior-based direction change in regards to controlling robots [5][6]. Under the ABAM process robots recognize the shape of their surroundings and then take action, i.e., turn and move toward the widest guaranteed space. Figure 1 depicts the different actions in the same situation under DBAM and ABAM, respectively [7][8]. As you can infer by their name, DBAM process selects d4 that is the direction of the longest distance from the robot. Otherwise, ABAM process selects a4 that has the widest area on the neighborhood.

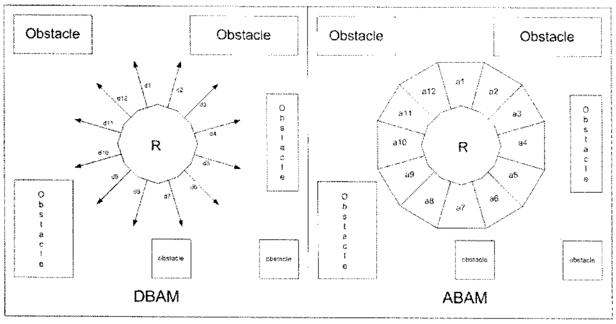


Fig. 1 Fusion model with DBAM and ABAM

2.2 Area based action making process

Area-based action making (ABAM) process is a process that determines the next action of a robot. The reason why this process is referred to ABAM is that a robot recognizes surrounding not by distances, from itself to obstacle, but by areas around itself. The key idea of the ABAM process is to reduce the uncertainty of its surrounding. It is similar with the behavior-based direction change, to control the robots. The robots recognize the shape of its surrounding, and then take an action (turn and move forward) to where the widest space will be guaranteed. Consequently, each robot can avoid an obstacle and collision with other robots. Our mobile robot has the twelve sonar sensor pairs, which are placed at an angle of 30 degrees with one another to cover 360 degrees.

The advantage of ABAM and DBAM are illustrated by the following example. Figure 2 presents the result of each action making process by DBAM and ABAM. In both case, the robot is surrounded by 4-obstacles. By DBAM, the robot will be confused because it perceives that there is no obstacle in the southeast direction, and then it will try to keep tracking to the southeast. Finally, it will get stuck between two obstacles. By ABAM, however, the robot will calculate the areas of its surrounding, and then it will recognize that an action to the northeast will guarantee the widest space. Therefore, the robot will change its direction to the northeast.

In addition to the obstacle avoidance, ABAM also make the robots to search their own space [9]. This feature is advantageous when 2 or 3 robots meet at the same place. When they face each other, each robot will try to find more wide space. Consequently, the robot will change its direction to avoid the other robots and start to search in its own space again

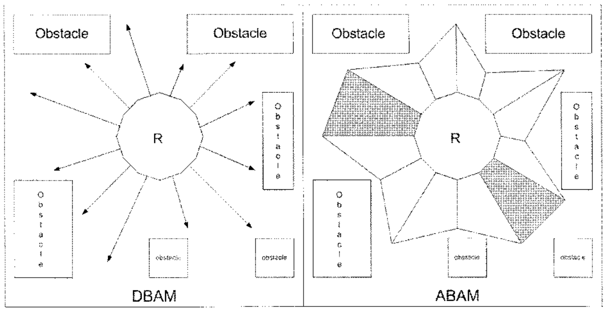


Fig. 2 Illustrative example of robot maneuvers by DBAM and ABAM

2.3 Fusion model with DBAM and ABAM

In this paper we use model that fused into DBAM and ABAM for action making process. DBAM process selects distance that is the direction of the longest distance from the robot and ABAM process selects area that has the widest area on the neighborhood.

DBAM process that considers only distance to incorrectly action. Otherwise ABAM process that considers only area to selects action, has an advantage to increase probability that can select correctly action but disadvantage to increase calculation quantity.

Fusion model with DBAM and ABAM process selects action that distance is longest and area is widest. In this paper we select method that widest area given weight, among others selects action that distance is longest from obstacle. Figure 4 presents the result of action making process by fusion model with DBAM and ABAM.

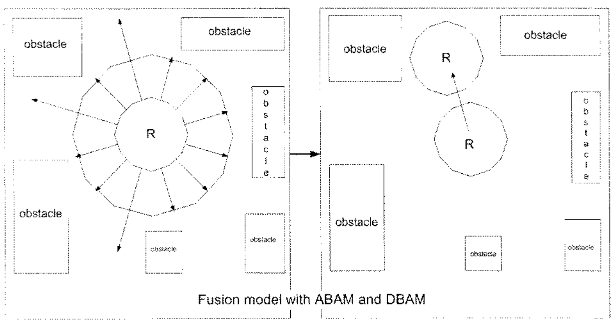


Fig. 3 Illustrative example of robot maneuvers by fusion model with ABAM and DBAM

3. Polygon-Based Q-Learning and SVM

3.1 Q-learning algorithm

Q-learning is a well-known algorithm for reinforcement learning. It leads an agent to acquire optimal control strategies from delayed rewards, even when there is no prior knowledge of the effects of its actions on the environment [10][11]. The Q-learning for our robot system was adapted to enhance the ABAM process. The adaptation can be performed with a simple and easy modification.

The Q-learning algorithm presented in Table 1, where s is a possible state, a is a possible action, r indicates the immediate reward value, and γ is the discount factor. The formula to update the table entry value is:

Table 1. Q-Learning Algorithm

For each s,a initialize the table entry $\hat{Q}(s,a)$ zero
Observe the current state s
Continue to infinity

- Select the action a and execute it
- Receive the immediate reward r
- Observe the new state s'
- Select the action
- Update the table entry for $\hat{Q}(s,a)$

$$\hat{Q}(s,a) \leftarrow r + \gamma \max_{s'} \hat{Q}(s',a) \quad (1)$$

- $s \leftarrow s'$

3.2 Hexagon-based Q-learning

The unique Q-learning type for this robot system was adapted to enhance the ABAM process. The adaptation can be performed with a simple and easy modification, namely, through hexagon-based Q-learning. The well-known standard Q-learning method is based on square state space. But hexagon-based Q-learning uses the different shape of the state space from the ordinary square-based state space.

The reason for changing the shape of state space from a square to a hexagon was that the hexagon is a polygon that can be expanded infinitely by its combination. According to this adaptation, the robot could perform an action in 6-directions and have 6-table entry values. Moreover, the hexagon-based Q-learning has extra advantages that it has fast responses and many radius of action.

Now, if the robot decides that +60 degrees guarantees the widest space after calculating its 6-areas of surroundings, the action of the robot would be. After the action is taken, if Area6' is the widest area, the value of $\hat{Q}(s_1, a_{+60})$ can be updated using formula (1) as

$$\begin{aligned} \hat{Q}(s_1, a_{right}) &\leftarrow r + \gamma \max_{a_2} \hat{Q}(s_2, a'_{\theta}) \\ &\leftarrow 0 + 0.9 \max_{a_2} \{Area1', \cdots, Area6'\} \quad (2) \\ &\leftarrow \gamma Area6' \end{aligned}$$

Where $r = 0$, $\gamma = 0.9$ are predetermined of values.

3.3 parallel SVM

In the structure shown in figure 4, the data are split into subsets and each one is evaluated individually for support vectors in the first layer. The results are combined two-by-two and entered as training sets for the next layer. The resulting support vectors are tested for global convergence by feeding the result of the last layer into the first layer, together with the non-support vectors. The advantage is that every SVM did not have to deal with the whole training dataset, and these multiple SVM classifiers can be trained in distribute computer network, so the training process is speeded up greatly [12]. Often, this cascade structure produces satisfactory accuracy with a single pass through, but if the global optimum has to be reached, the result of the last layer should be fed back into the first layer. Therefore, we should consider when a feed back is needed and how to collect support vectors efficiently.

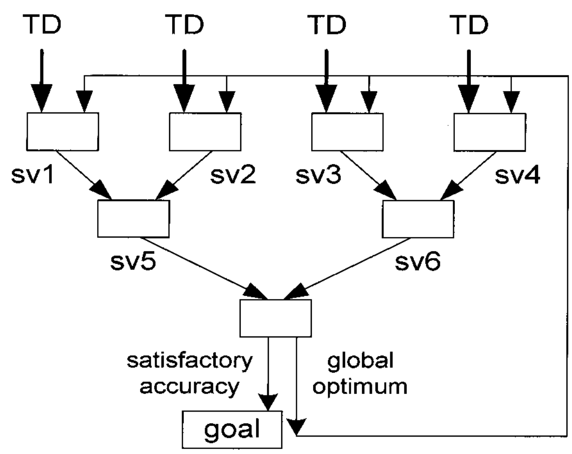


Fig. 4 Structure of cascade SVM

3.4 Dodecagon-based Q-learning and parallel SVM

Dodecagon-based Q-learning and parallel SVM algorithm are measures obstacle with 12 rectangular directions using 12 sonar sensors. And then the whole width that robot can recognize divided 12 areas. We maximize to each area that taken out obstacle area and distance from robot to obstacle, and set up Q-table about the next state.

The difference point of Q-learning algorithm is update method. Q-learning algorithm is returned former state when present Q-value is minimized after acts to compare with former Q-value. But proposed dodecagon based Q-learning and SVM algorithm is not returned former Q-value. Because Dodecagon based Q-learning and parallel SVM algorithm decide optimum state to make recognize state and select action using parallel SVM. In this paper, proposed algorithm is

$$\hat{Q}(s,a) \leftarrow r + \gamma \max_{s'} \hat{Q}(s',a)$$
$$\leftarrow r + \gamma \max_{s'} \hat{Q}\{(s_{Area1} - s_{obstacle})l_{obstacle}, \ldots, (s_{Area12} - s_{obstacle})l_{obstacle}\} \quad (3)$$

In this formula, $\hat{Q}(s_{AreaN} - s_{obstacle})$ be classified by SVM classifier.

$$\underset{b \in R}{\text{maximize}} \min\{\|S_{AreaN} - S_{obstacle}\| < S_{AreaN}, X > +b, i = 1, \ldots, n\} \quad (4)$$

In this paper, we use to SVM classifier for satisfactory accuracy result. This result is that we want to robot selects the best space to movement. And parallel SVM is better than the SVM classifier.

According to learning used to dodecagon based Q-learning and parallel SVM algorithm is right algorithm to autonomous movement robot system.

4. Experiment Results

In this paper we experiment on four algorithms to verify for proposed algorithm. We organized an experimental environment with one hundred mobile robots, two hundreds obstacles, and one object. The number of search time is two hundreds. First, we used the random search control method to find the hidden object. As it may increase iterations, random search method is not rule. And for characteristics of random search, it is not represent little to statistics. Second, we applied fusion model with DBAM and ABAM to the robots. This method that forty robots find to object on the average reinforce to ability of search. Third, we applied the hexagon-based Q-learning to fusion model with DBAM and ABAM as a modified control method. This method that fifty-five robots find to object on the average reinforce to fusion model with DBAM and ABAM. And this result shows that learning is important part. Finally, we adopted the dodecagon-based Q-learning and parallel SVM algorithm to robot. The results of our experiment are presented in figure 5,6,7,8.

The result show that dodecagon-based Q-learning and parallel SVM algorithm is better than the other algorithm to tracking for object.

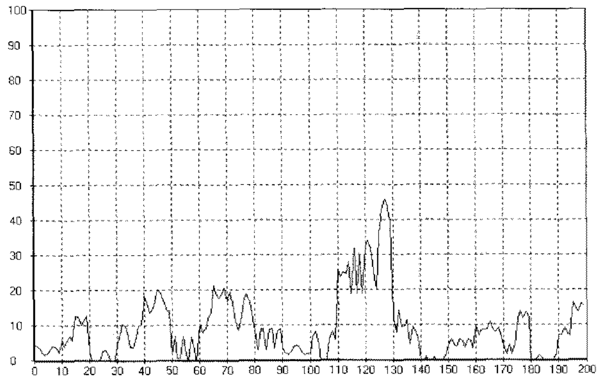


Fig. 5 Random Search

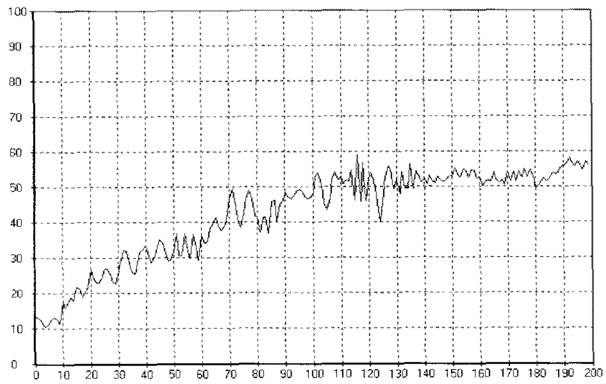


Fig. 6 Fusion model with DBAM and ABAM

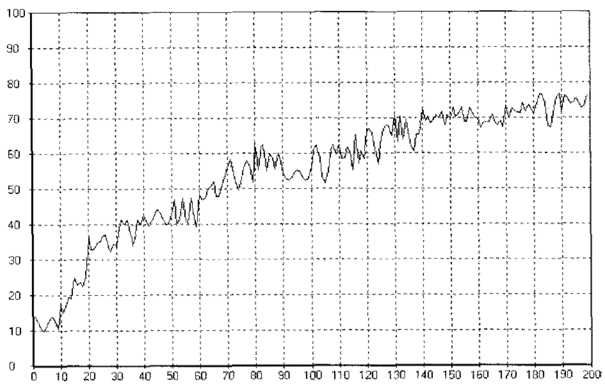


Fig. 7 Hexagon-based Q-learning

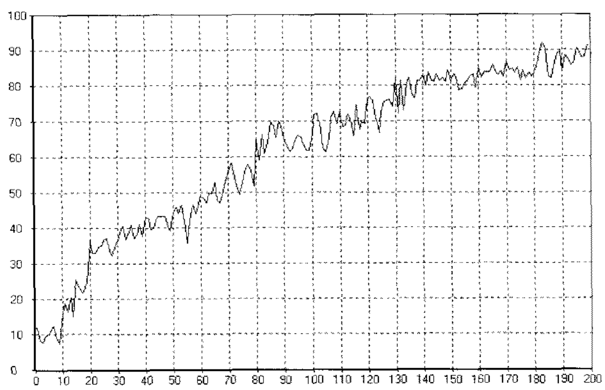


Fig. 8 Dodecagon-based Q-learning and cascade SVM

5. Conclusion and Future Works

In this paper, we presented dodecagon-based Q-learning and parallel SVM algorithm, hidden in unknown space, for one hundred mobile robots. The experimental results from the application of the four different control methods in the same environmental situations were presented. The four different control method: a random search, a fusion model with Distance-based action making (DBAM) and Area-based action making (ABAM) process to determine the next action of the robots, and hexagon-based Q-learning, and dodecagon-based Q-learning and parallel SVM algorithm to enhance the fusion model with Distance-based action making (DBAM) and Area-based action making (ABAM) process.

In our research, first we need to clarify the problem of accessing to the object. This means that if multiple robots are to carry out a task such as object transporting or block stacking, the robots need to recognize the object then approach to it. Therefore, we need to develop the robust accessing algorithm. The second, as bigger to swarm robot system, need to communication method and data mining method between robots.

In the future we are expected to closely inspect for effectiveness through variety experiment, and we applied that swarm robot system of real environment.

References

[1] Parker, L, "Adaptive Action Selection for Cooperative Agent Teams," *Proc. Of 2nd Int. Conference on Simulation of Adaptive Behavior*, pp.442-450, 1992.

[2] Ogasawara. G, Omata. T, Sato. T, "Multiple Movers Using Distributed, Decision-Theoretic Control," *Proc. of Japan-USA Symposium on Flexible Automation*, pp. 623-630, 1992.

[3] Ballard. D, "An Introduction to Natural Computation," *The MIT Press Cambridge*, 1997.

[4] Jang. J, Sun. C, Mizutani. E, "Neuro-Fuzzy and Soft Computing," *Prentice-Hall New Jersey*, 1997.

[5] W. Ashley and T. Balch, "Value-based observation with robot teams using probabilistic techniques," *Proceeding of International conference on Advanced Robotics*, 2003

[6] J. B. Park, B. H. Lee, and M. S. Kim, "Remote control of a mobile robot using distance-based reflective force," *Proceeding of IEEE International Conference on Robotics and Automation*, vol. 3, pp. 3415-3420, 2003.

[7] D. Patterson and J. Hennessy, "Computer Organization and Design," *Morgan-Kaufmann 3rd edition*, 2005

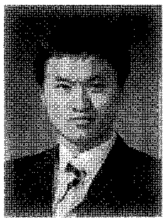
[8] T. Mitchell, "Machine Learning," *McGraw-Hill*, Singapore, 1997.

[9] Ogren. P, Leonard. N. E, "Obstacle Avoidance in Formation," *Proc. Of IEEE International Conference on Robotics and Automation*, pp. 2492-2497, 2003.

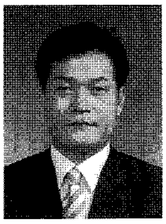
[10] C. Clausen and H. Wechsler, "Quad Q-learning," *IEEE Transaction on Neural Network*, Vol. 11, pp. 279-294, 2000.

[11] S. Russel and P. Norbig, Artificial Intelligence: A Modern Approach. *2nd edition Prentice-Hall*: New Jersey, 2003.

[12] Jian-Pei Zhang, Zhong-Wei Li and Jing Yang, "A Parallel SVM Training Algorithm on Large-Scale Classification Problems," *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics*, Vol. 3, pp. 1637-1641, 2005.



Sang-Wook Seo
He received his B.S degree in the Department of Electrical and Electronics Engineering from Chung-Ang University, Seoul, Korea, in 2007. He is currently Master course in the School of Electrical and Electronics Engineering from Chung-Ang University. His research interests include machine learning, multi agent robotic system, Evolutionary Computation, Evolutionary Robot, etc.



Hyun-Chang Yang
He received his M.S. degrees in the Department of Industrial Engineering from Soongsil University in 2002. He is currently Doctor course in the School of Electrical and Electronics Engineering from Chung-Ang University, Korea. His research interests include intelligent robot, smart home, and home network etc.

Kwee-Bo Sim
He received his B.S. and M.S. degrees in the Department of Electronic Engineering from Chung-Ang University, Korea, in 1984 and 1986 respectively, and Ph.D. degree in the Department of Electronics Engineering from The University of Tokyo, Japan, in 1990. Since 1991, he is currently a Professor. His research interests include artificial life, emotion recognition, ubiquitous intelligent robot, intelligent system, computational intelligence, intelligent home and home network, ubiquitous computing and Sense Network, adaptation and machine learning algorithms, neural network, fuzzy system, evolutionary computation, multi-agent and distributed autonomous robotic system, artificial immune system, evolvable hardware and embedded system etc. He is a member of IEEE, SICE, RSJ, KITE, KIEE, KIIS, and ICROS Fellow.