2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

Experimental Implementation of Flocking Algorithms in Wheeled Mobile Robots

A. Regmi*, R. Sandoval*, R. Byrne§, H. Tanner#, and C.T. Abdallah*
* Electrical and Computer Engineering Department
MSC01 1100
University of New Mexico
Albuquerque, NM 87131-0001
{aregmi,rsandova,chaouki}@eece.unm.edu

#Mechanical Engineering Department
MSC01 1150
1 University of New Mexico
Albuquerque, NM 87131-0001
tanner@unm.edu

§Intelligent Systems, Sensors, and Controls
Department 15211
Sandia National Labs
Albuquerque, NM 87185
rhbyrne@sandia.gov

Abstract— **In this paper we present an experimental implementation of the flocking algorithm presented in [1] for the double integrator model, in the case of wheeled mobile robots(WMR). We use the look ahead concept to feedback linearize the WMR model and obtain its double integrator form in which we apply the control signals in the form of voltages in wheel motors. A virtual leader is used to steer the flock in the desired direction. Comparisons of the experimental results with simulation verify the robustness of this algorithm to model uncertainties and communication delays.**

I. Introduction

A landmark paper by Reynolds [9] in 1987 attracted a significant interest in studying the flocking behavior of animals in the context of coordination and formation of multiple agents. Inspired by this work, numerous research groups have studied the problem of coordinating the motion of multiple autonomous agents for the possible applications in broad areas like automated highways, satellite formation, coordinated movement of a cluster of vehicles for the search and rescue operation [5], and formation flying of unmanned aerial vehicles (UAVs) for the purposes like surveillance and distribution of sensor networks [7].

The primary objective of the approaches presented in [1], [6], [5] and [7] is to design a local decentralized controller for individual agents so that a global objective for the flock can be achieved. In most of the work done in this area of formation control, the basic objectives of controller design can be summarized as.

- Avoid inter-agent collisions and maintain group cohesion by keeping a desired separation.
- Align to the average group heading.
- Match the velocity with neighbors.

In [1] and [2] , authors take double integrator models and design decentralized control laws based on potential functions for both, fixed and dynamic topology of flocks. The control laws have been extended for nonholonomic agents in [10]. In addition to the potential function, virtual leaders (beacon in [6] and γ-agents in [5]) are used in [6] and [5] to steer the flock in the desired direction.

A control Lyapunov function approach is proposed in [8]. Instead of designing a controller for some standard model, they have addressed the coordination problem for a class of robots for which control Lyapunov function can be found.

Although there has been a lot of research in an effort to understand this problem, practical implementations of such control algorithms have rarely been reported. The work in [7] is motivated by a practical implementation of meter-scale UAV formation and many practical issues are considered in their control laws,yet, a successful implementation of flocking algorithms on UAVs has not been reported.

In this paper we present an implementation of flocking control laws on wheeled mobile robots. In [1], the authors present a class of control laws for fixed topology for a group of double integrator agents such that their velocities converge to the average group velocity, and collisions between the agents are avoided by maintaining the desired separation

0-7803-9098-9/05/$25.00 ©2005 AACC

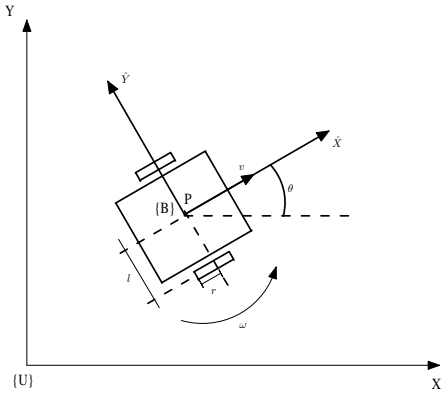

Fig. 1. A schematic of the mobile agent

between all the agents. We introduce the concept of look-ahead to feedback linearize the WMR model in order to obtain it in the double integrator form, and then implement the control law proposed in [1]. We have successfully implemented the coordination algorithm mentioned in [1], in two WMRs and one virtual leader.

The remainder of this paper is organized as follows. In section II, we describe and model the WMR that we are using in the implementation. Feedback linearization and control is presented in section III. Simulation and experimental results are discussed in section IV and conclusions are drawn in section V.

II. Wheeled Mobile Robot Modelling

Advancement in computation and communication technology has opened up a whole new domain of applications for the networked, distributed and decentralized control. In the course of preparation of a test bed for the practical implementation of proposed control strategies in the areas of coordination of multiple agents and networked control systems, we have built two wheeled mobile robots (3 & 4) in the Networked Control Systems Laboratory at the Electrical & Computer Engineering department of the University of New Mexico. These WMRs are actuated by DC motors attached to each of the wheels in a differential steering setting. The DC motors are driven with Advanced Motion Controls®power drivers, and powered by two 12 Volt batteries. Power drivers and other electronic peripherals are interfaced to a laptop computer via a PCMCIA data acquisition card. The displacement and velocity are obtained through high resolution encoders attached to the wheels, and dead reckoning algorithm are used to determine absolute position and orientation. The onboard laptop computer is connected to the 802.11b wireless LAN to communicate with other WMRs.

The kinematic equations of motion for the WMR agent moving in the global coordinate frame U, as shown in Figure 1, are given by,

$$\dot{x} = u\cos\theta \quad (1)$$
$$\dot{y} = u\sin\theta \quad (2)$$
$$\dot{\theta} = \omega \quad (3)$$

where u is the surge speed of a body-fixed frame B attached to the center of mass of the WMR. (x,y) denote the position of the frame B with respect to the global coordinate frame U. The orientation of the frame B is given by the angle θ of $\hat{X}$-axis with respect to the X-axis, and ω is the angular speed. The dynamic equations for the motion of the WMR are given by,

$$m\dot{u} = -\eta_1\dot{x} + (F_R + F_L) \quad (4)$$
$$J\dot{\omega} = -\phi_1\omega + (F_R - F_L)l \quad (5)$$

where m is the mass of the WMR, J is its moment of inertia, F_L and F_R are the forces generated by the left and right wheels respectively , ϕ_1 and η_1 are the coefficients of rotational and viscous frictions respectively and l denotes the moment arm of the forces where the center of geometry and center of mass in the WMR are assumed to coincide. Differentiating (1) and (2), yields

$$\ddot{x} = \dot{u}\cos\theta - u\sin\theta\,\dot{\theta} \quad (6)$$
$$\ddot{y} = \dot{u}\sin\theta + u\cos\theta\,\dot{\theta} \quad (7)$$

Substituting $\dot{u}$ from (4) and ω from (5) gives a complete model of the vehicle in the global coordinate frame.

$$m\ddot{x} = -\eta_1\dot{x} - \dot{y}\,\dot{\theta} + (F_R + F_L)\cos\theta \quad (8)$$
$$m\ddot{y} = -\eta_1\dot{y} + \dot{x}\,\dot{\theta} + (F_R + F_L)\sin\theta \quad (9)$$
$$J\ddot{\theta} = -\phi_1\dot{\theta} + (F_R - F_L)l \quad (10)$$

Since we do not have access to the force inputs, we would like to express the model so that the voltage applied to the left and right wheels appears as control inputs. We know that the force, F generated by a DC motor connected to a wheel with radius r when supplied with a voltage of V is

$$F = K\,V - \eta_2\nu$$

where, $K = \frac{K_T}{r}$ and $\eta_2 = \frac{K_T}{r}\frac{K_b}{R}$, K_T is motor torque constant, K_b is the back emf constant, R is the armature resistance and ν is the linear speed of the wheel. Now, the forces generated by left and right wheels when supplied with voltages V_R and V_L are,

$$F_L = KV_L - \eta_2\nu_L \quad (11)$$
$$F_R = KV_R - \eta_2\nu_R \quad (12)$$

where ν_L and ν_R are the linear speed of the left and right wheel respectively. Combining (8), (11) , (12), we obtain,

$$m\ddot{x} = -\eta_1\dot{x} - \dot{y}\,\dot{\theta} - \eta_2(\nu_R + \nu_L)\cos\theta + K(V_R + V_L)\cos\theta.$$

Define $\eta_1 + 2\,\eta_2 = \eta$, and since, $\frac{(\nu_L + \nu_R)}{2}\cos\theta = \dot{x}$

$$m\ddot{x} = -\eta\dot{x} - \dot{y}\,\dot{\theta} + K(V_R + V_L)\cos\theta. \quad (13)$$

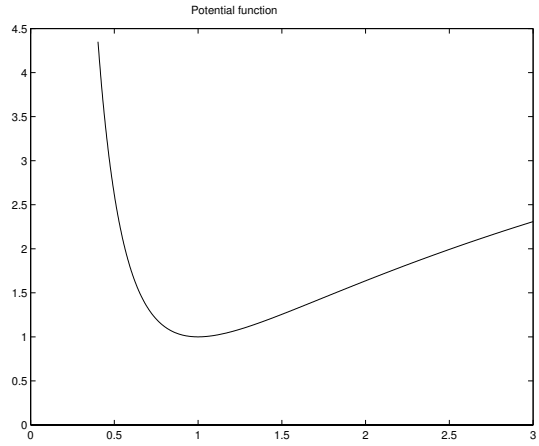

Fig. 2. A typical potential function

Similarly, for $\ddot{y}$ we obtain

$$m\ddot{y} = -\eta\dot{y} + \dot{x}\,\dot{\theta} + K(V_R + V_L)\sin\theta. \tag{14}$$

Combining (10), (11) and (12) we obtain,

$$J\ddot{\theta} = -\phi_1\dot{\theta} + K(V_R - V_L)l - \eta_2(\nu_R - \nu_L)l$$

Substituting $\dot{\theta} = \frac{\nu_R - \nu_L}{2l}$ and $\phi = 2l^2\eta_1 + \phi_1$ gives,

$$J\ddot{\theta} = -\phi\dot{\theta} + K(V_R - V_L)l \tag{15}$$

We finally re-write equations (13),(14), and (15) to obtain the dynamical model used for control design

$$\begin{aligned} m\ddot{x} &= -\eta\dot{x} - \dot{y}\,\dot{\theta} + K(V_R + V_L)\cos\theta & (16)\\ m\ddot{y} &= -\eta\dot{y} + \dot{x}\,\dot{\theta} + K(V_R + V_L)\sin\theta & (17)\\ J\ddot{\theta} &= -\phi\dot{\theta} + K(V_R - V_L) & (18) \end{aligned}$$

III. Coordination Strategy

In [1], the authors propose a decentralized coordination scheme based on virtual potentials to achieve flocking behavior in a group of multiple agents. In this approach, agents receive information (velocity and position) from other agents in their neighborhood, and based on this information all the agents are made to converge to a common group velocity, while keeping the desired separation between the agents.

In our work, we implemented coordination for three agents, composed of one virtual agent and two real agents. The dynamic equations of the real agents are given by (16), (17), (18) while the leader has simple dynamics of the form,

$$\dot{x} = U_x \tag{19}$$

$$\dot{y} = U_y \tag{20}$$

We opted to use a virtual leader because uncertainties in the system model and parameters may cause the group velocity to converge to zero or to saturate the control input. Hence, a virtual leader serves the purpose of providing a reference velocity and heading to which the group velocity and heading will converge, while keeping the control inputs from saturating. For a double integrator model of the agents, the control inputs are accelerations in both, x and y directions. For coordination, the feedback control law has the following form. [1],

$$u_i = k_1\,\alpha_i + k_2\,a_i \quad i = 1, 2..n-1 \tag{21}$$

where, k_1 and k_2 are the gains introduced to avoid saturating the control voltages. α_i in (21) causes the velocity vectors of all agents to align in the same direction with the same magnitude. a_i, which is a vector in the negative gradient of an artificial potential function U_i, contributes to collision avoidance and cohesion in the group. The acceleration inputs for agent i are defined as:

$$\begin{bmatrix} u_{ix} \\ u_{iy} \end{bmatrix} = \begin{bmatrix} -k_1\sum_{j\in N_i}(v_{ix} - v_{jx}) - k_2\sum_{j\in N_i}\nabla_{r_{ix}}U_{ij} \\ -k_1\sum_{j\in N_i}(v_{iy} - v_{jy}) - k_2\sum_{j\in N_i}\nabla_{r_{iy}}U_{ij} \end{bmatrix}$$

where U_{ij} is an artificial potential function and is a function of the distance $||r_{ij}||$ between agent i and its neighbor j. N_i is a set which contains all the neighbors of the agent i. Neighbors of an agent i are those agents with whom it can communicate. The potential U_{ij} is a differentiable, nonnegative, radially unbounded function of the distance r_{ij} between agents i and j, such that [1]

1) $U_{ij}(r_{ij}) \to \infty$ as $||r_{ij}|| \to 0$
2) U_{ij} attains its unique minimum when agents i and j have the desired separation.

A typical potential function is shown in Figure 2. We choose the following potential function candidate,

$$U_{ij} = \frac{1}{\left(\frac{||r_{ij}||}{d}\right)^2} + \log\left(\frac{||r_{ij}||}{d}\right)^2 \tag{22}$$

where, $||r_{ij}|| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$. Differentiating U_{ij} with respect to x_i and y_i, we obtain

$$\begin{aligned} \nabla_{r_{ix}}U_{ij} = \frac{\partial U_{ij}}{\partial x_i} &= \frac{-2(x_i - x_j)d^2}{\left((x_i - x_j)^2 + (y_i - y_j)^2\right)^2} \\ &\quad + \frac{2(x_i - x_j)}{(x_i - x_j)^2 + (y_i - y_j)^2} \\ \nabla_{r_{iy}}U_{ij} = \frac{\partial U_{ij}}{\partial y_i} &= \frac{-2(y_i - y_j)d^2}{\left((x_i - x_j)^2 + (y_i - y_j)^2\right)^2} \\ &\quad + \frac{2(y_i - y_j)}{(x_i - x_j)^2 + (y_i - y_j)^2} \end{aligned}$$

Next, we feedback-linearize the vehicle model (16)-(18) to transfer it to the double integrator form so that we can substiture the acceleration inputs. Let us choose the following two outputs,

$$y_1 = x + L\,\cos\theta \tag{23}$$

$$y_2 = y + L\,\sin\theta \tag{24}$$

Fig. 3. Lobot

where L is the look-ahead distance. The look-ahead concept consists of a point along the $\hat{X}$-axis of the WMR as the output instead of center of mass itself. This slight perturbation helps us in obtaining controllable linearization. Differentiating y_1 and y_2, we obtain,

$$\begin{bmatrix} \dot{y_1} \\ \dot{y_2} \end{bmatrix} = \begin{bmatrix} \dot{x} - L\,\sin\theta\,\dot{\theta} \\ \dot{y} + L\,\cos\theta\,\dot{\theta} \end{bmatrix}.$$

Differentiating again and substituting for $\ddot{x}$, $\ddot{y}$ and $\ddot{\theta}$, we obtain

$$\begin{bmatrix} \ddot{y_1} \\ \ddot{y_2} \end{bmatrix} = \begin{bmatrix} u_x \\ u_y \end{bmatrix} = A \begin{bmatrix} V_R \\ V_L \end{bmatrix} + B$$

where,

$$A = \begin{bmatrix} \frac{K}{m}\cos\theta - \frac{L}{J}Kr\sin\theta & \frac{K}{m}\cos\theta + \frac{L}{J}Kr\sin\theta \\ \frac{K}{m}\sin\theta + \frac{L}{J}Kr\cos\theta & \frac{K}{m}\sin\theta - \frac{L}{J}Kr\cos\theta \end{bmatrix}$$

$$B = \begin{bmatrix} -\frac{\eta}{m}\dot{x} - \dot{y}\dot{\theta} - L\,\cos\theta\,\dot{\theta}^2 + \frac{L\phi}{J}\sin\theta\,\dot{\theta} \\ -\frac{\eta}{m}\dot{y} + \dot{x}\dot{\theta} - L\,\sin\theta\,\dot{\theta}^2 - \frac{L\phi}{J}\cos\theta\,\dot{\theta} \end{bmatrix}.$$

Note that A is invertible. Hence, we can design

$$\begin{bmatrix} V_R \\ V_L \end{bmatrix} = A^{-1}\left(-B + \begin{bmatrix} u_x \\ u_y \end{bmatrix}\right)$$

IV. Simulation and Experiment

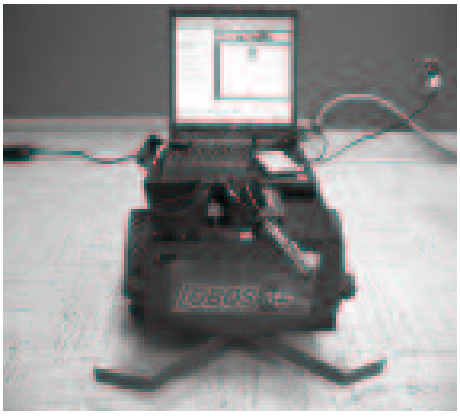

Fig. 4. Pino

	m	J	K	η	ϕ	r
Pino	25	1.1094	3.7054	178.57	7.2817	0.203
Lobot	27	1.445	9.5063	225	12.04	0.23

TABLE I
Physical Parameters of WMRs

We implemented the coordination scheme for two WMRs, Lobot (Figure 3) and Pino (Figure 4), and one virtual leader. The physical parameters of the WMRs are listed in Table I, where m is in Kg, J is in Kgm^2, K is in $\frac{N}{volts}$, η is in $\frac{Kg}{sec}$, ϕ is in $\frac{Nmsec}{rad}$, and r is in m. We define the control laws in continuous-time, however, we implemented it in a discrete-time fashion. We implemented the controller in LabView^{TM} and because of the computation delay we could manage to sample only as fast as 200msec. So, at the interval of 200msec, Pino and Lobot update their position, and velocity using odometry and send it to the respective neighbor.The virtual leader is simulated in LabView^{TM} and it also sends its position and velocity to Pino and Lobot at the same time interval. Communication between the agents is done using UDP across the WLAN. In each agent, a separate thread of receiver is running continuously and it updates the received data every time new data arrives. Control calculation is done using the latest information in receiver.

The initial condition for Pino is: surge speed 0.05 m/sec, angular speed 0.0rad/sec and orientation 35^0. The initial condition for Lobot is: surge speed 0.11 m/sec, angular speed 0.0 rad/sec, and orientation 55^0. The virtual leader is simulated to have surge speed 0.113m/sec, angular velocity 0.0 rad/sec and orientation 45^0. At the start of the experiment, the separation between Pino and Lobot (d_{LP}) is 1.59m , between Pino and the Leader (d_{LeP}) is 1.125, and between Lobot and the Leader (d_{LeL})is 1.125. The desired separation between each of them is 1m. After implementing the coordination algorithm, the surge velocities of Pino and Lobot converge to 0.113 m/sec, orientation of them converge to 45^0, and separation between all of them converges to the desired separation of 1m as shown in Figures (5, 7, 9, 11).

Simulation results for the same initial condition and sampling time are presented along with the experimental results (Figures 6, 8, 10, 12). Comparing the simulation and the experimental results, we can observe that the transient behavior of the surge velocity (Figure 5) in experiment is different than that of the simulation (Figure 6). This is because of the uncertainties involved in the modelling of the WMR and inaccurate system parameters. The varying nature of transmission and computation delay may also have contributed in the differences between the experimental and the simulation behavior. In the transient of the experiment, velocities of both, Pino and Lobot, decreases to almost zero. In this situation, if we had not used the virtual leader, both of them would have stopped and it would have looked

like they converged to a zero group velocity. The virtual leader provided a reference group objective and even in the presence of uncertainties, directed the group to flock.

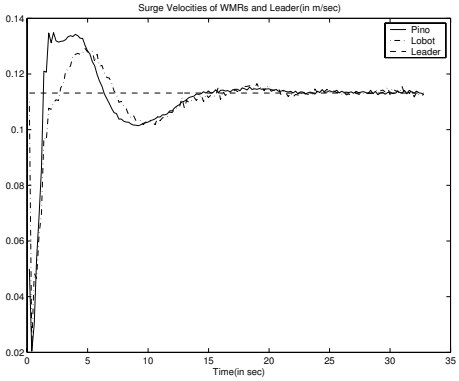

Fig. 5. Surge Velocities (Experiment)

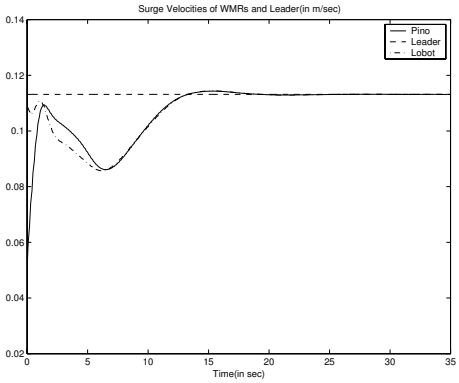

Fig. 6. Surge Velocities (Simulation)

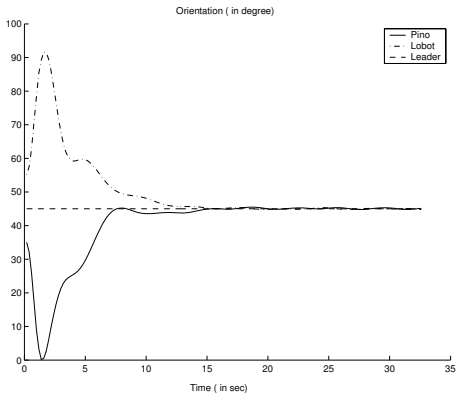

Fig. 7. Orientation (Experiment)

V. Conclusions

In this paper we presented an experimental implementation of flocking in wheeled mobile robots. We implemented control laws presented in [1] for the double integrator model

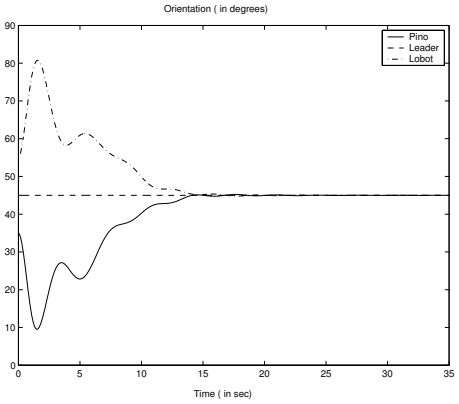

Fig. 8. Orientation (Simulation)

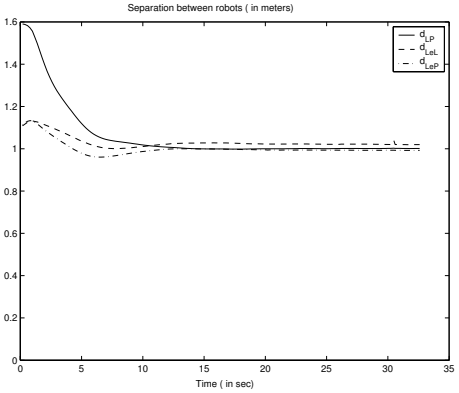

Fig. 9. Separation between the agents (Experiment)

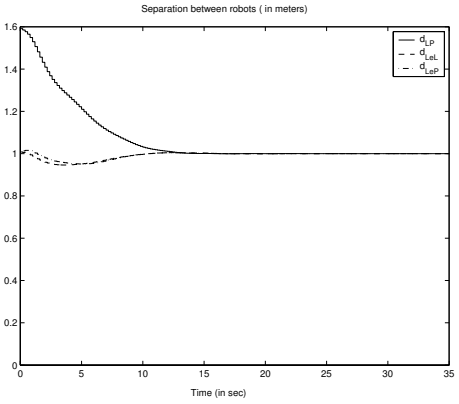

Fig. 10. Separation between the agents (Simulation)

in WMRs. We employed feedback linearization with look ahead to represent the WMR model in normal form and computed the control inputs for coordination. We introduced a virtual leader to steer the flock in the desired direction. Simulation and experimental results were presented. We observed a slight discrepancy between the simulation and experimental results, which we attribute to the uncertainties in the modelling of the robots and inconsistent computation

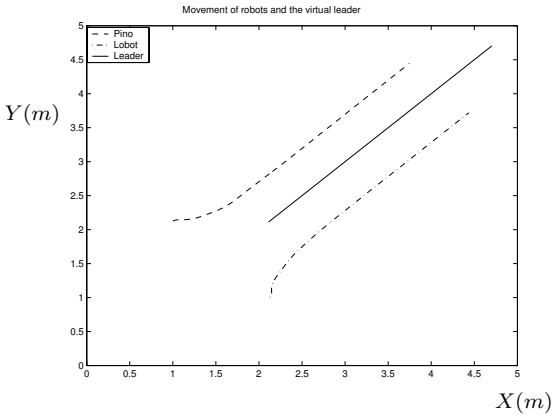

Fig. 11. Movement of the WMRs and the leader (Experiment)

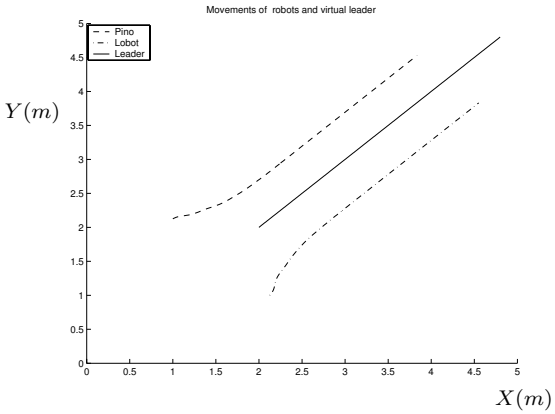

Fig. 12. Movement of the WMRs and the leader (Simulation)

time and network delays.

REFERENCES

[1] H.G. Tanner, A. Jadbabaie and G. J. Pappas. Stable flocking of mobile agents, Part I: Fixed Topology.42nd IEEE Conference on Decision and Control, Maui Hawaii, December 2003, pp. 2010-2015

[2] H.G. Tanner, A. Jadbabaie and G. J. Pappas. Stable flocking of mobile agents, Part II: Dynamic Topology.42th IEEE Conference on Decision and Control, Maui Hawaii, December 2003, pp. 2016-2021

[3] X. Yun and Y. Yamamoto. Internal dynamics of a wheeled mobile robot. Proceeding of IEEE/RSJ International Conference on Intelligent Robots and Systems, Japan, 1993.

[4] L. Cremean, W. B. Dunbar, D. V. Gogh, J. Hickey, E. Klavins, J. Meltzer, and R. M. Murray.The Caltech multi-vehicle wireless testbed. Conference on Decision and Control (CDC), 2002.

[5] R. Olfati and R. M. Murray. Flocking for multi-agent dynamic systems: algorithms and theory. Technical Report CIT-CDS 2004-005.

[6] N. E. Leonard and E. Fiorelli. Virtual leaders, artificial potentials, and coordinated control of groups. Proc. of the 40th IEEE Conference on Decision and Control, Orlando, FL, Dec. 2001.

[7] E. W. Justh and P. S. Krishnaprasad.Institute for Systems Research Technical Report 2002-38, 2002.

[8] P. Ögren, M. Egerstedt, X. Hu. A control Lyapunov function appraoch to muliagent coordination.. IEEE Transactions on Robotics and Automation, Vol. 18, No. 5, pp. 847-851, Oct. 2002.

[9] C. Reynolds. Flocks, birds and schools: a distributed behavioral model. Computer Graphics, 21:25-34, 1987.

[10] H.G. Tanner, A. Jadbabaie and G. J. Pappas. Flocks of autonomous mobile agents.Second Annual Symposium on Autonomous Intelligent Networks and Systems, Menlo Park CA, June 30-July 1, 2003.

[11] H. K. Khalil. Nonlinear systems, Third edition, Prentice Hall, 2002.