Goal-oriented Geometric Model Based Intelligent System Architecture for Adaptive Robotic Motion Generation in Dynamic Environment.

Dong-Hun Lee*, Kyung-Hun Hwang*, Chae-Wook Chung**, and Tae-Yong Kuc*

* School of Information and Communication Engineering SungKyunKwan University, 300 Chunchun-dong, Suwon, Korea
(Tel : +82-31-290-7202; E-mail: neowintop@ece.skku.ac.kr)
(Tel : +82-31-290-7202; E-mail: mrdkby@ece.skku.ac.kr)
(Tel : +82-31-290-7202; E-mail: tykuc@yurim.skku.ac.kr)
**Department of Electronic Communication Ansan College of Technology, 170 Choji-dong, Ansan, Korea
(Tel : +82-31-490-6064; E-mail: skechy@ieee.org)

Abstract: Control architecture of the action based robot engineering can be divided into two types of deliberate type - and reactive type- controller. Typical deliberate type, slow in reaction speed, is well suited for the realization of the higher intelligence with its capability to forecast on the basis of environmental model according to time flow, while reactive type is suitable for the lower intelligence as it fits to the realization of speedy reactive action by inputting the sensor without a complete environmental model. Looking at the environments in the application areas in which robots are actually used, we can see that they have been mostly covered by the uncertain and unknown dynamic changes depending on time and place, the previously known knowledge being existed though. It may cause, therefore, any deterioration of the robot performance as well as further happen such cases as the robots can not carry out their desired performances, when any one of these two types is solely engaged. Accordingly this paper aims at suggesting Goal-oriented Geometric Model(GGM) Based Intelligent System Architecture which leads the actions of the robots to perform their jobs under variously changing environment and applying the suggested system structure to the navigation issues of the robots. When the robots do perform navigation in human life changing in a various manner with time, they can appropriately respond to the changing environment by doing the action with the recognition of the state. Extending this concept to cover the highest hierarchy without sticking only to the actions of the robots can lead us to apply to the algorithm to perform various small jobs required for the carrying-out of a large main job.

Keywords: System Architecture, Navigation of Mobile Robot, Obstacle Avoidance, Intelligent System, Robot System

1. INTRODUCTION

With the advancement of robots and times goes-on, robots have expanded their roles in the support of human being. It is not possible adequately to respond to the external environment changed extensively by the time only with a simple control over the single assembly process robots which had been done in the past so as to carry out their roles in human society. Consequently this requires artificially intelligent control elements which can recognize, adapt to, learn and respond to external environment, and a lot of studies on fulfilling the requirements for human-type robots have been progressed.

Control architecture of the action based robot engineering can be divided into two types of deliberate type - and reactive type- controller. Typical deliberate type, slow in reaction speed, is well suited for the realization of the higher intelligence with its capability to forecast on the basis of environmental model according to time flow, while reactive type is suitable for the lower intelligence as it fits to the realization of speedy reactive action by inputting the sensor without a complete environmental model.

Looking at the environments in the application areas in which robots are actually used, we can see that they have been mostly covered by the uncertain and unknown dynamic changes depending on time and place, the previously known knowledge being existed though. It may cause, therefore, any deterioration of the robot performance as well as further happen such cases as the robots can not carry out their desired performances, when any one of these two types is solely engaged. Accordingly this paper aims at suggesting Goal-oriented Geometric Model(GGM) Based Intelligent System Architecture which leads the actions of the robots to perform their jobs under variously changing environment and applying the suggested system structure to the navigation issues of the robots.

The actions of robots can be broadly divided into navigation and manipulation, where we can regard the actions of robots as the matter of navigation as manipulation can be considered as one of the navigations the end-effect is addressing to the objects through avoiding the obstacles. In this regard, this paper, taking the navigation as the sample, suggests GGM Based Intelligent System Architecture and tries to convince its efficiency by actually putting it into movable robot system.

The suggested system structure represents the concept in way of modeling the relations between robots and environment, formulating it in the state set, constructing the actions in action set, linking properly these sets to the relations between them and inducing the actions of robots. Here in case modeling the environment, to describe the correlations in the form of sets between robot and environment is referred to as GGM. With this, we can make it possible to reduce the elements of environment models, to initiate the modeling of environment in a very simple way, and to generate the modeling for various environments

The state set and action set completed in this way can continue to generate the next state through the state transition which causes the robots to follow. Continuing these transitions make the robot perform the action following time in accordance with the time assigned and this finally will lead the robots to reach the goal.

When the robots do perform navigation in human life changing in a various manner with time, they can appropriately respond to the changing environment by doing the action with the recognition of the state. Extending this concept to cover the highest hierarchy without sticking only to the actions of the robots can lead us to apply to the algorithm to perform various small jobs required for the carrying-out of a large main job.

In Chapter 2 describes the definition of GGM, and the state set modeling the robot environment and the action set modeling the generation of the robot actions based on GGM in Chapter 3, followed by the relations between these two sets in Chapter 4, the cases where the algorithm is actually applied for the experiment in Chapter 5, the results from the experiment in Chapter 6 and in Chapter 7 conclusion.

2. GOAL-ORIENTED GEOMETRIC MODEL(GGM)

In the representation of the state of the robot(coordinates, angle, external environment), we draw the virtual line called Goal-oriented Geometric Model(GGM) from the robot to the goal and with this line show the neighboring state. Fig.2.1 shows the state of the robot on the absolute coordinates and the picture of Fig. 2.1 converted into GGM in Fig. 2.2

A lot of models is needed when representing the environment on the basis of absolute coordinates, but GGM can provide us a very simple modeling for the environment.

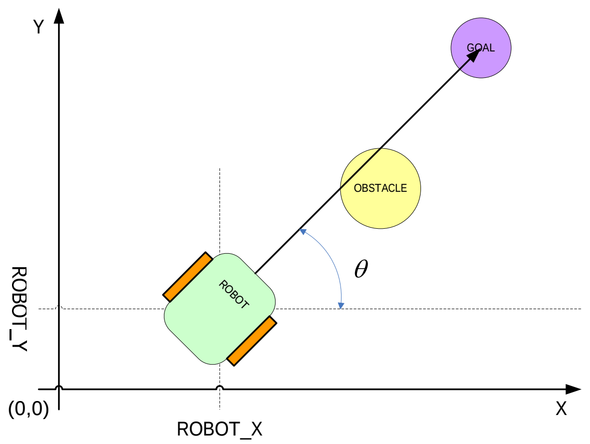


Fig.2.1) Relationship between Goal and Obstacle of Robot on Absolute Coordinates

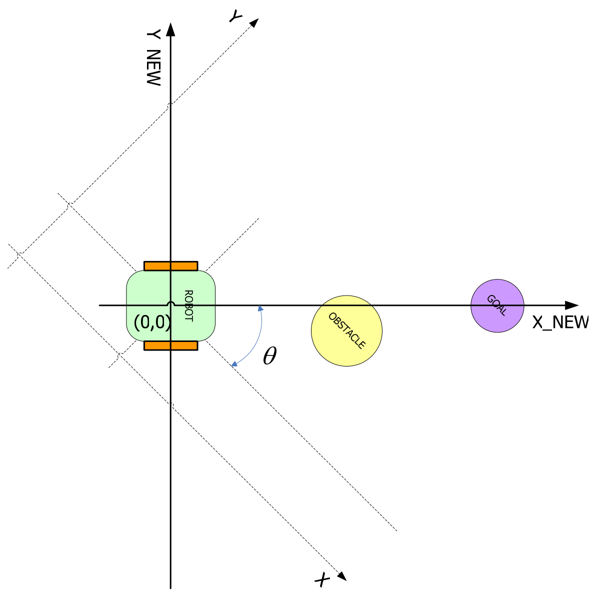


Fig.2.2) Relationship between Goal and Obstacle of Robot on GGM

3. STATE-SET & ACTION-SET

3.1 State Set

When current coordinates of robot and the coordinates of goal are determined, the relationship between robot and obstacle can be shown by coordinates transformation on the definition of GGM.

State set can be defined as $S = \{s_1, s_2, s_3, ..., s_n\}$ and the number of elements(n) depends on how specifically to do modeling the environment. This paper construct the set $S = \{s_1, s_2, s_3, ..., s_n\}$ as shown in Fig. 3.1

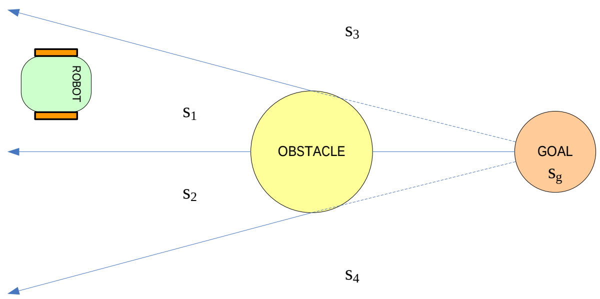


Fig. 3.1 State Modeling

The process the robot recognize the corresponding state in accordance with the state set prepared with this environment model can be made as described in Fig. 3.2. In the k-th sampling, robot adopt one of the elements within the state set as the current state after comparing the state sets with this environment data which has been recognized.

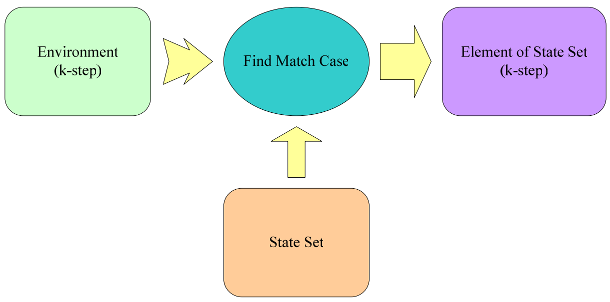
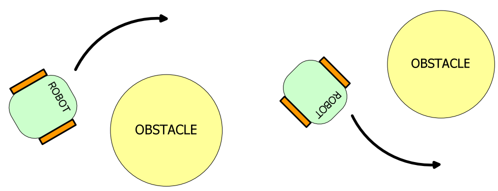


Fig.3.2 Searching for an Element within State Set

3.2 Action Set

Action set can be defined as $A = \{a_1, a_2, a_3, ..., a_m\}$ and the number of elements(m) also depends on how to do modeling the action of the robot. This paper constructs the set $A = \{a_1, a_2, a_3, ..., a_m\}$ as shown in Fig. 3.3



a_{al} : Avoid Left a_{ar} : Avoid Right

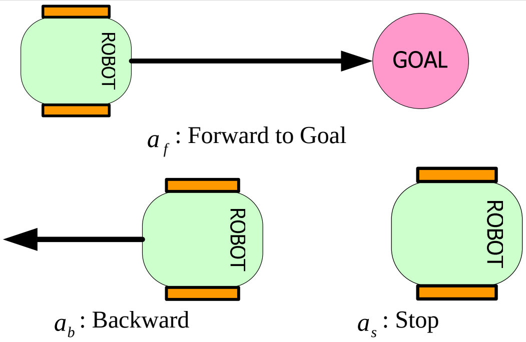


Fig. 3.3 Action Set

As such, the robot prepared with the action model as such will determine the action to take after finding the element in the action set corresponding to the state element which has been made in the state set Fig. 3.4

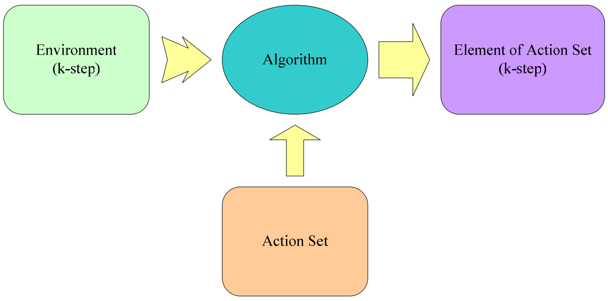
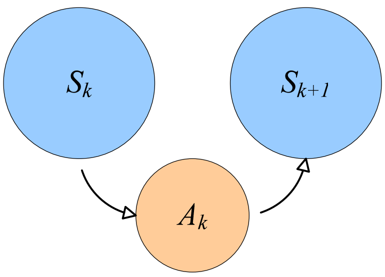


Fig. 3.4 Finding-out an Element in the Action Set

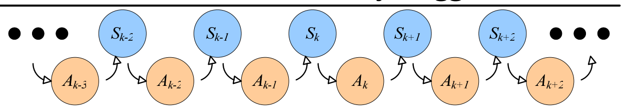
4. TRANSITION of STATE-SET & ACTION-SET

Assuming the present as k-th sample, S_k is one of the element of the set S $s_1, s_2, s_3, ..., s_n$ $(S_k \in S)$, and the action following this A_k can be defined as one of the elements of the Set A $a_1, a_2, a_3, ..., a_m$ $(A_k \in A)$.

In other words, assuming the state of environment model at the sample step (k) as S_k, then following this can be selected and accordingly this make the robot act to move and to generate next state S_{k+1}. Accompanying A_k which follows S_k, the transition to the next state S_{k+1} is occurred. This state is called transition which is shown in Fig. 4.1.



(a) State Transition $(S_k \in S)$, $(A_k \in A)$



(b) Continuous State Transition according to the flow of the sampling time

Fig. 4.1 State Transition

Assuming S_k as s_1 out of the state models in Fig. 3.1, S_{k+1} is to be succeeded to a state among s_1, s_2 or s_3 according to the implementation of one of the A_k elements based on the algorithm. Fig. 4.2 describes the change in the set of the state action till the robot is addressing to the goal in case it is on s_1, avoiding obstacles.

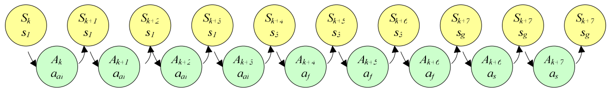


Fig.4.2 State Transition of Robot's Movement

Looking into the resulting output, therefore, the output is shown as listing the elements of the set a depending on time and the robot bring it to converge on the desired state when it performs as listed.

Also letting the location following the sample be P_k in such a view that the reaction of the state set according to the state set means the change in the robot location, the following result as shown in Fig. 4.3 can be obtained as seeing the change in P_k by implementing the action set of A_k $(A_k \in A)$ according to the state set of S_k $(S_k \in S)$.

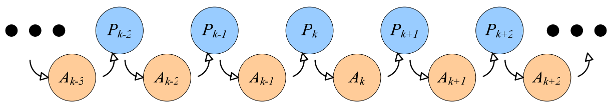


Fig. 4.3 Change in Location in the Continuous Time Flow

5. APPLICATION to ROBOT SYSTEM

When moving robot system used in the experiment is performing the action to address the destination, the flow of overall signal is as shown in Fig. 5.1. Once the destination is determined, environment recognition part recognizes the current state of robot and can get the current state S_k through environment model set. Based on this, A_k can be generated through the algorithm.

In the path-planning(Fig. 5.2) sub-goal is made to act according to the action set(A_k) with the coordinates of GGM, and line velocity and angle velocity of robot to move to the sub-goal in consideration of the current action state of the robot in motion generator belonging to the subordinate controller. The theory of architecture to be used to solve current architecture of robot to transform the line and angle velocity into the velocity of both wheels and to make out output.

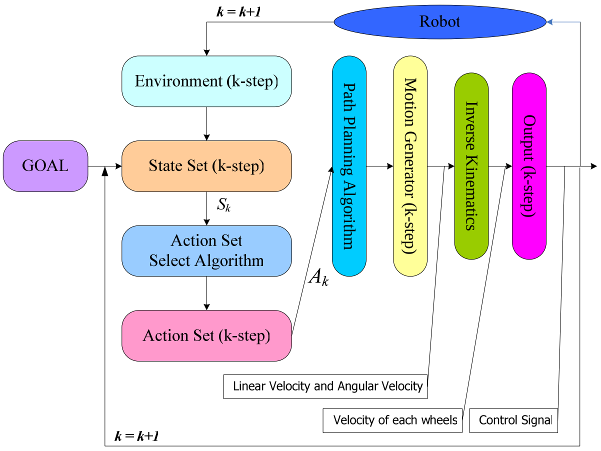
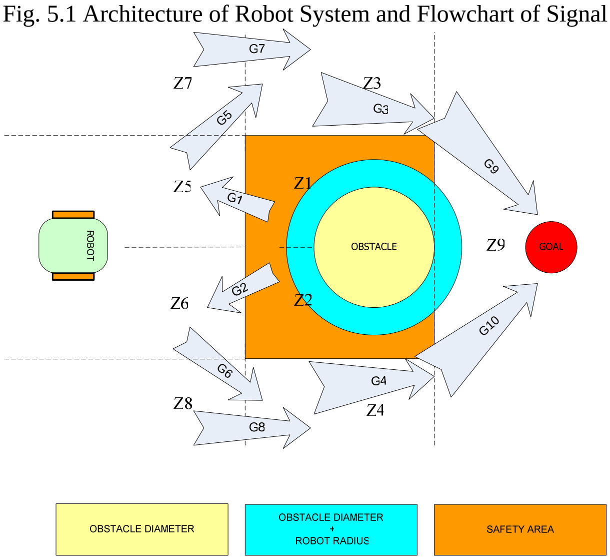


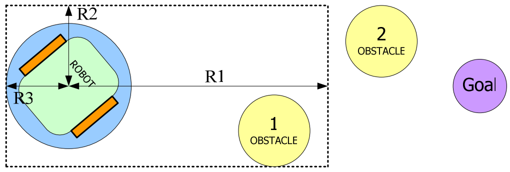
Fig. 5.1 Architecture of Robot System and Flowchart of Signal



.Fig. 5.2 Condition for Generating Sub-goal for Path-planning

The system architecture suggested by this paper does not require for recognizing all the environment of the whole zone in doing the environment, that is, it means that there is no need to recognize in advance the obstacles located far away. It's more efficient to recognize the obstacles only with the state of adjacent environment in consideration of the robot velocity. The question how far the architecture recognizes is called as dynamic window DW and defined in this experiment as follows; Fig. 5.3 shows it. As seen in the Fig., the obstacle 1 can recognize the state as being in the DW zone, but the obstacle 2 does not recognize it as being outside the DW zone.

Also the zone reachable in a specific time period is established. That is, applying variable window according to the robot velocity is referred to as (reachable zone) RZ and as (admissible zone) AZ for that excluding the obstacles within this zone. Fig. 5.4 depicts it where the zone within the dotted line is RZ and that without the obstacles AZ.



$$t = \frac{v_d - v_0}{a}$$

$$R1 = \frac{v_0 t + 0.5 \cdot a \cdot t^2}{g} + h$$

$$R2 = R3 + \frac{v_0 t + 0.5 \cdot a \cdot t^2}{i} + \frac{RobotAngle}{j} + k$$

$$R3 = \frac{RobotWidth}{2} + l$$

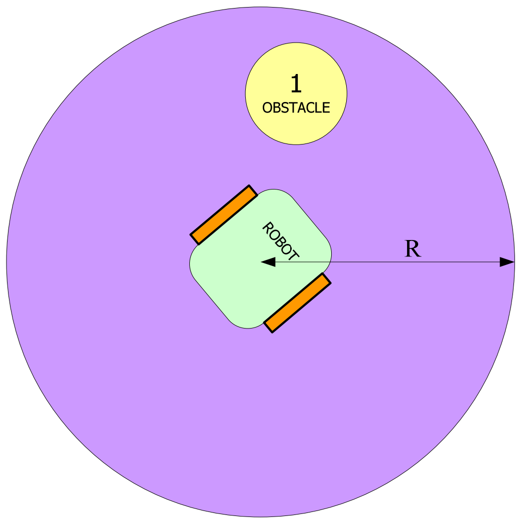
v_d :desired velocity

v_0 :current velocity

a :acceleration

$g \neq 0, i \neq 0, j \neq 0$

Fig. 5.3 Dynamic Window



$$R = \frac{v_{max}}{m} + n \quad (m \neq 0)$$

v_{max} :maximum velocity

Fig. 5.4 reachable zone and admissible zone

AZ_k can be defined as follows, and assuming the obstacle zone as OZ and the zone reachable by the robot under current state as RZ_k, it comes $AZ_k = RZ_k \cap \overline{OZ}$ as shown in the above Fig, where $P_{k+1} \in AZ_k$ is always satisfied. Robot can be converge on the desired location as the time passes. The arbitrary constants g, h, i, j, k, l, m, n in Fig. 5.3 and Fig. 5.4 are the values to be obtained by the experiment.

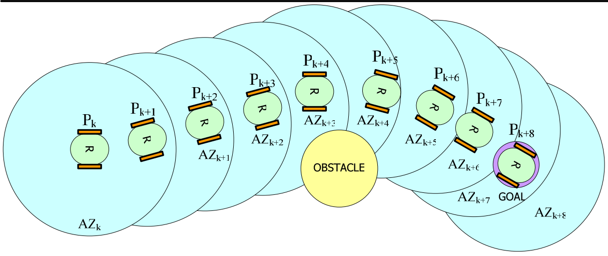


Fig. 5.5 AZ_k, RZ_k, P_{k+1} in Continuous Movement

6. RESULT from EXPERIMENT

The experiment on the navigation has been made by applying robot soccer system on the basis of Goal-oriented Geometric Model(GGM) Based Intelligent System Architecture suggested by this paper where 5 robots performed navigation with different destination.

There exist 4 moving obstacles excepting its own when the 5 fixed obstacles and 5 robots move. That is, navigation has been made for the total 9 obstacles consisting of the 5 fixed obstacles and the 4 non-fixed obstacles. The system construction and time to spend are as follows;

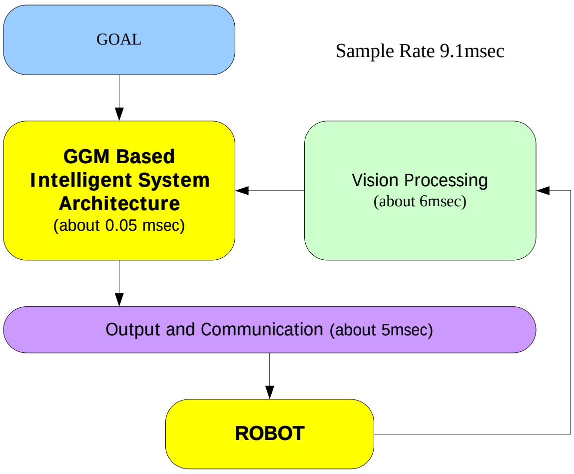


Fig. 6.1 System Flowchart

P4 3.2GHz PC has been used for the experiment in which 5 robots has been located on the bottom left of soccer stadium (5:5 235cm X 180cm) and 5 robots moved towards the upper right. As seen in the Fig. it takes only 0.05ms for the algorithm when carrying out the experiment with the control architecture suggested by this paper.

Fig. 6.2 and 6.3 show the process which have been captured by snapshot. As seen in the result, the robots have successfully moved to the destination, avoiding them excluding the 5 fixed obstacles and its own.

Apart from the fixed object, it is not easy to know how the movement of moving object(robot excluding own) is occurred and it is found that this algorithm has indicated tenacity and responsiveness under the situation where errors exist.

7. CONCLUSION

The algorithm proposed in this paper, as known in the result from the experiment, has shown a good performance in the uncertain and changing environment according to time. When the robot is used at home, it can be put under the changing and uncertain environment according to time as that in the experiment. This occurs both in navigation problem and in the overall actions of robots. It is, therefore, emphasized to design the control architecture with more tenacity and responsiveness if we can do modeling of the jobs given to robot and its action.

Later it is planned to apply the controller suggested in this paper to the assigned jobs to robot to evaluate the performance casting off navigation problem. Also when selecting the action set in the state set, various selection algorithm will be put into place to consider the shortest time, safety and etc.

ACKNOWLEDGMENTS

This research was supported by University IT Research Center Project.

REFERENCES

[1] Agah A., Tanie K., "Robots Playing to Win : Evolutionary Soccer Strategies," *IEEE Int. Conf. Robotics & Automation*, pp. 632-637, Albuquerque, New Mexico April 1997.

[2] Arkin R. C., "Motor Schema Based Mobile Robot Navigation," *International Journal of Robotics Research*, vol. 8, no. 4, pp. 92-112, 1989.

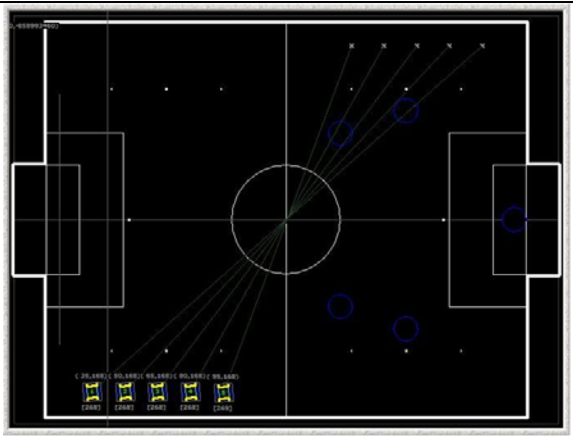
[3] Maes P. and Brooks R., "Learning to Coordinate Behaviors," *Proceedings of the Eighth National Conference on Artificial Intelligence(AAAI)*, pp. 796-802, August, 1990, Boston.

[4] Mataric M., "Synthesizing Group Behaviors," *Proceedings of the Workshop on Dynamically Interacting Robots, International Joint Conference on Artificial Intelligence(IJCAI)*, pp. 1-10, August 1993, Chambery, France.

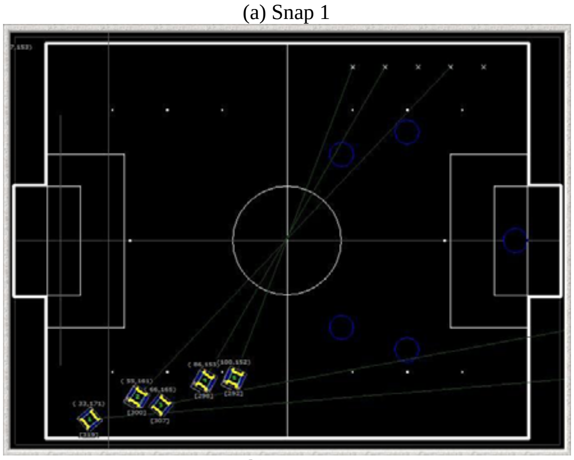
[5] John Craig, "*Introduction to Robotics: Mechanics and Control*," Addison-Wesley, 1986

[6] *Micro-Robot World Cup Soccer Tournament (MIROSOT '97) Proceedings*, KAIST, Daejon, Korea, 1997.

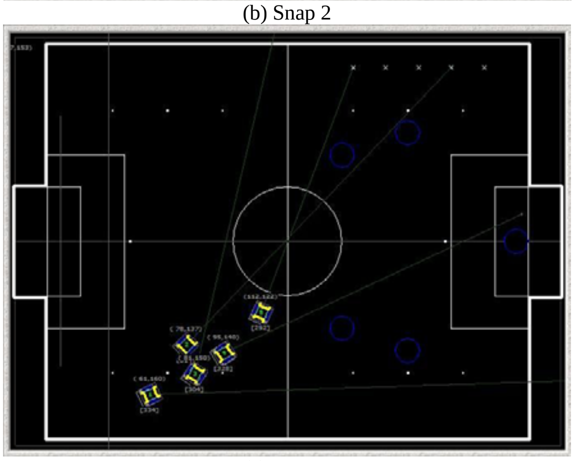
[7] Micro-Robot World Cup Soccer Tournament (MIROSOT '96) Proceedings, KAIST, Daejon, Korea, 1996



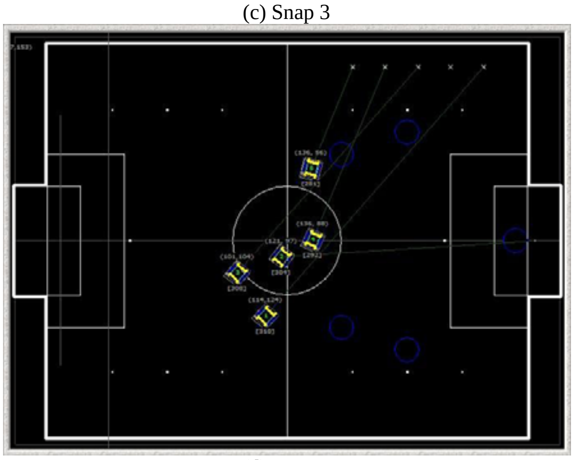
(a) Snap 1



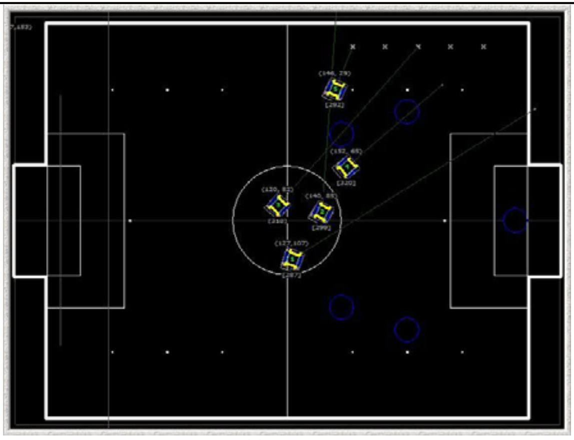
(b) Snap 2



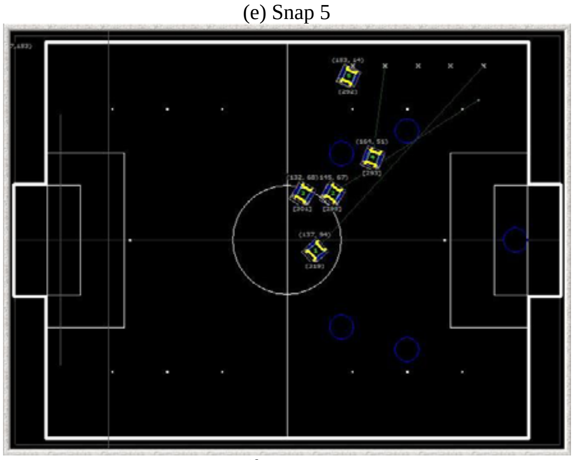
(c) Snap 3



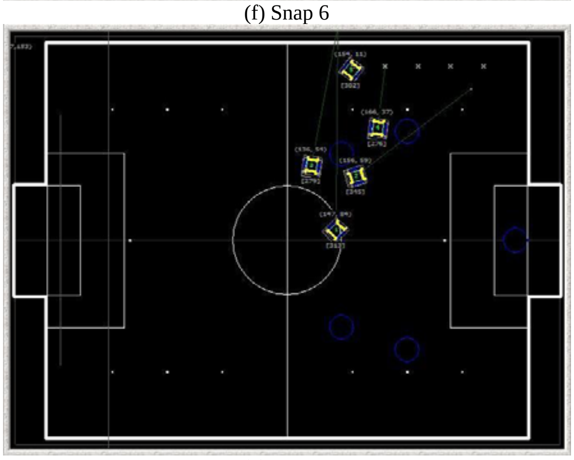
(d) Snap 4



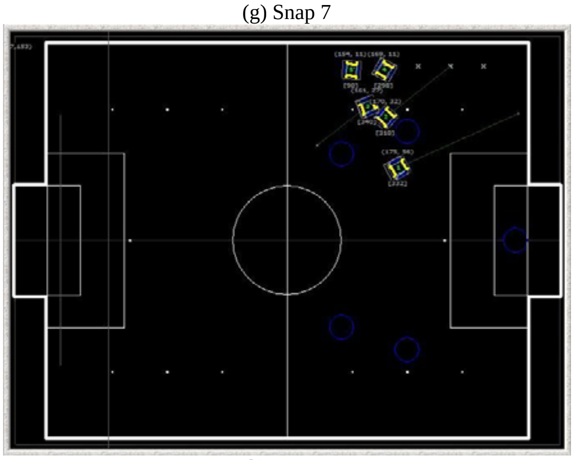
(e) Snap 5



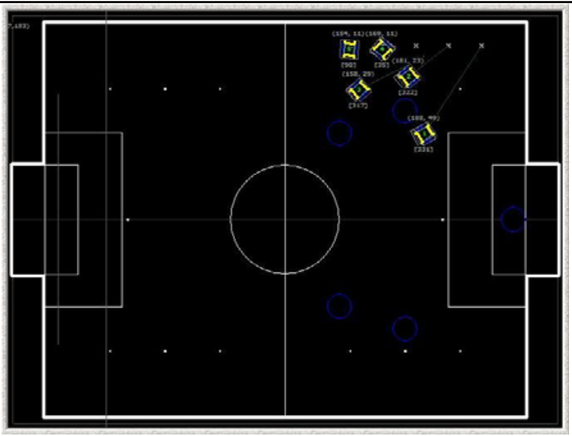
(f) Snap 6



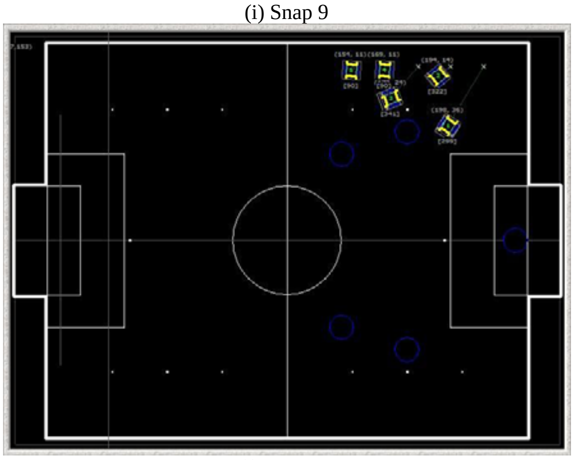
(g) Snap 7



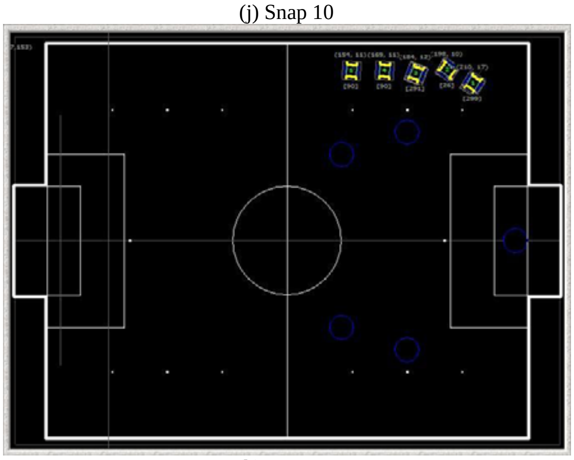
(h) Snap 8



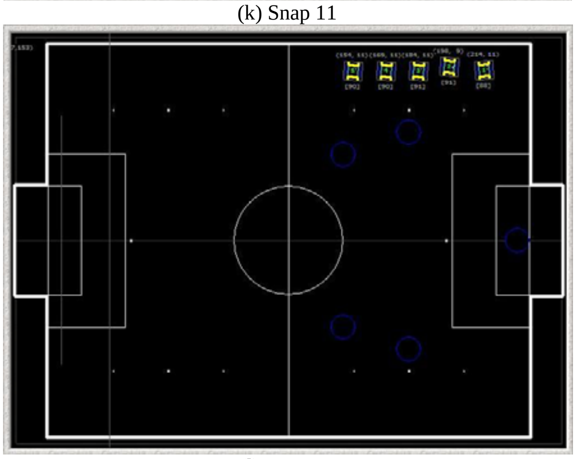
(i) Snap 9



(j) Snap 10

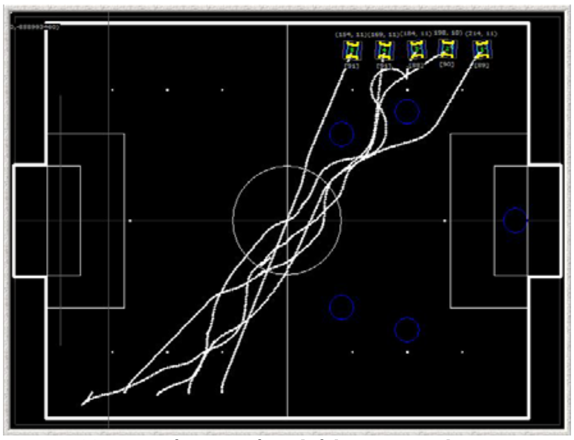


(k) Snap 11

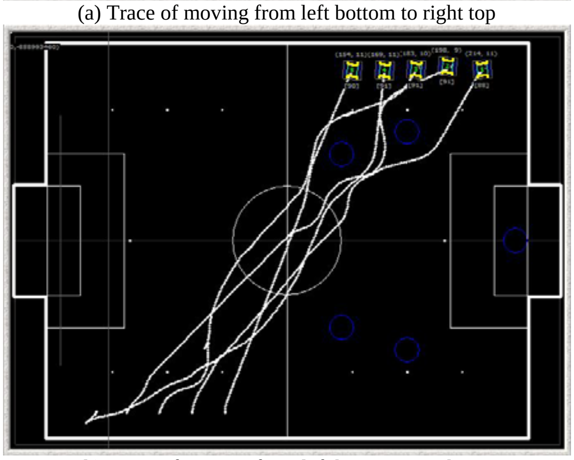


(l) Snap 12

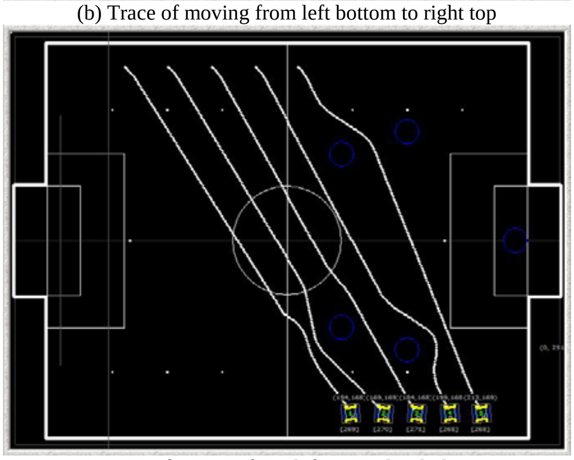
Fig. 6.2 Snap shot of moving robots.



(a) Trace of moving from left bottom to right top



(b) Trace of moving from left bottom to right top



(c) Trace of moving from left top and right bottom

Fig. 6.3 Trace of moving robots.