Lifelike Behaviors of Collective Autonomous Mobile Agents

Suk Ki Min[a], and Hoon Kang[b]

a. Dept. of Control and Instrumentation Engineering Chungang University
221 Huksuk-Dong, Dongjak-Ku, Seoul, 156-756 Korea
Tel:+82-2-820-5320 Fax:+82-2-816-1856 E-mail:dolki@sirius.cie.cau.ac.kr

b. Dept. of Control and Instrumentation Engineering Chungang University
221 Huksuk-Dong, Dongjak-Ku, Seoul, 156-756 Korea
Tel:+82-2-820-5320 Fax:+82-2-816-1856 E-mail:hkang@cau.ac.kr

Abstract

We may gaze at some peculiar scenes of flocking of birds and fishes. This paper demonstrates that multiple agent mobile robots show complex behaviors from efficient and strategic rules. The simulated flock are realized by a distributed behavioral model and each mobile robot decides its own motion as an individual which moves constantly by sensing the dynamic environment.

Keywords

flocking, complex bebavior, artificial life, autonomous mobile robot

I. Introduction

Autonomous mobile robot agents (AMRAs) have been studied for years to investigate emergent behaviors of collective or cooperative intelligence in terms of evolutionary strategies and self-organization in artificial life.[1] These possess the characteristics of autonomy and decentralization, which are the main features of living thing in natural world, and determine its own behaviors independently. This model have the advantage that it doesn't increase the complexity of system even though the number of agent increases. Moreover, with the study of architecture and behaviors of living thing a robot takes up most suitable action according to its local perception of the dynamic environment. Brooks proposed behavior-based robots[2] and there are results that produce complex and purposive group behaviors from a set of simple local interactions such as homing, following, dispersion, aggregation, collision avoidance.[3][4] Craig realized the collision avoidance and the flocking of birds, known as "boid", using fuzzy rules.[5] The problem brought up for discussion in this paper is that multiple AMRAs move around in flock with simple strategic rules. We verify the effectiveness of the proposed method by simulation.

II. Autonomous Mobile Robot Agents (AMRA)

Autonomous mobile robots considered here are primarily multiple agents and their movement is determined by interactions among those agents. The followings are hypotheses imposed on the AMRA system:

Hypotheses (mobile robots):

A1. The AMRA system use the global coordinates system.

A2. The robot speed is constant.

A3. Each robot is able to detect other robots in any direction..

A4. The maximum rotation angle of the mobile robots in one step is exist.

A5. The physical size of the mobile robots is small enough with respect to that of the window region.

Although it's difficult to implement the above physical hypotheses due to overlapping mobile robots and the cost of sensors, the relative position of neighborhood mobile robots can be found with ultrasonic or laser sensors. Also the gaps between computer simulation and real environments can be reduced with the evolution of robot or the sampling of real world.[5] The sensing area of AMRA is shown in Fig. 1 and this is separated two area, that is flocking area and collision avoidance area.

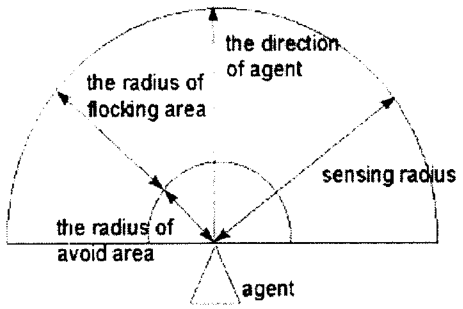


Fig. 1 The sensing radius of a AMRA

Table 1 shows the information that each agent holds. A weight value is an important factor to decide the next state of AMRAs. The function of each information will be described in the following section in detail.

Information	Data Type	Function
(x,y)	coordinates (real, real)	cartesian coordinates of agents
w	weight (real) [0,1]	the strength of connectivity
flag	orientation parameter -1,0,1	-1: left turn 1: right turn in obstacle avoidance mode
(r, θ)	propagation parameter (real, real)	coordinates of the next state
$\overrightarrow{p_{ij}}$	pointer to the other agents (real, real)	

Table. 1. The data structure of ith agent

III. Classifying AMRA behavior type

AMRAs have following 4 types of behavior.

1. random motion
2. flocking motion
3. boundary(obstacle) avoidance motion
4. collision avoidance motion

1.1 random motion

If no object exists within sensing area, it behaves (moves) freely within maximum degree (which it can move) in one step.

$$\vec{d}(t) = \vec{d}(t-1) + random(LIMIT) \quad (1)$$

$\vec{d}(t)$ means direction of robot in time t and LIMIT is the maximum degree as it can move in one- step.

1.2 flocking motion

When other AMRAs exist in a flocking area, a robot evaluates the distance function of the other robots so that it may follow or avoid other robots. If other robots exist in the flocking area, $\overrightarrow{p_{ij}}$, the position vector of distance and location about others as seen fig. 2, becomes system input. Here a direction component of a robot in the next step can be calculated by summing two vectors; one is the input position vector($\overrightarrow{p_{ij}}$) multiplied by the proper weight(w_i), the other is the orthogonal unit vector ($\overrightarrow{o_{ij}}$). If the number of AMRA in flocking area is more than two, the direction of an agent is determined by vector-summing each component. We use the summation of vector as the main operator for the calculation of the system output. The reason is that a robot will move directly to the high-density area. The concept is shown in figure 2, and the system block diagram is shown in fig. 3.

$$\vec{d}(t)=\sum_{j=0}^{N-1}(w_i\cdot\overrightarrow{p_{ij}}+\overrightarrow{o_{ij}}),\quad i\neq j \qquad (2)$$

In equation (2), $\vec{d}(t)$ means the direction of a robot in time t, $\overrightarrow{p_{ij}}$ is the position vector from ith agent point to jth agent, $\overrightarrow{o_{ij}}$ is the unit vector orthogonal to $\overrightarrow{p_{ij}}$, and N is the number of agents. There are two orthogonal vector in $\overrightarrow{o_{ij}}$ and if other robot exist on right side it choose the left orthogonal vector, otherwise the right orthogonal vector. It is a problem to have the proper weight. What value is proper? First, consider θ as a function of the distance $|\overrightarrow{p_{ij}}|$ in fig. 2. When another robot appears on the maximum sensing distance, We will define x be as $w\cdot|\overrightarrow{p_{ij}}|$.

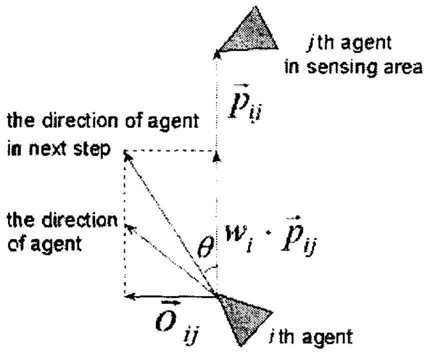


Fig. 2. The relation of position vector $\overrightarrow{p_{ij}}$ and orthogonal unit vector $\overrightarrow{o_{ij}}$ to this vector.

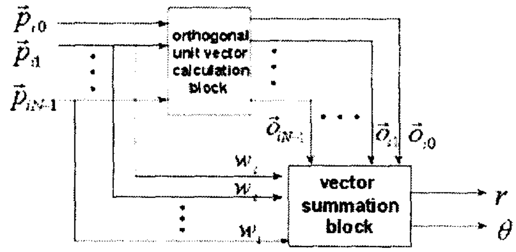


Fig. 3. System block diagram for ith agent

Our ideal guess is the function θ vs. the distance as shown in figure 4-(a), in the real flocking behavior will be $\tan^{-1}$ as shown in figure 4-(b).

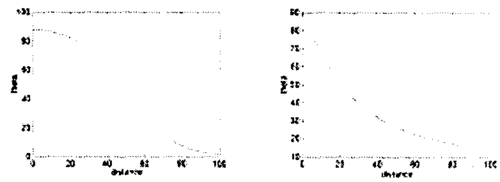


Fig. 4. (a) an ideal function of distance vs. (b) a real function of distance vs.

Then the function θ of distance will be $\tan^{-1}(\frac{1}{x})$ and w can be determined by the selection of proper curve. Increasing x emphasizes the property of following other robots, but the probability of collision grows high and vice versa. Therefore, the selection of the proper x is important. Note that increasing x means the following behavior on the straight line. In fact, so it has less probability of flocking.

1.3 obstacle(boundary) avoidance motion

When there are obstacle or boundary, It is similar to above the flocking. The only difference is to use the flag bit. It has three states which are TURN LEFT, TURN RIGHT, and NO OBSTACLE. If it is turning right, it keeps turning until there is no obstacle (boundary) on moving direction. It means robot keeps direction decided once until success to avoid obstacle (boundary). The flag bit is effective only in avoiding obstacle (boundary) mode.

```
if [ on my left side or flag=RIGHT TURN]
    choose right orthogonal unit vector
    set flag RIGHT TURN
 else if [ on my right side or
          flag=LEFT TURN]
      choose left orthogonal unit vector
      set flag LEFT TURN
      else reset flag NO OBSTACLE
fi
```

4. Collision avoidance motion

When other robot comes in avoiding area, it is important to decide which direction it will move to the next step. In this work it choose the safest direction as checking the danger ratio. The definition of the danger ratio is as follows:

$$\text{danger ratio} = 1 - \frac{|\overrightarrow{p_{ij}}|}{AR} \tag{3}$$

AR is the radius of collision avoidance area.

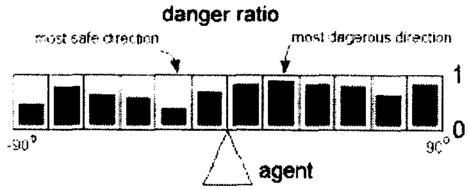


Fig. 5 danger ratio

Fig 5 shows an example of the danger ratio. A sector divided sensing area with an arbitrary size is called a cell. The value of each cell denoted by **C[i]**. Robot has the limited angle in one step, so it choose a cell that has the smallest value within cells considering the limit angle.

$$\vec{d}(t) = \vec{d}(t-1) + DIR(\min_{i=0}^{n-1} C[i]) \tag{4}$$

In above equation (4), n is the number of cells, DIR is the operator to return $d\theta$ represented by each cell. If the minimum value of **C[i]** has more than two, eq. (5) is repeated several times.

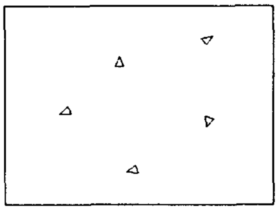
$$C[i] = \begin{cases} \dfrac{\mathrm{V}}{\sum_{k=0}^{m} \mathrm{V}[k]} \cdot \begin{bmatrix} C[i-m/2] \\ \vdots \\ C[i-1] \\ C[i] \\ C[i+1] \\ \vdots \\ C[i+m/2] \end{bmatrix} & 0 \le i \le n-1 \\ 0 & otherwise \end{cases} \tag{5}$$

The m is the number of the neighbor cells including itself, and **V** is a $1 \times \mathrm{m}$ vector. We have chosen n=12, m=3, and **V**=[1 2 1] in this paper. By setting **V**=[1 2 1] it means that the robot affected by itself more than neighbor cells, and whenever such process is repeated the self-influence will be spread as much as

$\frac{m}{2}$ cells. So, if it repeats the process of equation (5) l times, the ith cell can be affected by those cells form the ith to $(i\pm\frac{l\cdot m}{2})$th cell.

IV. Simulation Result

Based on the rules described earlier, we have the following results. Figure 6-8 shows the process of flocking by setting AR=20 (collision avoidance radius), N=5 (the number of agent), and w=0.04(weight). Fig. 9 shows an unexpected phenomenon resulting in a circle during flocking process.



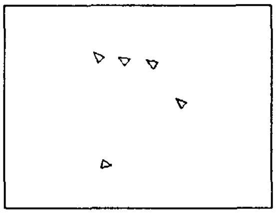
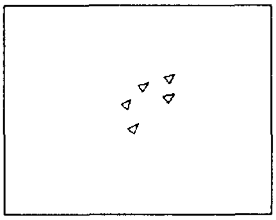


Fig. 6. initial status Fig. 7. transition status



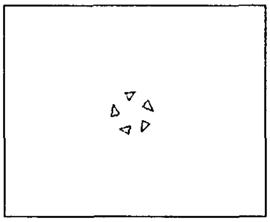


Fig. 8. flocking status Fig 9. unexpected status

V. Conclusion and Future Works

This paper addresses the simplicity of the flocking rules with which the robots move around in the dynamic environment. Even though the robots are not evolved by dynamic environment, we shows the weight has an important properties to decide the next behavior, so we expect to make more efficient strategy not by predefining the weight values, but by modifying the weight values of the control structure properly according to its local perception of the dynamic environment.

VI. Reference

[1] C. Langton, "Artificial Life" in *Artificial Life*, C. Langton (ed.), Addison-Wesley, pp.1-47, 1989

[2] R. A. Brooks, "Behavior Humanoid Robotics", *Proc. of Int. Conf. on IROS.* pp.1-8, 1996

[3] M. J. Mataric, "Designing Emergent Behaviors: From Local Interactions to Collective Intelligence", *Proc. 2nd Int. Conf. on Simulation of Adaptive Behavior*, pp.432-441, 1993

[4] C. R. Kube and H. Zhang, "Collective Robotic Intelligence", *Proc. 2nd Int. Conf. on Simulation of Adaptive Behavior*, pp.460-468, 1993

[5] Craig W. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral Model", *Proc. of SIGGRAPH*, pp.25-34, 1987

[6] O. Miglino, H. H. Lund, and S. Nolfi, "Evolving Mobile Robots in Simulated and Real Environments", *Artificial Life*, vol.2, no.4, pp.417-434, MIT Press, 1996