Fast and Efficient Formation Flocking for a Group of Autonomous Mobile Robots

Naixue Xiong[1,2], Yingshu Li[1]
[1]Depa. of Computer Science
Georgia State University, Atlanta, USA
{nxiong, yli}@cs.gsu.edu

Laurence T. Yang
Depa. of Computer Science
St. Francis Xavier University, Canada
lyang@stfx.ca

Jong Hyuk Park
Depa. of Computer Science and Engineering
Kyungnam University
449 Wolyoung-dong, Masan
Kyungnam, 631-701, Korea
parkjonghyuk1@hotmail.com

Yan Yang[2], Sun Tao[3]
[2]School of Information Science
Japan Advanced Insti. of Scie. and Tech.
[3]Wuhan Second Ship Design &
Research Institute, Wuhan, 430064, China
y.yang@jaist.ac.jp

Abstract

The control and coordination of mobile robots in groups that can freely cooperate and move on a plane is a widely studied topic in distributed robotics. In this paper, we focus on the flocking problem: there are two kinds of robots: the leader robot and the follower robots. The follower robots are required to follow the leader robot wherever it goes (following), while keeping a formation they are given in input (flocking). A novel scheme is proposed based on the relative motion theory. Extensive theoretical analysis and simulation results demonstrate that this scheme provides the follower robots an efficient method to follow the leader as soon as possible with the shortest path. Furthermore, this scheme is scalable, and the processing load for every robot is not increased with the addition of more robots.

Keywords. *Distributed system, Mobile robots, Flocking algorithm, Motion theory, Control and coordination.*

1. Introduction

The control and coordination of mobile robots in groups is a widely studied topic in distributed robotics. Robots are used to relieve humans of tasks that are included in the four aspects of robotics: dirty, dull, dangerous or difficult [6, 13-17]. It is much like man that has always developed technology to save time for other more challenging tasks, now the time has come for robots. Mobile robots cooperating in groups offer several advantages, e.g., redundancy and flexibility, and can sometimes perform tasks that would be impossible for one single robot [2, 3, 5, 7]. Recent advances in robotics have started making it feasible to deploy large numbers of inexpensive robots for tasks such as surveillance and search. However, coordination motion of multiple robots in a plane to accomplish such tasks remains a challenging problem.

In this paper, we study the flocking problem, where there are two kinds of robots: the leader and the follower. The leader acts independently of the others, and we can assume that it is driven by a human pilot or supervisor. Every follower can distinguish the leader from the other followers. And the followers are required to follow the leader wherever it goes (following), and stay in a formation they are given in input (flocking). In this context, a formation is simply a pattern described as a set of points in the plane. And all the robots have the same formation in input. There are several applications for flocking of a group of robots, such as in military [8], or in factories, where robots can be asked to move heavy loads [4].

Although many papers have addressed the flocking problem, few studies have focused on the efficiency of the algorithm [18-19]. To this question, this paper presents a novel scheme to obtain the shortest distance and the shortest time to finish the formation and move in some pattern. Simulation results further demonstrate that this can solve the flocking of a group of autonomous mobile robots quickly.

The remainder of the paper is organized as follows: Section 2 discusses more work related to flocking problem. In Section 3, the system model and basic concepts are defined.

Section 4 presents the novel flocking algorithm. In Section 5, we evaluate the performance of the proposed algorithm. Finally, we conclude our work and discuss further work in Section 6.

2. Related work

In order to design and evaluate the solutions of flocking, the approach usually adopted is to design solutions based on heuristics and tailored to the capabilities of the robots employed, and then test the solutions by computer simulations, or on real robots.

For instance, experiments in [9] were conducted on a team of simple mobile units in order to produce complex behaviors, by compounding basic ones. For example, safe-wandering, i.e., the ability to avoid collisions while moving; dispersion, i.e., the ability of the robots to spread out over an area; aggregation, i.e., the ability to gather; and gathering, i.e., the ability to reach a predetermined destination, and so on. In particular, the author points out that flocking can be obtained by combining safe-wandering, aggregation, dispersion, and gathering.

T. Balch and C. Arkin studied formation and navigation problems in a team of multi-robots. In particular, in [10] the problem of specifying the behavior for the navigation of a mobile unit is analyzed, and results of both computer simulation and real experimentation are reported. In [8], the approach is extended to multi-robot teams that navigate the environment maintaining particular formations: in particular the cases of a line, column, diamond and wedge are examined. In their study, the authors assumed that the path along which the group of robots had to move was known in advance to every unit.

In [11], the robots were asked to move in a matrix shape along a path represented by a straight line followed by a right turn and then a straight line again. In contrast, in this paper we do not assume any knowledge by the followers of the path that the leader will follow. The followers have only a common description of the formation they have to keep while moving.

A similar problem is studied in [12], where the author derives equations describing navigational strategies for robots moving in formation, and following the movement described by one (or more) leader. In the studied framework, the robots have identities, hence their positions in the formation are fixed. Moreover, in order for the i^{th} robot to compute its position at time t, it has to know the position of either the $(i-1)^{th}$ robot or the leader at time t. Hence, some degree of synchrony has to be introduced in order to implement these strategies.

In [1], an algorithm solving this problem has been tested by using computer simulation; the algorithm assumes no agreement. All the experiments demonstrated that the algorithm performed well, and in all cases the followers were able to assume the desired formation and to maintain it while following the leader along its route. Moreover, the oblivious of the robots in the algorithm contributes to the result, since the followers do not base their computation on leader's past positions. Unfortunately, in all the above papers, there is no consideration about the speed, i.e., how fast the followers can follow the leader to form into a certain pattern.

3. System Model

We study the distributed coordination and control of a set of asynchronous, anonymous, memory-less mobile robots that can freely move on the two-dimensi-onal plane, and the leader can communicate with the followers. In particular, we study the flocking problem, i.e., to form a certain pattern and follow a designated robot, the leader, while maintaining the pattern. And, we assume all the robots are viewed as points, that is, the size of the robot is not considered. The robots can not communicate each other (see Fig. 1).

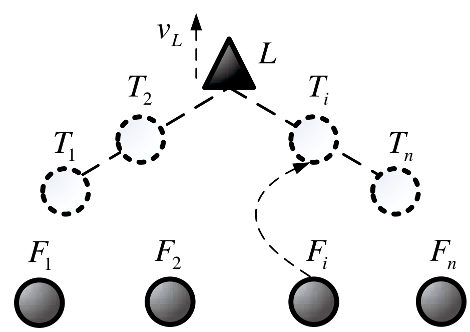


Figure 1. A basic robot model for directed targets [4].

The robots unit have simple computational capabilities. They are equipped with sensors that let all robots observe the positions of the leader of the flock. All robots are able to observe their surroundings, computing a destination based on what they observed, and moving towards the computed destination; hence they performs an (endless) cycle of observing, computing, and moving [4].

All robots have their own local view of the world. This view includes a local Cartesian coordinate system having an origin[1], a unit of length, and the directions of two coor-

[1] Without losing generality, we can assume the origin of the coordinate system to be the position of the robot.

Table 1. Parameter description

Parameter	Description
v_L	The velocity of the leader L
F	A follower robot
v_F	The velocity of the follower F
T	The target position of the follower robot F
v_{max}	The maximum available velocity for each follower

dinate axes[2], together with their orientations, identified as the positive and negative axes. In general, there is no agreement among the followers on the properties of the local coordinate systems (i.e., the robots have different concepts of where North, East, South, and West are) [4].

4. The New Flocking Algorithm

In detail, we consider the basic robot model for directed targets as shown in Fig. 1 (directed wedge formation). Notice that, there are n multiple followers (F_1, F_2, $\cdots$, F_i, $\cdots$, F_n), one leader (L), and a certain directed target pattern. The follower F_i will get the target T_i and then follow the leader L.

We use the relative motion theory of motion of objects to solve the cooperative flocking of the mobile robots. We assume that every robot has its own coordinates in the plane. In order to make all the followers approach the leader as soon as possible, each follower follows its leader at its maximum available velocity (v_{max}) before reaching its target. Each robot can observe using its own compass, compute and move.

The follower robot can get its relative velocity to the leader by the equation:

$$v = \delta S/\delta t$$

where δS is the difference of the distance between the follower and the leader during the period δt. And the orientation of the velocity can be gotten by the follower's own compass.

In order to further describe our algorithm, we assume the variables in Table 1.

Among these variables, v_L, v_F and v_{max} are the vectors. Based on the relative motion theory, if we assume the leader is a reference object, i.e., the leader is still, then the follower robot F will have a velocity $|v_L|$, and the direction of relative velocity of the follower is opposite to the velocity of the leader L, v_L. In details, we discuss the motion of the followers in the following two cases based on the angle α between the vector v_L and the vector $\overrightarrow{FT}$.

[2]We can refer to two coordinate axes as the x and y axes.

4.1. case 1: $\alpha = 0$ or π

When the direction of the leader's velocity v_L is parallel with the line $\overrightarrow{FT}$ (see Fig. 2(a)), the follower F moves with maximum velocity along the direction of $\overrightarrow{FT}$, i.e., $|v_F| = |v_{max}|$.

In Fig. 2(a), a coordinate system can be built using the position of the follower F as the origin. Assume the coordinate of the target is (x_t, y_t), then the follower F will arrive at the target in the period $\sqrt{((x_t)^2 + (y_t)^2)}/(|v_{max} - v_L|)$.

4.2. Case 2: $\alpha \neq 0$ and $\alpha \neq \pi$

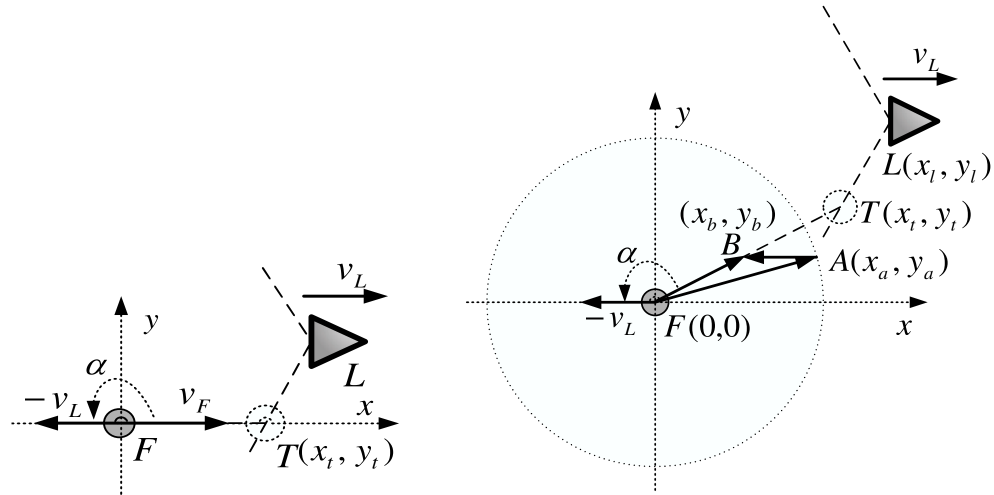
When the direction of the leader's rate of the leader is not parallel with the line $\overrightarrow{FT}$, i.e., there is an acute angle or an obtuse angle between $-v_L$ and the vector $\overrightarrow{FT}$ (see Fig. 2(b), i.e., for the angle $\angle\alpha$, $\angle\alpha \neq 0$, and $\angle\alpha \neq \pi$). The acute angle and the obtuse angle cases have the same analysis algorithm, here we just choose the obtuse angle case as an example.

In general, we assume that the velocity of the leader L is less than that of the followers; otherwise, the followers cannot follow the leader, i.e., $|v_L| < min\{|v_{F1}|, |v_{F2}|, \cdots, |v_{Fn}|\}$ [1]. The motion target of the follower F is to arrive at the target position T that is near the leader, and to form some formation.

In order to describe the algorithm, we build the coordinate system as shown in Fig. 3, and we assume that the coordinates of the leader is $L(x_l, y_l)$, while the follower F is the origin of the coordinate system. The target coordinates of the follower F is $T(x_t, y_t)$.

Therefore, we can get the function of the line $\overrightarrow{FT}$:

$$y = \frac{y_t}{x_t} \cdot x. \quad (1)$$



(a) (b)

Figure 2. The presented fast flocking algorithm: (a) The orientation of the leader's velocity v_L is parallel with the vector $\overrightarrow{FT}$; (b) The orientation of the leader's velocity v_L is not parallel with the vector $\overrightarrow{FT}$.

The detailed description is as follows:

Based on the above coordinate system, we can construct a circle, where the center of the circle is F, and the radius is the velocity $|v_F|$ (here we choose the maximum velocity for the robot to let the robot reach the target T as soon as possible, i.e., $|v_F = v_{max}|$). Then one gets the function of the circle C:

$$x^2 + y^2 = |v_F|^2. \tag{2}$$

If the follower F can reach the target T, then the sum vector of vectors v_F and $-v_L$ must have the same direction as the vector $\overrightarrow{FT}$. So we assume that there is a point B (x_b, y_b), where $\overrightarrow{FB} = v_F + (-v_L)$. Then B must be in the beeline FT described by Function (1). So

$$y_b = \frac{y_t}{x_t} \cdot x_b. \tag{3}$$

Based on the point B, we draw vector $\overrightarrow{AB}$, where point A (x_a, y_a) is in the circle (2) and $\overrightarrow{AB}$ has the same direction as vector $-v_L$. So

$${x_a}^2 + {y_a}^2 = |v_F|^2. \tag{4}$$

If $|\overrightarrow{AB}| = |(-v_L)|$, i.e.,

$$\sqrt{(x_b - x_a)^2 + (y_b - y_a)^2} = |v_L|. \tag{5}$$

Then we can get $\overrightarrow{AB} = -v_L$, i.e.,

$$v_F + (-v_L) = \overrightarrow{FB}. \tag{6}$$

Therefore, the follower F moves with the maximum velocity along $\overrightarrow{FA}$. And it will spend time $|\overrightarrow{FT}|/|v_F + (-v_L)|$ to reach the target T.

Furthermore, $\overrightarrow{AB} \parallel (-v_L)$, and $-v_L$ is in the line axis x, so

$$y_a = y_b. \tag{7}$$

From Equations (5) and (7), we can obtain

$$(x_b - x_a)^2 = |v_L|^2. \tag{8}$$

The angle $\angle\alpha$ is an obtuse angle, so $y_a = y_b > 0$, $x_a > x_b > 0$. Then from Equation (8), we can obtain

$$x_a = x_b + |v_L|. \tag{9}$$

Substitute y_b of Function (3) into Function (4), and $y_a = y_b$, one has

$${x_a}^2 + (\frac{y_t}{x_t})^2 \cdot {x_b}^2 = |v_F|^2. \tag{10}$$

Substitute x_a of Function (9) into Function (10) and obtain:

$$(x_b + |v_L|)^2 + (\frac{y_t}{x_t})^2 \cdot {x_b}^2 = |v_F|^2, \tag{11}$$

i.e.,

$$(1 + \frac{{y_t}^2}{{x_t}^2}) \cdot {x_b}^2 + 2 \cdot |v_L| \cdot x_b + |v_L|^2 - |v_F|^2 = 0. \tag{12}$$

For the unknown parameter x_b in Equation (12), the discriminant $\triangle$ is

$$\triangle = 4 \cdot |v_L|^2 + 4 \cdot (1 + \frac{{y_t}^2}{{x_t}^2}) \cdot (|v_F|^2 - |v_L|^2) \quad (13)$$

Since $|v_L|^2 < |v_F|^2$, then $\triangle > 0$. Then

$$x_b = \frac{-2|v_L| \pm \sqrt{4 \cdot |v_L|^2 + 4 \cdot (1 + \frac{{y_t}^2}{{x_t}^2}) \cdot (|v_F|^2 - |v_L|^2)}}{2(1 + \frac{{y_t}^2}{{x_t}^2})} \quad (14)$$

For

$$4 \cdot |v_L|^2 + 4 \cdot (1 + \frac{{y_t}^2}{{x_t}^2}) \cdot (|v_F|^2 - |v_L|^2) > 4 \cdot |v_L|^2, \quad (15)$$

then

$$\sqrt{4 \cdot |v_L|^2 + 4 \cdot (1 + \frac{{y_t}^2}{{x_t}^2}) \cdot (|v_F|^2 - |v_L|^2)} > 2 \cdot |v_L|. \quad (16)$$

Furthermore, since $x_b > 0$, then

$$x_b = \frac{-{x_t}^2|v_L| + x_t\sqrt{{x_t}^2|v_F|^2 + {y_t}^2(|v_F|^2 - |v_L|^2)}}{{y_t}^2 + {x_t}^2}. \quad (17)$$

Substitute x_b of Equation (17) into Equation (9) and obtain:

$$x_a = |v_L| + \frac{-{x_t}^2 \cdot |v_L| + x_t \cdot \sqrt{{x_t}^2 \cdot |v_F|^2 + {y_t}^2 \cdot (|v_F|^2 - |v_L|^2)}}{{y_t}^2 + {x_t}^2}, \quad (18)$$

i.e.,

$$x_a = \frac{{y_t}^2 \cdot |v_L| + x_t \cdot \sqrt{{x_t}^2 \cdot |v_F|^2 + {y_t}^2 \cdot (|v_F|^2 - |v_L|^2)}}{{y_t}^2 + {x_t}^2}. \quad (19)$$

Substitute x_b of Equation (17) into Equation (3), and $y_a = y_b$, then we can obtain

$$y_a = y_b = \frac{-x_t y_t \cdot |v_L| + y_t \cdot \sqrt{{x_t}^2 \cdot |v_F|^2 + {y_t}^2 \cdot (|v_F|^2 - |v_L|^2)}}{{y_t}^2 + {x_t}^2}. \quad (20)$$

Based on the above analysis, the follower F moves with maximum velocity along the direction $\overrightarrow{FA}$ with the slope $arctan(y_a/x_a)$, i.e.,

$$arctan(\frac{-x_t y_t|v_L| + y_t\sqrt{{x_t}^2|v_F|^2 + {y_t}^2(|v_F|^2 - |v_L|^2)}}{{y_t}^2|v_L| + x_t\sqrt{{x_t}^2|v_F|^2 + {y_t}^2(|v_F|^2 - |v_L|^2)}}). \quad (21)$$

And the time to reach the target T is shown by $|\overline{FT}|/(v_F + (-v_L))$ (i.e., $\sqrt{({x_t}^2 + {y_t}^2)}/\sqrt{({x_b}^2 + {y_b}^2)}$). Substitute x_b and y_b of Equations (17) and (20) into the above equation, and the time is

$$\frac{{x_t}^2 + {y_t}^2}{\sqrt{{x_t}^2 \cdot |v_F|^2 + {y_t}^2 \cdot (|v_F|^2 - |v_L|^2)} - x_t \cdot |v_L|}. \quad (22)$$

After the follower robot reaches its target, it will follow the track of the leader closely. The distance between the leader and the target of the follower F is R_F. When the leader rotates at the joint rate ω, the velocity of the follower F will become $(v_L + R_F * \omega)$ [1].

4.3. Analysis of the algorithm correctness

In the above algorithm, using the follower F as the origin, the coordinate system is built. So each follower has its own local coordinate system. If all followers use a public coordinate system, then the follower F has the corresponding coordinates (x_f, y_f). In this case, we use a similar algorithm, and only exchange the coordinate system, i.e., using $(x - x_f)$ and $(y - y_f)$ instead of x, y respectively in the above analysis. Thus, this algorithm is suitable for a local coordinate system, and a public coordinate system.

The above algorithm is suited to the case that the leader has the same velocity v_L before the followers arrive at the targets. Actually, the leader can change velocity sometimes. Then in that case, each robot should dynamically change the corresponding relative velocity $-v_L$, and change to the new velocity based on the above scheme. So our scheme is still suited to this dynamic case.

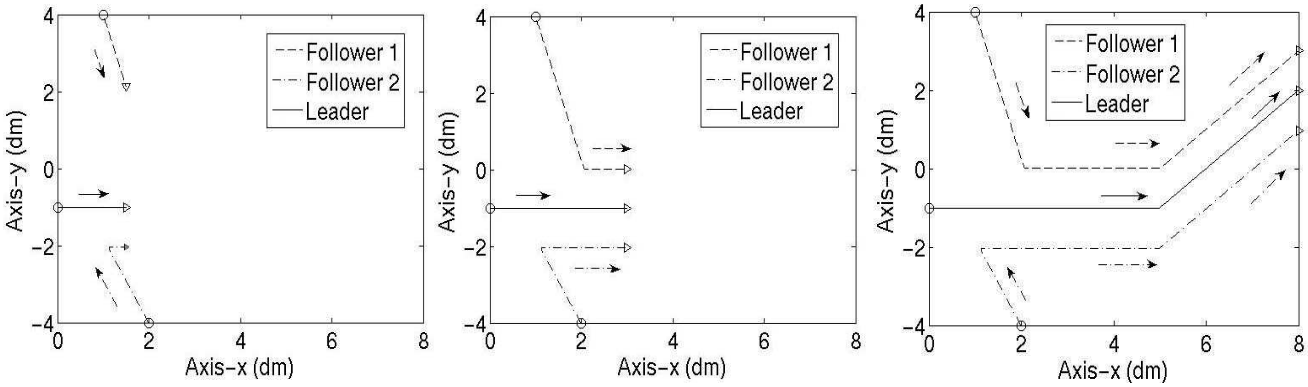
Theorem 1: The follower F moves along the direction of $\overrightarrow{FT}$, and at the velocity of v_{max}, and it must arrive at its target by (1) the shortest distance and (2) the least time.

Proof:

(1) The follower F moves along the direction of $\overrightarrow{FT}$, which is the shortest path to arrive at its target. Based on the geometric theorem, the line segment $\overrightarrow{FT}$ is the shortest path between two points F and T. Here we assume the length of the line segment $\overline{FT}$ is s_{least}, where $s_{least} = \sqrt{(y_F - y_t)^2 + (x_F - x_t)^2}$, here, (x_f, y_f) are the coordinates of the follower F.

(2) In order to save time, the follower F will move as fast as it can, i.e., at the maximum rate of v_L. Based on the equation of velocity, the time $t = s/v$, where s is the distance between the follower and its target, and v is the velocity of the follower robot. Therefore, we can get the least time $s_{least}/|v_F - v_L|$ to reach its target.

From the above analysis, we arrive at the conclusion that in this scheme the follower robot can follow the leader and form a pattern with the shortest path and the least time. And using the same method, all the followers can eventually follow the leader with some predefined pattern.



(a) 1.5 s (b) 3 s (c) 8 s

Figure 3. The navigation track of two followers and the leader. Here before the two followers reach targets, the leader doesn't change direction.

5. Performance Evaluation

To evaluate the performance of the proposed solution of flocking, we focus on the following parameters: path length, and moving time. We assume that the robots moving time at a certain velocity is dominant compared to the acceleration period.

Our method is inherently highly flexible in the kinds of geometric formations that can be maintained. Here we give a representative example, where there are two followers and one leader, and the two targets are both sides of the leader to form a line formation, and each one is 10 cm distant from the leader. The simulation model is similar to Fig. 1. The line connecting the two targets is vertical with the direction of the leader rate, and all the robots use the public coordinate system. The initial place for the two followers and the leader are $(1, 4)$, $(2, -4)$, and $(-1, 0)$ respectively. The maximum available velocity for the followers are the same 20 cm/s, and the leader's velocity are 10 cm/s, while direction can be changed freely.

We divide the simulation into two kinds of cases based on whether the leader changes direction before the two followers reach targets. The simulation results are shown in Figs. 3-4.

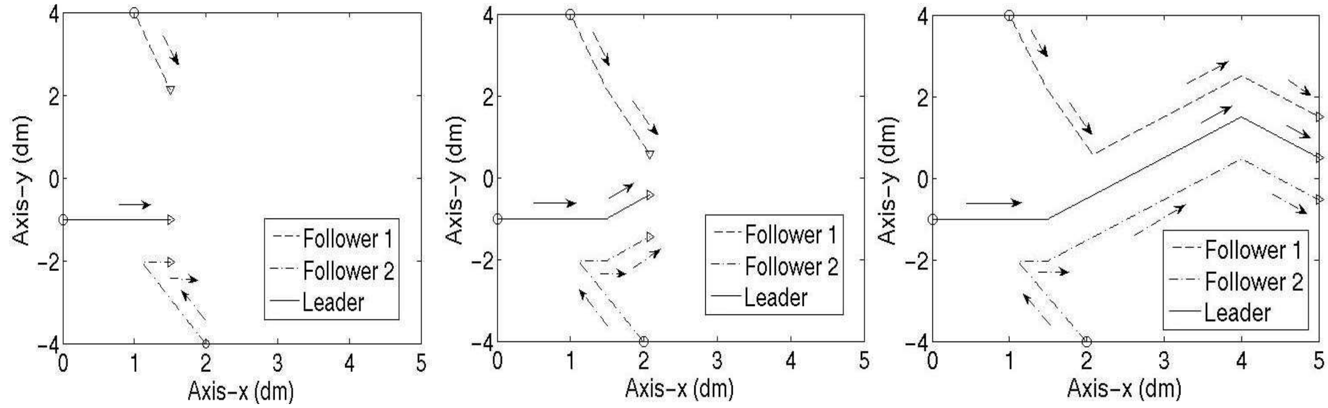
In Fig. 3, before the two followers reach targets, the velocity of the leader doesn't change. With the change of the simulation time, Fig. 3(a), Fig. 3(b) and Fig. 3(c) show the navigation track of two followers and the leader for 1.5 s, 3 s, and 8 s, respectively. In every sub-figure, the circles mean the initial positions of the two followers and the leader. The arrowhead means the position for a mobile robot at the end of the simulation time. From the simulation results, we find that Follower 1 spent $(\sqrt{52} - 1)/3$ s to reach the target $((\sqrt{52} - 1)/3, 0)$, and Follower 2 spent $(\sqrt{28} - 2)/3$ to reach the target $((\sqrt{28} - 2)/3, -2)$. After that, the followers go with the leader in a formation. At 5 s, the leader changed direction, and we could find the velocity of the followers also changed to the same direction as that of the leader, then they kept in a formation.

In Fig. 4, before the two followers reach targets, the leader has changed its direction. Here Fig. 4(a), Fig. 4(b) and Fig. 4(c) show the navigation track of two followers and the leader at 1.5 s , 2.1 s, and 5 s, respectively. For this case we use the same simulation setting as in the above, so we can comparatively analyze them. During the first 1.5 s period, Fig. 3 and Fig. 4 have same results. While, at the 1.5 s, the leader changed its direction. At this time, the Follower 2 has reached the target, while the Follower 1 is on the way, but did not reach the target. In this case, Follower 2 also changed its velocity to the same direction and the same velocity as the leader. While, Follower 1 re-computed the new direction based on the leader rate, then it spent about 2.08 s to reach the target $(2.08, 0.58)$. After that, Follower 1 also changed its rate to the same direction and the same velocity as the leader. At 4 s, the leader changed its rate again. Then the two followers changed their rates accordingly, and kept in a formation.

In summary, the simulation results demonstrated that our algorithm is an efficient solution of flocking with the shortest path and time.

6. Conclusions

This paper addressed the flocking problem of a group of mobile robots, and presented a novel efficient scheme for robot flocking. Furthermore, the extensiveness theoretical analysis proves the effective of this algorithm; and the sim-



(a) 1.5 s (b) 2.1 s (c) 5 s

Figure 4. The navigation track of two followers and the leader. Here before the two followers reach targets, the leader has changed direction.

ulation results demonstrate the efficiency of the proposed algorithm with the shortest path and time. In future work, we will test this scheme with real robots. More mobile robots will be developed to solve the searching and rescuing problems, while with more robots joining, we also should consider conflict avoidance among robots when flocking.

7. Acknowledgment

This research has been supported by the US National Science Foundation CAREER Award under Grant No. CCF-0545667.

We would like to thank many colleagues and anonymous reviewers for their constructive criticism and helpful suggestions for improving the overall quality of this paper.

References

[1] V. Gervasi and G. Prencipe, "Coordination without communication: the case of the flocking problem," Discrete Applied Mathematics, 2003. http://citeseer.ist.psu.edu/article/gervasi03coordination.html.

[2] E. Bahceci, O. Soysal, and E. Sahin, "A review: pattern formation and adaptation in multi-robot systems," Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Technical Report CMU-RI-TR-03-43, October 2003.

[3] G. Prencipe and N. Santoro, "Distributed algorithm for autonomous mobile robots," In International Federation for Information Processing, Vol. 209, Fourth IFIP International Conference on Theoretical Computer Science-TCS 2006, pp. 47C62, 2006.

[4] V. Gervasi and G. Prencipe, "Flocking by a set of autonomous mobile robots," Technical Report TR-01-24, Dipartimento di Informatica, Universita di Pisa, Italy, Oct. 2001. http://citeseer.ist.psu.edu/gervasi01flocking.html.

[5] G. Prencipe, "CORDA: Distributed Coordination of a Set of Autonomous Mobile Robots," In Proceeding of Fourth European Research Seminar on Advances in Distributed Systems (ERSADS 2001), pp. 185-190, May 2001, http://citeseer.ist.psu.edu/prencipe01corda.html.

[6] J. L. Fuller, "Robotics: introduction, programming, and projects". Prentice Hall, 1999.

[7] Magnus Lindhé, "A flocking and obstacle avoidance algorithm for mobile robots," Technical Report IR-RT-EX-0413, Stockholm, Sweden, 2004.

[8] T. Balch and R. C. Arkin, "Behavior-based formation control for multi-robot teams," IEEE Transactions on Robot and Automatics, 14(6), 1998.

[9] I. Suzuki and M. Yamashita, "Distributed anonymous mobile robots: formation of geometric patterns," Sima Journal of Computing, 28(4), pp. 1347-1363, 1999.

[10] E. Bonabeau, M. Dorigo, and G. Theraulaz, "Swarm intelligence," Oxford University Press, 1999.

[11] F. R. Noreils, "Toward a robot architecture integrating cooperation between mobile robots: application

to indoor environment," The International Journal of Robotics Research, pp. 79-98, 1993.

[12] G. Prencipe, "CORDA: Distributed cooperation of a set of autonomous mobile robots," In Proceeding of Fourth European Research Seminar on Advances in Distributed Systems (ERSADS 2001), pp. 185-190, 2001.

[13] J. Cortés, S. Martinez, T. Karatas, and F. Bullo, "Coverage Control for Mobile Sensing Networks," In Proceeding of the IEEE Conference on Robotics and Automation, pp. 1327-1332, Arlington, VA, May 2002.

[14] N. Tomatis, I. Nourbakhsh, K. Arras, and R. Siegwart, "A hybrid approach for robust and precise mobile robot navigation with compact environment modeling," In Proceeding of IEEE International Conference on Robotics and Automation, pp. 1111- 1116, Seoul, Korea, May 21-26, 2001.

[15] Y. Ikemoto, Y. Hasegawa, T. Fukuda, and K. Matsuda, "Zipping, weaving: control of vehicle group behavior in non-signalized intersection," In proceeding of IEEE International Conference on Robotics and Automation, pp. 4387- 4391, New Orleans, LA, April, 2004.

[16] P. E. Rybski, N. P. Papanikolopoulos, S. A. Stoeter, D. G. Krantz, K. B. Yesin, M. Gini, R. Voyles, D. F. Hougen, B. Nelson and M. D. Erickson, "Enlisting rangers and scouts for reconnaissance and surveillance," In Proceedings of IEEE Robotic and Automation Magazine, vol.7, no. 4, pp. 14-24, 2000.

[17] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer, "Gathering of autonomous mobile robots with limited visibility," In proceeding of the 18th International Symposium on Theoretical Aspects of Computer Science (STACS 2001), LNCS 2010, 2001, pp. 247-258, 2001.

[18] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer, "Distributed coordination of a set of autonomous mobile robots," In Proceding of 18th International Symposium on Theoretical Aspects of Computer Science (STACS 2001), LNCS 2010, pp. 247-258. Dresden, Germania, Feb. 15-17, 2001.

[19] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer, "Distributed coordination of a set of autonomous mobile robots," In Proceeding of IEEE Intelligent Vehicles Symposium (IVS2000), pp. 480-485, Dearborn, Mi, USA, Oct. 1-5, 2000.