7-1-2007

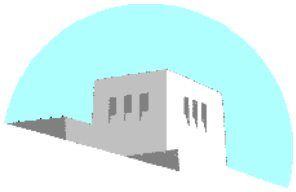
Leader-Follower Control with Odometry Error Analysis

Jorge Piovesan

Chaouki Abdallah

Herbert Tanner

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING



SCHOOL OF ENGINEERING

UNIVERSITY OF NEW MEXICO

Leader-Follower Control with Odometry Error Analysis

Jorge L. Piovesan, Chaouki T. Abdallah, and Herbert G. Tanner [1] [2]

UNM Technical Report: EECE-TR-07-002

Report Date: April 3, 2007

[1]J. Piovesan and C. Abdallah are with the Department of Electrical and Computer Engineering of the University of New Mexico, Albuquerque NM 87131, (jlpiovesan,chaouki)@ece.unm.edu. H. Tanner is with the Department of Mechanical Engineering of the University of New Mexico, Albuquerque NM 87131, tanner@unm.edu.

[2]The work of J. Piovesan and C. Abdallah is partially supported by NSF award #0233205

Abstract

In this paper we present a leader-follower control law that enables a mobile robot to track a desired trajectory, and allows us to specify the position in the plane of the follower robot with respect to the leader robot. We first describe the dynamic model of the plant, including input torques, and friction forces. Then the control law is developed using backstepping, and it is proved to asymptotically stabilize the tracking error to the origin. Simulation and experimental results of the closed loop system are presented, highlighting its potential application to formation control. The special case of pure tracking (without bi-dimensional position information use) is analyzed, showing that it can be applied to particular classes of non-feasible trajectories. Finally, motivated by some observations on the experiments, the effects of odometry errors are analyzed, revealing that boundedness of the tracking errors can be guaranteed if absolute position information becomes available periodically.

Keywords

Mobile robots, leader-follower control, backstepping, odometry.

1 Introduction

Trajectory tracking for mobile robots has been extensively treated in robotics and controls literature, both at the kinematic [1–6] and the dynamic level [7–13]. Tracking is about controlling a mobile robot such that it can follow a desired spacial trajectory in real time, with the least possible error. A different but related problem is that of formation control [14–34]. We refer to the problem where mobile agents have to fall into formation while navigating in a known area as "navigation in formation." Tracking and navigation in formation are related problems, since in both cases the desired motion can be described in the form of reference trajectories. Thus, navigation can be achieved through tracking. Examples of this approach appear in literature classified as leader-following [14, 19, 20, 22, 29, 32].

Trajectory tracking has been addressed mostly from a model-based perspective. Authors usually derive the control algorithm based on a model of the plant. Depending on the type of model, one can classify tracking controllers either as kinematic [1–6], or dynamic [7–13]. In terms of design techniques, feedback-linearization [6,12], LQR [5], backstepping [1, 7, 10, 11], and sliding mode control [8, 9] are commonly used. Most of the available results [1, 2, 4–13] assume that the desired trajectory is feasible, i.e, it can be exactly reproduced by the model of the plant. In [3], such assumption is not made and this problem is solved by including an estimator that approximates the non-feasible velocities of the target (as referred in [3]) with feasible ones. Once these approximations are made, the control inputs (linear and angular velocities) are driven to match the desired velocities given by the estimator.

The navigation in formation problem has been approached from several different perspectives, among them being leader-follower [14,19,20,22,29,32], flocking/swarming [18,21,23–28], model independent formation [17], virtual structure [31], Internet Like Protocol [30, 35, 36], distributed receding horizon control [33], and control Lyapunov function approaches [34]. Most of these results concentrate on the interactions between the agents and the environment, assuming simple models for each individual agent. For example, most of flocking/swarming algorithms assume a double integrator model [18, 21, 23–28]. Other simplifying assumptions include the ability of the agent to track any desired trajectory planned for the formation [17], or its holonomic nature [20]. Less restrictive assumptions are made in approaches where a kinematic model (with [31] or without [29] a dynamic extension) is used.

We exploit the relationship between trajectory tracking and navigation in formation. We use the classical model-based tracking approach to derive a modified tracking law, and then adapt it to formation control. The result is a two-dimensional, asymptotically stable, tracking law based on a model for a unicycle robot, that respects the nonholonomic constraints, includes friction, and is implemented by means of input torques. We show in simulations and experimental studies that this tracking law can be applied to formation control, and discuss the technical details of such implementation. Motivated by experimental observations, we prove boundedness of the tracking error under bounded odometry errors. Moreover, the special case of tracking is shown to be asymptotically stable for a special class of non-feasible reference trajectories.

This paper makes contributions both in multiagent coordination and tracking: regarding tracking, we develop a tracking law, which is similar to previous works [7, 11, 37], but includes the possibility of tracking non-feasible trajectories; regarding formation control, we address issues about the model and the odometry of the agents engaging in formation tasks, and demonstrate that model selection is an important part of the coordination algorithm. However, since we do not address issues related to the interaction between agents, we prefer not to view this work as a pure formation control approach.

The reminder of the paper is organized as follows. Section 2 describes the robots and their model. Section 3 presents the development of the leader follower control law, and addresses the special case of tracking non-feasible trajectories. In sections 4 and 5 simulations and experimental results are presented respectively. Section 6 analyzes the effects of odometry errors. Our conclusions are summarized in section 7.

2 Experimental Testbed and Modeling

The mobile platforms on which our algorithms are implemented are unicycle-type nonholonomic mobile robots constructed in a box-shaped frame, with two independently steered wheels powered by DC Pancake-Type motors, and one (or two) low friction omnidirectional caster(s) located below the frame to stabilize the robot horizontally (Figure 1). The robots carry laptops on which the control inputs are produced, through National Instruments® Data Acquisition Cards.

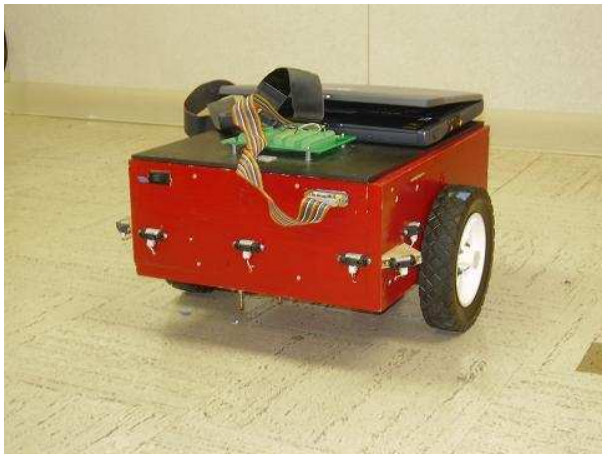


Figure 1: One of the mobile robots used for the experimental testing of the leader-follower control law.

The kinematic and dynamic model of a mobile robot is developed using a global coordinate frame $\{U\}$, and a body-fixed coordinate frame $\{B\}$ as shown in Figure 2.

Assuming that the center of the wheels' axis of our differential-type mobile robot coincides with origin of $\{B\}$, the kinematics of the latter can be expressed using the unicycle equations:

$$\dot{x} = v\cos\theta \tag{1a}$$

$$\dot{y} = v\sin\theta \tag{1b}$$

$$\dot{\theta} = \omega, \tag{1c}$$

where x and y are the position coordinates of the origin of $\{B\}$ in $\{U\}$, θ is the orientation of the robot in $\{U\}$, and v and ω are the linear and rotational speeds of the robot in the body-fixed coordinate frame.

The dynamics of the robot model can be formulated in reference to $\{B\}$. Under the assumption that friction can be modeled as a combination of viscous forces and rotational torques, the dynamic equations can be written as:

$$m\dot{v} = -\eta v + \frac{1}{r}(\tau_r + \tau_l) \tag{2a}$$

$$J\dot{\omega} = -\psi\omega + \frac{l}{r}(\tau_r - \tau_l), \tag{2b}$$

where m is the mass of the robot, J is the inertial moment, r is the radius of the driving wheels, l is half of the distance between the driving wheels, τ_r and τ_l are the torques of the motors attached to the left and right wheel respectively, and η and ψ are the viscous and rotational friction coefficients, respectively.

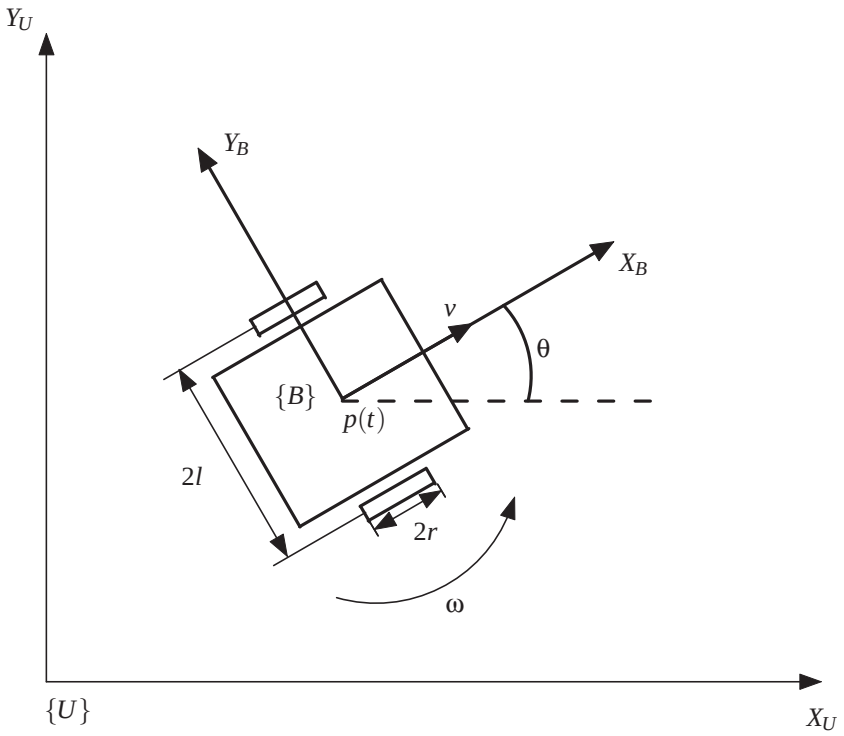


Figure 2: Schematic of a unicycle-type mobile robot, and the coordinate frames considered.

The model can be expressed in a more compact form:

$$\dot{p} = \begin{bmatrix}\dot{x}\\ \dot{y}\end{bmatrix} = v\begin{bmatrix}\cos\theta\\ \sin\theta\end{bmatrix} \tag{3a}$$

$$\dot{\theta} = \omega \tag{3b}$$

$$m\dot{v} = -\eta v + u_1 \tag{3c}$$

$$J\dot{\omega} = -\psi\omega + u_2, \tag{3d}$$

where p is $\begin{bmatrix}x & y\end{bmatrix}^T$, and the control inputs u_1 and u_2 are equal to $\frac{1}{r}(\tau_r+\tau_l)$ and $\frac{l}{r}(\tau_r-\tau_l)$, respectively.

For the two robots used in the experiments the model parameters that have been identified are summarized in Table I.

Parameter	Agent 1	Agent 2	Units
m	27	25	Kg
η	180	133.7	Kg/sec
J	1.42	1.03	Kg m
ψ	9.47	5.51	Kg m/sec
l	0.23	0.203	m
r	0.101	0.10	m

Table I: Mobile robots dynamic model parameters.

3 Development of the Leader-Follower Control Law

The control objective is to have a follower agent track the leader while maintaining a fixed relative (instantaneous) position with respect with the leader's trajectory. The trajectories should be "parallel," in the sense that one can be obtained from the other by translation. Our working assumption is that arbitrarily small segments of the leader's smooth trajectory can be approximated as a circular section of a given radius (infinite radius for straight lines). If such trajectory is time varying, a smoothness assumption guarantees that the time varying radius of curvature has no discontinuities over time. Let us define the following terms before formally stating the leader-follower control problem:

Definition 1 *The terms L_X, L_Y, L_{Xoff}, L, L_{Yoff}, and R_C are defined as follows:*

- $\begin{bmatrix} L_X & L_Y \end{bmatrix}^T$ *is the desired* $\begin{bmatrix} x & y \end{bmatrix}^T$ *position of the leader in the followers body fixed frame $\{B\}$ (Figure 3);*
- $L_{Xoff} \geq 0$ *is a control parameter used to specify the desired separation between the leader and the follower, measured in the same direction as L_X;*
- $L > 0$ *is an arbitrarily small control parameter, measured in the same direction as L_X, such that $L_X = L + L_{Xoff}$ and $L_X \approx L_{Xoff}$ (implying $L_X > 0$).*
- L_{Yoff} *is the translation along L_Y direction that the leader's trajectory has to sustain to produce the desired trajectory for the follower. It can be though of as the distance between reference trajectories (Figure 4).*
- R_C *is the radius of curvature of the segment of the trajectory followed by the leader. $R_C \triangleq V_d/\omega_d$ where V_d and ω_d are the leader's linear and angular speeds (Figure 4). Note that R_C can be infinite for a straight line-trajectory ($\omega_d = 0$) without causing problems in controller implementation.*

Since our objective is to have the agents follow "parallel" smooth trajectories, the distance L_{Ycomp} is dependent on the curvature of the trajectory followed by the leader, as shown in Figure 4, and is thus a time-varying parameter to be determined. From Figure 4 it is also evident that $L_Y = L_{Yoff} + L_{Ycomp}$, implying that L_Y is time varying too.

From Figure 4 (using geometric arguments) it follows that,

$$L_{Ycomp} = R_C - \sqrt{R_C^2 - L_{Xoff}^2}. \tag{4}$$

The validity of this equation for all cases requires some additional assumptions:

First let us assume that the leader robot always moves forward, that is, $V_d \geq 0$. Under this assumption, R_C can be positive or negative depending on the sign of the angular speed of the leader. This modifies (4) as follows:

$$L_{Ycomp} = \text{sign}(R_C)\big(|R_C| - \sqrt{R_C^2 - L_{Xoff}^2}\big).$$

Since L_{Ycomp} has to be real, we need $R_C \geq L_{Xoff}$, that is, the desired distance L_{Xoff} can never be larger than the curvature of the leader's trajectory. So the sharper the leader turns, the closer the leader must be. Then to avoid obtaining complex results in the case where $R_C < L_{Xoff}$, we truncate the term $\sqrt{R_C^2 - L_{Xoff}^2}$, leaving $L_{Ycomp} = R_C$. For $\omega_d = 0$ or $R_C = \infty$ (the case of a straight line trajectory) L_{Ycomp} is reduced to *zero*. Thus, L_{Ycomp} is finally given with reference to Figure 5 as:

$$L_{Ycomp} = \begin{cases} \text{sign}(R_C)\big(|R_C| - \sqrt{R_C^2 - L_{Xoff}^2}\big), & |R_C| \geq L_{Xoff}, \omega_d \neq 0, \\ R_C, & |R_C| < L_{Xoff}, \omega_d \neq 0, \\ 0, & \omega_d = 0. \end{cases} \tag{5}$$

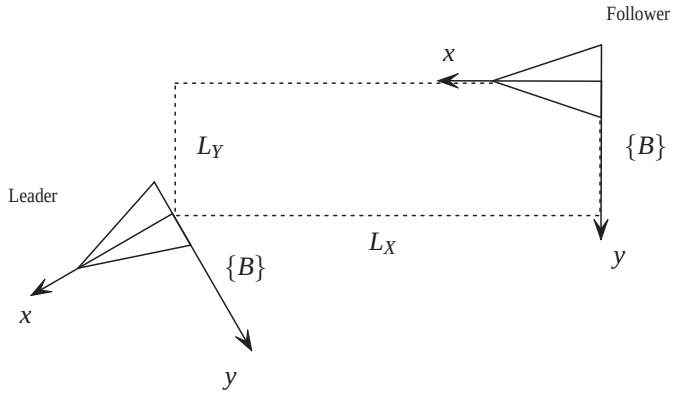


Figure 3: Graphical representation of L_X and L_Y.

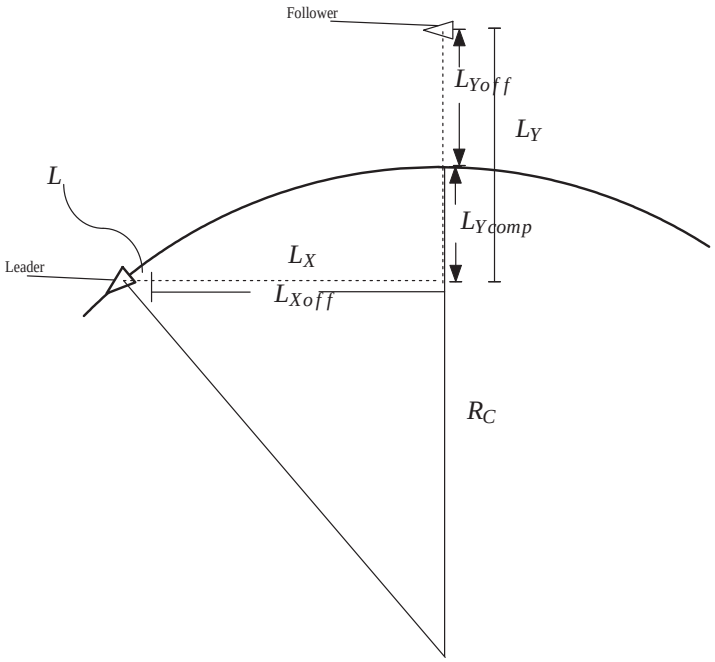


Figure 4: Graphical representation of L_{Xoff}, L, L_{Yoff}, L_{Ycomp} and R_C with respect to leader and follower body coordinate frames.

By definition, L_{Ycomp} is non-smooth, however it can be reasonably approximated by the smooth function (Figure 5):

$$L_{Ycomp} \approx 2G(2\arctan(\frac{R_C}{\delta}) - \arctan(\frac{R_C-\varepsilon}{\delta}) - \arctan(\frac{R_C+\varepsilon}{\delta}) \quad (6)$$

where G, δ and ε are smooth functions of L_{Xoff}, estimated via calibration and basic curve statistical fitting. This smooth approximation of equation (5) is necessary because we will subsequently assume that L_Y is smooth.

Remark 1 *The term L_{Ycomp} is negligible in cases where $R_C >> L_{Xoff}$ (which covers the majority of the practical cases). The significance of L_{Ycomp} is that its inclusion in the control law guarantees that the formation will be maintained (at the expense of large control signals) even in cases where R_C is not significantly larger than L_{Xoff}, and that stability is guaranteed even when $R_C < L_{Xoff}$.*

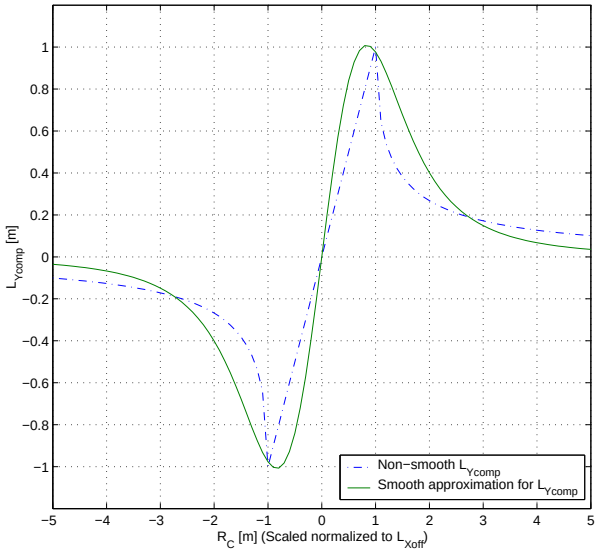


Figure 5: Function L_{Ycomp} and its smooth approximation.

Definition 2 *Define e as* $e = p(t) - p_d(t) + R^T \begin{bmatrix} L_X & L_Y \end{bmatrix}^T$, *where* $p(t)$ *and* $p_d(t)$ *are the current positions of the follower and the leader robots in* $\{U\}$, *and* $R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$ *is a rotational matrix that maps an angular position in* $\{U\}$, *to the follower's* $\{B\}$.

Remark 2 *By driving e to zero, we force the follower robot to track the trajectory of the leader robot from a specific relative position* $\begin{bmatrix} L_X & L_Y \end{bmatrix}^T$.

The leader-follower control problem is the following: *Given a sufficiently smooth time-varying leader's trajectory* $p_d(t) : [0,\infty) \longrightarrow \mathbb{R}^2$, *and a smooth desired separation between* $p(t)$ *and* $p_d(t)$, $\begin{bmatrix} L_X & L_Y \end{bmatrix}^T$, *find* τ_r *and* τ_l *such that the tracking error e converges to a neighborhood of the origin that can be made arbitrarily small.*

Let us express the error of Definition 2 in the body frame of the follower robot:

$$e = R(p - p_d) + \begin{bmatrix} L_X \\ L_Y \end{bmatrix}. \tag{7}$$

Differentiating, we obtain the error dynamics:

$$\dot{e} = -\begin{bmatrix} 0 & -\omega \\ \omega & 0 \end{bmatrix} e + \begin{bmatrix} -L_Y\omega + v \\ L_X\omega \end{bmatrix} - R\dot{p}_d + \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix}. \tag{8}$$

Consider the Lyapunov function candidate $V_1 = \frac{1}{2}e^T e$, the derivative of which is:

$$\dot{V}_1 = e^T \left(\begin{bmatrix} -L_Y\omega + v \\ L_X\omega \end{bmatrix} - R\dot{p}_d + \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix} \right) \tag{9}$$

We define a virtual control input $\begin{bmatrix} -L_Y\omega + v & L_X\omega \end{bmatrix}$ (in the spirit of integrator backstepping), and define a new error z:

$$z = \begin{bmatrix} -L_Y\omega + v \\ L_X\omega \end{bmatrix} - \left(R\dot{p}_d - \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix} - K_e e \right), \tag{10}$$

and substituting in $\dot{V}_1$ leads to $\dot{V}_1 = -K_e\|e\|^2 + e^T z$. The derivative of z is

$$\dot{z} = \begin{bmatrix} -\dot{L_Y}\omega - \frac{L_Y}{J}(-\psi\omega + u_2) + \frac{1}{m}(-\eta v + u_1) \\ \frac{L_X}{J}(-\psi\omega + u_2) \end{bmatrix} + SR\dot{p}_d - R\ddot{p}_d + K_e\left(-Se + \begin{bmatrix} -L_Y\omega + v \\ L_X\omega \end{bmatrix} - R\dot{p}_d + \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix} \right) + \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix},$$

where $S = \left[\begin{smallmatrix} 0 & -\omega \\ \omega & 0 \end{smallmatrix}\right]$. We define a new Lyapunov function candidate: $V_2 = V_1 + \frac{1}{2} z^T z$, the derivative of which is:

$$\dot{V}_2 = -K_e \|e\|^2 + z^T \left(h + \begin{bmatrix} -\frac{L_Y}{J} u_2 + \frac{1}{m} u_1 \\ \frac{L_X}{J} u_2 \end{bmatrix} \right), \tag{11}$$

with h defined as

$$h = \begin{bmatrix} -\dot{L}_Y \omega + \frac{L_Y}{J} \psi\omega - \frac{\eta}{m} v \\ -\frac{L_X}{J} \psi\omega \end{bmatrix} + SR\dot{p}_d - R\ddot{p}_d + \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix} + K_e \Big(- Se + \begin{bmatrix} -L_Y \omega + v \\ L_X \omega \end{bmatrix} - R\dot{p}_d + \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix} \Big). \tag{12}$$

Based on (11), we define the control signals u_1 and u_2:

$$u_1 = m \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}^T (-h - K_z z) + \frac{L_Y}{L_X} \begin{pmatrix} 0 \\ 1 \end{pmatrix}^T (-h - K_z z) \right\} \tag{13a}$$

$$u_2 = \frac{J}{L_X} \begin{pmatrix} 0 \\ 1 \end{pmatrix}^T (-h - K_z z), \tag{13b}$$

making $\dot{V}_2$ equal to $-K_e \|e\|^2 - K_z \|z\|^2 < 0$ if $K_e, K_z > 0$. We thus have the following result:

Theorem 1 *In the closed loop system* (3) *and* (13)*, with h given by* (12)*, z by* (10)*, and* $K_e, K_z > 0$*, the error e converges asymptotically to zero.*

Remark 3 *The leader-follower control law, depends on the first and second derivatives of* L_Y*, which are a function of* L_{Ycomp}*. The inclusion of these parameters helps the control law guarantee the convergence of the error to zero even in extreme cases when the trajectory exhibits high curvatures at the expense of making the control law more complex. However if* $L_{Xoff} < 1/4R_C$ *it is possible to neglect* $\dot{L}_Y$ *and* $\ddot{L}_Y$ *in the control law calculation, keeping only* L_Y*, (Figure 5).*

This leader-follower control law could serve as an starting point towards a formation control law. This may be done using the leader-follower pair as a building block to define a larger formation, in which a single leader can steer a group of *n* robots. Scaling up the formation would then be possible, once string/mesh stability conditions [32] are satisfied, or that errors fall within the acceptable bounds given by a Leader to Formation Stability analysis [19].

A special case of our leader-follower control law is the pure trajectory tracking one. In this case the control problem is formulated as: *Given a sufficiently smooth time-varying desired trajectory* $p_d(t) : [0, \infty) \longrightarrow \mathbb{R}^2$*, find* τ_r *and* τ_l *such that the tracking error* $e = p - p_d$ *converges to a neighborhood of the origin that can be made arbitrarily small.*

Under these conditions, $L_X = L$ and $L_Y = 0$. Then a similar analysis to the one described in (7)–(13) leads us to the control law

$$u_1 = m \begin{bmatrix} 1 & 0 \end{bmatrix} (-h - K_z z) \tag{14a}$$

$$u_2 = \frac{J}{L} \begin{bmatrix} 0 & 1 \end{bmatrix} (-h - K_z z), \tag{14b}$$

where

$$z = \begin{bmatrix} v \\ L\omega \end{bmatrix} - \left(R\dot{p}_d - K_e e \right) \tag{15}$$

and

$$h = \begin{bmatrix} \frac{\eta}{m} v \\ -\frac{L}{J} \psi\omega \end{bmatrix} + SR\dot{p}_d - R\ddot{p}_d + K_e \left(- Se + \begin{bmatrix} v \\ L\omega \end{bmatrix} - R\dot{p}_d \right). \tag{16}$$

establishing the following result:

Corollary 1 *In the closed loop system* (3) *and* (14)*, with h given by* (16)*, z by* (15)*, and* $K_e, K_z > 0$*, the error* $e = p - p_d$ *converges asymptotically to a neighborhood of zero that can be made arbitrarily small.*

Remark 4 *For the pure tracking control law, no feasibility assumptions are made for the desired trajectory, other than smoothness. This requirement is satisfied if* $p_d(t) \in \mathcal{C}^3$*, i.e. if the desired trajectories are continuous up to its third derivative with respect to time. As long as the desired trajectory satisfies this smoothness assumption, it can be non-feasible in the sense that it may not be reproduced by the dynamic equations of the robot.*

An example of a trajectory that satisfies the smoothness condition $p_d(t) \in \mathcal{C}^3$ w.r.t. and is at the same time non-feasible is depicted in Figures 6 and 7. The non-feasible section of the trajectory is at the coordinate $(0,0)$, where the trajectory presents a sudden change of movement that can not be reproduced by the robot's dynamics. The trajectory shown in Figures 6 and 7 is given by

$$x_d(t) = (t-1)^3 \tag{17a}$$

$$y_d(t) = \begin{cases} 0 & t < 1 \\ (t-1)^3 & t \geq 1 \end{cases} \tag{17b}$$

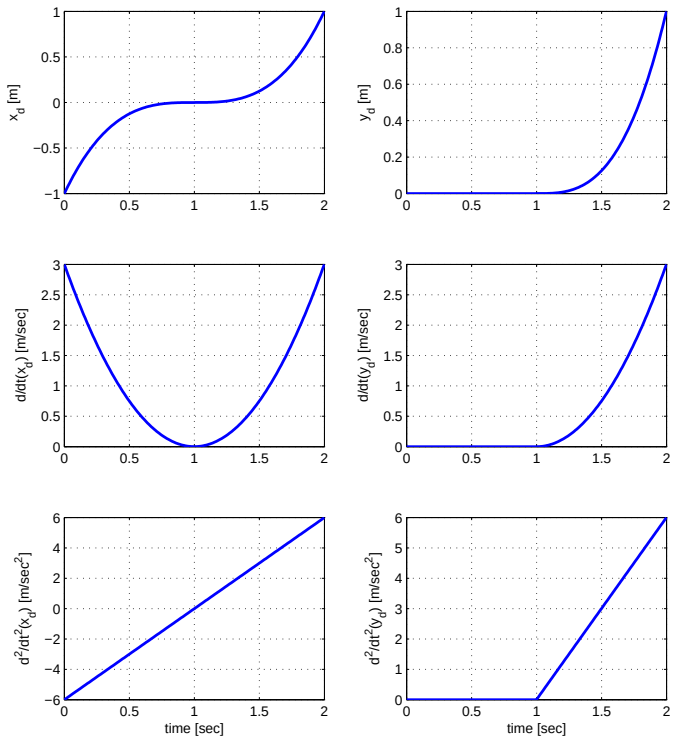


Figure 6: Example of a desired trajectory that is non-feasible for a unicycle but satisfies the smoothness assumption $p_d(t) \in \mathcal{C}^3$ w.r.t. The two plots at the top show the evolution of the x and y coordinates over time. The plots below give their first and second time derivatives. The non-feasible section of the trajectory is located at $t = 1s$.

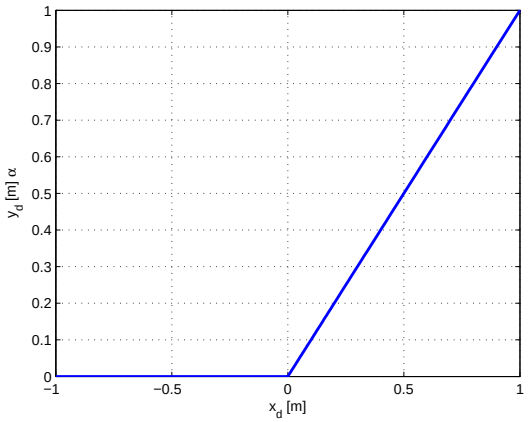


Figure 7: $X-Y$ plot of non-feasible desired trajectory that belongs to the C^3 class (corresponding to the previous Figure and equation (17)). The non-feasible section is located $(x,y)=(0,0)$ coordinates, where the trajectory has a sudden change in direction. Note that an (x,y) trajectory will not necessarily be C^3 on the plane (i.e. considering dx/dy, d^2x/dy^2 and d^3x/dy^3) even though it is C^3 with respect to time.

4 Simulation Results

4.1 Simulations for the two-dimensional tracking law

In order to evaluate the performance of the proposed control law, we present computer simulation results obtained using Simulink©. We run two simulations tests to assess the performance of four leader-follower pairs in a formation of four identical robots with a virtual leader, and plot the described trajectories and the errors in x and y coordinates.

A "virtual" reference cart is programmed to follow a circular trajectory first, and then a lemniscate trajectory. Using the same $[L_{Xoff}\ L_{Yoff}]^T$, agents 1 and 2 are tracking the reference trajectory, agent 3 tracks agent 1, and agent 4 tracks agent 2 (dotted lines are used in Figures 8 and 10 to indicate each leader-follower pair). The controller is implemented identically in the four agents using $K_e=1.5$, $K_z=10$, and $L=0.05$ m. For simulation purposes, we assume that $\dot{L}_Y$ and $\ddot{L}_Y$ are negligible (see Remark 3). We calculate L_Y using (6), using the following values for G, δ and ε, setting $\lambda = L_{Xoff}$:

$$\begin{aligned} G &= -4.0501\lambda^4 + 9.8511\lambda^3 - 6.2869\lambda^2 + 1.6154\lambda - 0.0772, \\ \varepsilon &= 6.9639\lambda^4 - 14.778\lambda^3 + 8.7458\lambda^2 - 0.0207\lambda + 0.1361, \\ \delta &= -8.3333\lambda^4 + 15.37\lambda^3 - 7.8889\lambda^2 + 1.7657\lambda - 0.0311, \end{aligned}$$

for the region where $0.1 < L_{Xoff} < 0.9$. The circular reference trajectory is given by

$$\dot{x} = V_d\cos\theta_d, \qquad \dot{y} = V_d\sin\theta_d, \qquad \dot{\theta}_d = \omega_d, \tag{18}$$

with $V_d = 0.5$ and $\omega_d = 0.1$. The initial conditions of the reference cart are $(x_d(0), y_d(0), \theta_d(0)) = (0,0,0)$ and agents 1, 2, 3, and 4 start with initial conditions: $(-0.3,-1.5,0^\circ)$, $(-0.2,1.6,0^\circ)$, $(-0.8,-0.2,0^\circ)$, and $(-0.5,0.1,0^\circ)$, respectively. The desired offset (L_{Xoff}, L_{Yoff}) between each follower and its associated leader are $(0.5,0.8)$, $(0.5,-0.8)$, $(0.5,-0.4)$, and $(0.5,0.4)$, for agents 1, 2, 3, and 4, respectively, giving rise to a pentagon formation.

The lemniscate reference trajectory used in the second simulation test is defined as:

$$x_d = \frac{a\cos(c\,t)}{1+\sin^2(c\,t)}, \qquad y_d = \frac{a\sin(c\,t)\cos(c\,t)}{1+\sin^2(c\,t)}, \tag{19}$$

with $a = 10$ and $c = 0.1$. The initial position and orientation of the reference cart is $(10, 0, 90^\circ)$, and the initial conditions of agents 1 through 4 are $(10.9, -1, 90^\circ)$, $(0, -0.9, 90^\circ)$, $(11.2, -3, 90^\circ)$, and $(8, -1.1, 90^\circ)$, respectively. The trajectory offset for agents 1 through 4 is as follows: $(0.5, 0.5)$, $(0.5, -0.5)$, $(0.5, 0.5)$, and $(0.5, -0.5)$, respectively, forming a wedge formation.

From Figures 8 and 10, the robots start at different positions with respect to the leader, and after the transient period during which they approach the leader's trajectory, they converge to the formation defined by their corresponding (L_{Xoff}, L_{Yoff}). Dotted lines between agents indicate a leader-follower pair. The error between the desired interagent distance and its actual value is given in Figures 9 and 11; the steady-state error in x does not vanish because of the controller variable L being set to 0.05[1]. When the robots pass from one lobe to the other, following the lemniscate, their reference trajectories approximate straight lines.

4.2 Simulations for the pure tracking special case

In this section we present a simulation result for the pure tracking special case. We use a trajectory similar to that of Eq. (17):

$$x_d(t) = \frac{1}{16}(t-4)^3 \tag{20a}$$

$$y_d(t) = \begin{cases} 0 & t < 4 \\ \frac{1}{16}(t-4)^3 & t \geq 4 \end{cases} \tag{20b}$$

The control parameters are $K_e = 1.5$, $K_z = 10$ and $L = 0.05$ m, and the initial conditions of the robot are $(-4, 0.4, -90^\circ)$. The results are shown in figures 12, 13, and 14.

As seen in Figure 12, the robot tracks the desired trajectory, maintaining asymptotic stability even along the non-feasible section of the reference trajectory, which is more evident in Figure 13. The robot can not change direction suddenly along y, but converges to the desired trajectory after a transient period. The effects of the abrupt change in direction of the reference can also be observed in the linear and angular speeds, position and heading errors, and torques plotted in Figure 14.

5 Experimental Results

In this section we present experimental tests performed on two robotic platforms like the one shown in Figure 1. We consider formations of one virtual leader and two nonholonomic mobile robots. In what follows, we first explain the experimental setup, and then present the results obtained in two tests.

5.1 Test set-up and communication issues

During this experimental test we assume that $\dot{L}_Y$ and $\ddot{L}_Y$ are negligible. The experimental set-up includes four computers (Figure 15). The first computer acts as a central command or a virtual leader, generating the reference trajectory and transmitting it to the robots. Two other computers are laptops on board the robots used to control them. The fourth computer collects and organizes the experimental data. The virtual leader and the data collector

[1] L is never set to zero because it is used as safety parameter such that L_{Xoff} can be zero, in this form $L_X = L_{Xoff} + L$ is guaranteed to be greater than zero independently of the choice of L_{Xoff} such that a singularity in the control input is avoided (see equation (13)).

computers are linked over a LAN, while the robots are connected over a wireless LAN (IEEE 802.11b). We chose User Datagram Protocol (UDP) as the transmission protocol, because TCP exhibits packet retention due to its "slow start" feature [38]. The desired trajectory is transmitted to the mobile agents, who re-transmit it along with their own state to the data collector (Figure 15). No information is communicated directly between the virtual leader computer and the data collection computer, and therefore synchronization is not needed. We use the transmitted desired trajectory, to concatenate the data in an ordered form once the experiment is concluded. Controller implementation becomes an issue, due to the needed concurrency in process execution, and is dealt with using multithreading capability of LabView© (Figure 15).

5.2 Results

Two reference trajectories are used: a circle and lemniscate section (restricted due to limited space in the lab). Results are comparable to those obtained in the simulations. Videos are available at [39].

In the test described in Figure 16, a circular trajectory is calculated online in the central command computer, and transmitted to the mobile robots. The controller parameters L, K_e, K_z, G, δ and ε keep their simulation values. For the reference trajectory (18), we use $V_d = 0.04$ m and $\omega_d = 0.04$ rad/sec, with initial conditions $(0,0,0)$. The initial conditions of robots 1 and 2 are $(0,-0.6,90^\circ)$, and $(-0.6,0.3,0^\circ)$, respectively, and the corresponding desired offsets $(0.30,0.35)$ and $(0.30,-0.35)$.

For the test shown in Figure 17, a section of the lemniscate reference trajectory (19) is calculated with $a = 2.4m$, $c = 0.02$ and initial conditions $(2.4,0,90^\circ)$. The initial conditions of robots 1 and 2 are $(2.8,-0.3,90^\circ)$, and $(2.0,-0.9,0^\circ)$, respectively and the corresponding offsets $(0.1,0.3)$ and $(0.1,-0.3)$.

Figures 16 and 17 show that after a transient, in which the robots converge to their desired position in the leader-follower pair setting, they move along the trajectory maintaining the desired (triangle) formation. Without the availability of environment measurements, odometry has to be accurately calibrated. In addition, communication delays during desired trajectory transmission, caused by temporary link failures resulted in the robots stopping and converging to the last received position. When delays are short, the robots are able to recover their position in the formation, but if the delay is excessively long or permanent, the the leader follower pairs are destabilized.

6 Odometry Cumulative Error Analysis

The cumulative effect of odometry errors, evident in the videos of [39] motivates the analysis of this section.

6.1 Odometry system description

The odometry system consists of one optical encoder attached to each motor that provides the DAC with pulsed signals that allow it to estimate small angular displacements. Let Δ_r and Δ_l be the linear displacement of the right and left wheels respectively during one program iteration. Then the linear and angular displacement (Δ and Θ) of the robot for each iteration can be calculated as:

$$\Delta = \frac{\Delta_r + \Delta_l}{2} \tag{21a}$$

$$\Theta = \frac{\Delta_r - \Delta_l}{2l}, \tag{21b}$$

With the linear and angular speeds v and ω being constant over each (constant) sampling time T_s, the discrete-time kinematics obtained from the kinematic model equations (1) are

$$x_n = \begin{cases} \Delta\cos\theta_n + x_{n-1} & \Theta = 0 \\ \frac{\Delta}{\Theta}(\sin\theta_n - \sin\theta_{n-1}) + x_{n-1} & \Theta \neq 0 \end{cases} \tag{22a}$$

$$y_n = \begin{cases} \Delta\sin\theta_n + y_{n-1} & \Theta = 0 \\ -\frac{\Delta}{\Theta}(\cos\theta_n - \cos\theta_{n-1}) + y_{n-1} & \Theta \neq 0 \end{cases} \tag{22b}$$

$$\theta_n = \Theta + \theta_{n-1}, \tag{22c}$$

where subscripts denote time steps. In this form, using equations (21), and (22) the position of the cart can be estimated from readings of the wheel's encoders and the initial conditions provided to the robot at the beginning of any experiment.

6.2 Instantaneous error in position measurement

In order to analyze the effect of the odometry errors on the system we make the following assumptions.

Assumption 1 *Assume that the errors in the measurement of the linear displacement of the wheels are the same for all iterations, identical for both encoders and additive to the real displacement, i.e.,*

$$\Delta_r = \hat{\Delta}_r + \zeta \tag{23a}$$

$$\Delta_l = \hat{\Delta}_l + \zeta, \tag{23b}$$

where $\hat{\cdot}$ denotes measured value and ζ is the additive error.

Assuming that the error is identical for both encoders may seem to be restrictive, but we consider it a reasonable price to pay for the simplification of the analysis task, if the odometry system is designed carefully. This assumption implies that the estimation of the orientation of the robot is error free (see equation (24)), which makes the rotational matrix R in the Lyapunov analysis independent of the error (note that R includes trigonometric functions of the orientation so obtaining a useful result considering errors in orientation estimation becomes very difficult or even impossible). Also note that although the errors may not be the same for all iterations, assuming they are is not restrictive because our objective is to obtain a condition on the number of iterations that the robot is allowed to estimate its position using dead-reckoning before an external measurement is needed. This can be achieved considering the maximum possible error instead of the exact error for each iteration.

Assumption 2 *Assume that the linear and angular speeds can be measured directly and accurately.*

This assumption makes sense because the robot uses Frequency-to-Voltage Converters (FVC) to convert the pulsed signals of the encoders to a constant voltage provided to the DAC that is proportional to the linear speeds of the wheels, so while the errors in the position estimation are cumulative, the errors in the velocity estimation are not.

Next, define $\hat{\Delta} = \frac{\hat{\Delta}_r + \hat{\Delta}_l}{2}$, and $\hat{\Theta} = \frac{\hat{\Delta}_r - \hat{\Delta}_l}{2l}$. Then using Assumption 1 and equation (21) we obtain

$$\Delta = \hat{\Delta} + \zeta \tag{24a}$$

$$\Theta = \hat{\Theta}, \tag{24b}$$

so the angular displacement is error free under Assumption 1. Including (24) in (22) we obtain

$$x_n = \hat{x}_n + \tilde{x}_n \tag{25a}$$

$$y_n = \hat{y}_n + \tilde{y}_n \tag{25b}$$

$$\theta_n = \hat{\theta}_n, \tag{25c}$$

where $\hat{x}_n$, $\hat{y}_n$, and $\hat{\theta}_n$ are given by (22) after substituting Δ and Θ for $\hat{\Delta}$ and $\hat{\Theta}$, respectively, and $\tilde{x}_n$, $\tilde{y}_n$ are the measurement errors given by:

$$\tilde{x}_n = \begin{cases} \zeta \cos\theta_n, & \text{if } \Theta = 0 \\ \frac{\zeta}{\Theta}(\sin\theta_n - \sin\theta_{n-1}), & \text{if } \Theta \neq 0 \end{cases} \tag{26a}$$

$$\tilde{y}_n = \begin{cases} \zeta \sin\theta_n, & \text{if } \Theta = 0 \\ -\frac{\zeta}{\Theta}(\cos\theta_n - \cos\theta_{n-1}), & \text{if } \Theta \neq 0. \end{cases} \tag{26b}$$

Using trigonometric properties this equation can be rewritten as

$$\tilde{x}_n = \begin{cases} \zeta \cos\theta_n, & \text{if } \Theta = 0 \\ \frac{2\zeta}{\Theta}\cos(\frac{\theta_n+\theta_{n-1}}{2})\sin(\frac{\theta_n-\theta_{n-1}}{2}), & \text{if } \Theta \neq 0 \end{cases} \tag{27a}$$

$$\tilde{y}_n = \begin{cases} \zeta \sin\theta_n, & \text{if } \Theta = 0 \\ \frac{2\zeta}{\Theta}\sin(\frac{\theta_n+\theta_{n-1}}{2})\sin(\frac{\theta_n-\theta_{n-1}}{2}), & \text{if } \Theta \neq 0 \end{cases} \tag{27b}$$

and approximating, $\frac{\theta_n+\theta_{n-1}}{2} \approx \theta_n$, and $\sin(\Theta) \approx \Theta$ for Θ very small[2] we obtain

$$\tilde{x}_n = \zeta \cos\theta_n \tag{28a}$$

$$\tilde{y}_n = \zeta \sin\theta_n, \tag{28b}$$

so the errors $\tilde{x}_n$ and $\tilde{y}_n$ can be bounded by ζ i.e., $\tilde{x}_n \leq \zeta$ and $\tilde{y}_n \leq \zeta$ for all Θ. Thus we can write:

$$x_n \in (\hat{x}_n - \zeta, \hat{x}_n + \zeta) \tag{29a}$$

$$y_n \in (\hat{y}_n - \zeta, \hat{y}_n + \zeta) \tag{29b}$$

$$\theta_n = \hat{\theta}_n, \tag{29c}$$

6.3 Impact of odometry errors in convergence

In this section we perform a worst case analysis to quantify the effect of odometry errors on stability within time intervals where absolute position measurements are unavailable. Propagating (29) through (22) we obtain

$$x_N \in (\hat{x}_N - N\zeta, \hat{x}_N + N\zeta) \tag{30a}$$

$$y_N \in (\hat{y}_N - N\zeta, \hat{y}_N + N\zeta) \tag{30b}$$

$$\theta_N = \hat{\theta}_N, \tag{30c}$$

but considering the worst case scenario we can assume that,

$$x_N = \hat{x}_N + N\zeta \tag{31a}$$

$$y_N = \hat{y}_N + N\zeta \tag{31b}$$

$$\theta_N = \hat{\theta}_N, \tag{31c}$$

So, unless absolute position measurements become available (after, say, N steps), the tracking performance will deteriorate with time. Assuming that absolute position measurements are somehow made at step N, we quantify the impact on convergence. Consider the proposed control law

$$u_1 = m\left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}^T (-\hat{h} - K_z\hat{z}) + \frac{L_Y}{L_X}\begin{pmatrix} 0 \\ 1 \end{pmatrix}^T (-\hat{h} - K_z\hat{z}) \right\} \tag{32a}$$

$$u_2 = \frac{J}{L_X}\begin{pmatrix} 0 \\ 1 \end{pmatrix}^T (-\hat{h} - K_z\hat{z}), \tag{32b}$$

[2] Θ will be guaranteed to be very small by using a small sampling period T_s given that the robot is only capable of finite angular speeds.

where $\hat{h}$ and $\hat{z}$ are given by (12) and (10) substituting the exact quantities by the measured quantities, i.e.,

$$\hat{z} = \begin{bmatrix} -L_Y\omega+v \\ L_X\omega \end{bmatrix} - \left(R\dot{p}_d - \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix} - K_e\hat{e} \right), \tag{33}$$

$$\hat{h} = \begin{bmatrix} -\dot{L}_Y\omega + \frac{L_Y}{f}\psi\omega - \frac{\eta}{m}v \\ -\frac{L_X}{f}\psi\omega \end{bmatrix} + SR\dot{p}_d - R\ddot{p}_d + \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix} + K_e\left(- S\hat{e} + \begin{bmatrix} -L_Y\omega+v \\ L_X\omega \end{bmatrix} - R\dot{p}_d + \begin{bmatrix} 0 \\ \dot{L}_Y \end{bmatrix} \right), . \tag{34}$$

and where $\hat{e}$ is given by

$$\hat{e} = R(\hat{p} - p_d) + \begin{bmatrix} L_X \\ L_Y \end{bmatrix}, \tag{35}$$

and $\hat{p}$ is the measured position vector $\begin{bmatrix} \hat{x} & \hat{y} \end{bmatrix}^T$

Substituting the control law (32), in $\dot{V}_2$ (equation (11)) we obtain $\dot{V}_2 = -K_e\|e\|_2^2 + z^T\left(h - \hat{h} - K_z\hat{z}\right)$, which after the inclusion of (33), and (34) becomes $\dot{V}_2 = -K_e\|e\|_2^2 - K_z\|z\|_2^2 - z^T\left(K_eS\tilde{e} + K_eK_z\tilde{e}\right)$ where $\tilde{e} = e - \hat{e} = R\begin{bmatrix} \tilde{x} & \tilde{y} \end{bmatrix}^T$. With this expression, and equation (29) $\dot{V}_2$ can be further simplified to

$$\dot{V}_2 = -K_e\|e\|_2^2 - K_z\|z\|_2^2 + K_eN\zeta z^T \begin{bmatrix} -\left(K_z(\cos\theta+\sin\theta)+\omega(\cos\theta-\sin\theta)\right) \\ -\left(K_z(\cos\theta-\sin\theta)-\omega(\cos\theta+\sin\theta)\right) \end{bmatrix}.$$

Applying the inequality $\alpha^T\beta \leq \|\alpha\|_2\|\beta\|_2$, and given that the angular speed has maximum attainable magnitude $\omega \leq \omega_M$, $\dot{V}_2$ can be bounded as

$$\dot{V}_2 \leq -K_e\|e\|_2^2 - K_z(1-\gamma)\|z\|_2^2 - K_z\gamma\|z\|_2^2 + \sqrt{2}K_eN\zeta\|z\|_2\left(K_z^2 + w_M^2\right)^{1/2}$$

for some $\gamma \in (0,1)$. Thus $\dot{V}_2 \leq -K_e\|e\|_2^2 - K_z(1-\gamma)\|z\|_2^2$ for all $\|z\|_2 \geq \frac{\sqrt{2}K_eN\zeta}{K_z\gamma}\left(K_z^2 + w_M^2\right)^{1/2}$. Based on Theorem 4.18 of [40], we establish the following result:

Theorem 2 *In the closed loop system* (3) *and* (32)*, with* $\hat{h}$ *given by* (34)*,* $\hat{z}$ *by* (33)*, and* $K_e, K_z > 0$*, z is uniformly ultimately bounded by a linear function of the number of steps* N *:* $\frac{\sqrt{2}K_eN\zeta}{K_z\gamma}\left(K_z^2 + w_M^2\right)^{1/2}$.

Remark 5 *The importance of this result is that it enables us to know how soon we need ro "recalibrate" the system in order to maintain bounded the tracking errors within the formation.*

7 Conclusions

We have developed a leader-follower control law that makes a mobile robot track a desired trajectory at a specific position in the plane with respect to its leader and highlighted its potential use in formation control. This control law has been designed using backstepping, based on a unicycle model that includes a dynamic extension.

The use of the control law in the case of pure tracking has also been discussed, and we have demonstrated our ability to track non-feasible, sufficiently smooth trajectories. Finally, the effects of the odometry cumulative errors in the closed-loop system have been analyzed, showing that error boundedness can be guaranteed if the position estimates are periodically corrected using absolute position measurement.

Future research directions for this work include the extension of this control law considering agent interactions, obstacle avoidance, and compensation for communication issues.

References

[1] T. Lee, K. Song, C. Lee, and C. Teng, "Tracking control of unicycle-modeled mobile robots using a saturation feedback controller," *IEEE Trans. on Control Systems Technology*, vol. 9, no. 2, pp. 305–318, Mar. 2001.

[2] W. Dixon, D. Dawson, and E. Zergeroglu, "Tracking and regulation control of a mobile robot system with kinematic disturbances: A variable structure-like approach," *Journal of Dynamic Systems, Measurement and Control*, vol. 122, pp. 616–623, Dec. 2000.

[3] G. Artus, P. Morin, and C. Samson, "Tracking of an omnidirectional target with a nonholonomic mobile robot," in *Proc. of the International Conference on Advanced Robotics*, Coimbra, Portugal, June 2003, pp. 1468–1473.

[4] E. Lefeber, J. Jakubiak, K. Tchon, and H. Nijmeijer, "Observer based kinematic controllers for a unycicle-type mobile robot," in *Proc. of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, May 2001, pp. 2084–2089.

[5] A. Divelbiss and J. Wen, "Trajectory tracking control of a car-trailer system," *IEEE Trans. on Control Systems Technology*, vol. 5, no. 3, pp. 269–278, Mar. 1997.

[6] G. Oriolo, A. D. Luca, and M. Vendittelli, "WMR control via dynamic feedback linearization: Design, implementation, and experimental validation," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 6, pp. 835–851, Nov. 2002.

[7] R. Fierro and F. Lewis, "Control of a nonholonomic mobile robot: Backstepping kinematics into dynamics," *Journal of Robotic Systems, by John Wyley & Sons, Inc.*, vol. 14, no. 3, pp. 149–163, 1997.

[8] J. Yang and J. Kim, "Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots," *IEEE Trans. on Robotics and Automation*, vol. 15, no. 3, pp. 578–587, June 1999.

[9] D. Chwa, "Sliding mode control of nonholonomic wheeled mobile robots in polar coordinates," *IEEE Trans. on Control Systems Technology*, vol. 12, no. 4, pp. 637–644, July 2004.

[10] T. Fukao, H. Nakagawa, and N. Adachi, "Adaptive tracking control of a nonholonomic mobile robot," *IEEE Trans. on Robotics and Automation*, vol. 16, no. 5, pp. 609–615, October 2000.

[11] Z. Jiang and H. Nijmeijer, "Tracking control of mobile robots: A case study in backstepping," *Automatica, Elsevier Science Ltd.*, vol. 33, no. 7, pp. 1393–1399, 1997.

[12] D. Wang and G. Xu, "Full-state tracking and internal dynamics of nonholonomic wheeled mobile robots," *IEEE/ASME Transactions on Mechatronics*, vol. 8, no. 2, pp. 203–214, June 2003.

[13] D. Panfilov and S. Tkachev, "Tracking of reference trajectory for wheeled mobile robot," in *Proceedings of the 14th. International Conference Process Control*, Strbske Pleso, Slovakia, June 2003, pp. 095/1–095/7.

[14] M. Khatir and E. Davison, "Decentralized control of a large platoon of vehicles operating on a plane with steering dynamics," in *Proc. of the American Control Conference*, Portland, OR, USA, June 2005, pp. 2159–2165.

[15] J. Marshall, M. Broucke, and B. Francis, "Formations of vehicles in cyclic pursuit," *IEEE Trans. on Automatic Control*, vol. 49, no. 11, pp. 1963–1974, Nov. 2004.

[16] J. Marshall, T. Fung, M. Broucke, G. D'Eleuterio, and B. Francis, "Experimental validation of multi-vehicle coordination strategies," in *Proc. of the American Control Conference*, Portland, OR, USA, June 2005, pp. 1090–1095.

[17] M. Egerstedt and X. Hu, "Formation constrained multi-agent control," *IEEE Trans. on Robotics and Automation*, vol. 17, no. 6, pp. 947–951, Dec. 2001.

[18] R. Olfati-Saber and R. Murray, "Flocking with obstacle avoidance: Cooperation with limited communication in mobile networks," in *Proc. of the IEEE Conference on Decision and Control*, Maui, Hawaii, USA, Dec 2003, pp. 2022–2028.

[19] H. Tanner, G. Pappas, and V. Kumar, "Leader-to-formation stability," *IEEE Trans. on Robotics and Automation*, vol. 20, no. 3, pp. 443–455, June 2004.

[20] H. Hsu and A. Liu, "Multi-agent based formation control using a simple representation," in *Proc. of the IEEE International Conference on Networking, Sensing and Control*, Taipei, Taiwan, March 2004, pp. 276–281.

[21] V. Gazi and M. Passino, "Stability analysis of swarms," *IEEE Trans. on Automatic Control*, vol. 48, no. 4, pp. 692–697, Apr. 2003.

[22] S. Swaroop and J. Hedrick, "String stability of interconected systems," *IEEE Trans. on Automatic Control*, vol. 41, no. 3, pp. 349–356, Mar. 1996.

[23] A. Jadbabaie, J. Lin, and S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. on Automatic Control*, vol. 48, no. 6, pp. 988–1001, June 2003.

[24] Y. Liu, K. Passino, and M. Polycarpou, "Stability analysis of m-dimensional asynchronous swarms with a fixed communication topology," *IEEE Trans. on Automatic Control*, vol. 48, no. 1, pp. 76–95, Jan. 2003.

[25] H. Tanner, A. Jadbabaie, and J. Pappas, "Stable flocking of mobile agents, part I: Fixed topology," in *Proc. of the IEEE Conference on Decision and Control*, Maui, Hawaii, USA, Dec 2003, pp. 2010–2015.

[26] ——, "Stable flocking of mobile agents, part II: Dynamic topology," in *Proc. of the IEEE Conference on Decision and Control*, Maui, Hawaii, USA, Dec 2003, pp. 2016–2021.

[27] H. Tanner, "Flocking with obstacle avoidance in switching networks of interconnected vehicles," in *Proc. of the IEEE International Conference on Robotics and Automation*, New Orleans LA, USA, May 2004, pp. 3006–3011.

[28] A. Regmi, R. Sandoval, R. Byrne, H. Tanner, and C. Abdallah, "Experimental implementation of flocking algorithms in wheeled mobile robots," in *Proc. of the American Control Conference*, Portland, OR, USA, June 2005, pp. 4917–4922.

[29] J. Desai, J. Ostrowski, and V. Kumar, "Controlling formations of multiple mobile robots," in *Proc. of the IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998, pp. 2864–2869.

[30] R. Sandoval-Rodriguez, C. Abdallah, P. Hokayem, E. Schamiloglu, and R. Byrne, "Robust mobile robotic formation control using internet-like protocols," in *Proc. of the IEEE Conference on Decision and Control*, Maui, Hawaii, USA, Dec 2003, pp. 5109–5112.

[31] B. Young, R. Beard, and J. Kelsey, "A control scheme for improving multi-vehicle formation maneuvers," in *Proc. of the American Control Conference*, vol. 2, Arlington, VA, USA, June 2001, pp. 704–709.

[32] A. Pant, P. Seiler, and K. Hedrick, "Mesh stability of look-ahead interconnected systems," *IEEE Trans. on Automatic Control*, vol. 47, no. 2, pp. 403–407, Feb. 2002.

[33] W. Dumbar and R. Murray, "Distributed receding horizon control with application to multi-vehicle stabilization," California Institute of Technology, Pasadena, CA. 91125, Tech. Rep. CaltechCDSTR:2004.003, 2004.

[34] P. Ogren, M. Egerstedt, and X. Hu, "A control lyapunov function approach to multiagent coordination," *IEEE Trans. on Robotics and Automation*, vol. 18, no. 5, pp. 847–851, Oct. 2002.

[35] R. Sandoval, "Teleautonomy of mobile robots," Doctoral Dissertation, Electrical and Computer Engineering Department - University of New Mexico, Albuquerque, NM, USA, 2004.

[36] A. Regmi, "Experimental implementation for mobile robots coordination," Masters Thesis, Electrical and Computer Engineering Department - University of New Mexico, Albuquerque, NM, USA, 2004.

[37] A. Aguiar and J. Hespanha, "Logic-based switching control for trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," in *Proc. of the American Control Conference*, vol. 4, Boston, MA, USA, June 2004, pp. 3004–3010.

[38] R. Sandoval-Rodriguez, C. Abdallah, H. Jerez, and R. Byrne, "Experimental results on the effects of 802.11b WLAN on networked control system," in *Proc. of the Mediterranean Conference on Control and Automation*, Limassol, Cyprus, June 2005, pp. 1113–1118.

[39] http://www.ece.unm.edu/~jlpiovesan/MSThesis.htm.

[40] H. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2002.

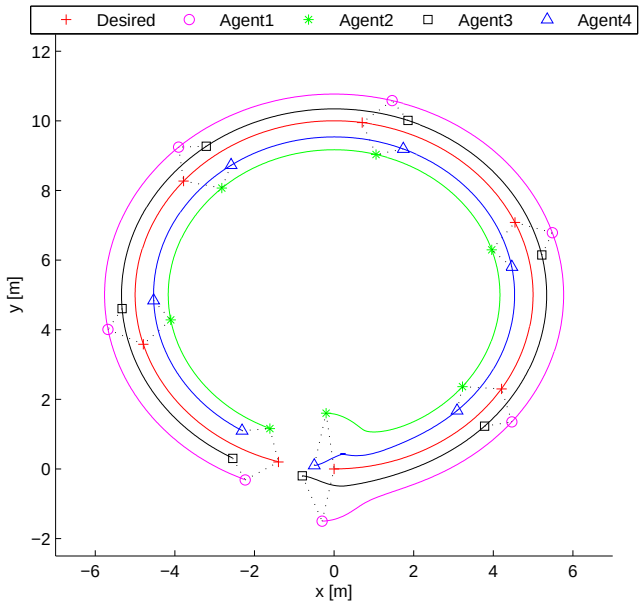


Figure 8: Four-agent formation tracking a circular trajectory.

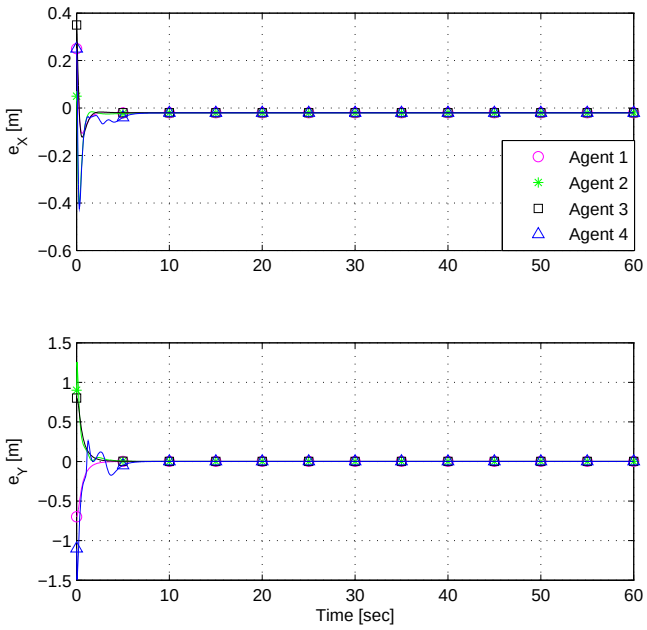


Figure 9: Time evolution of the position error for the circular trajectory.

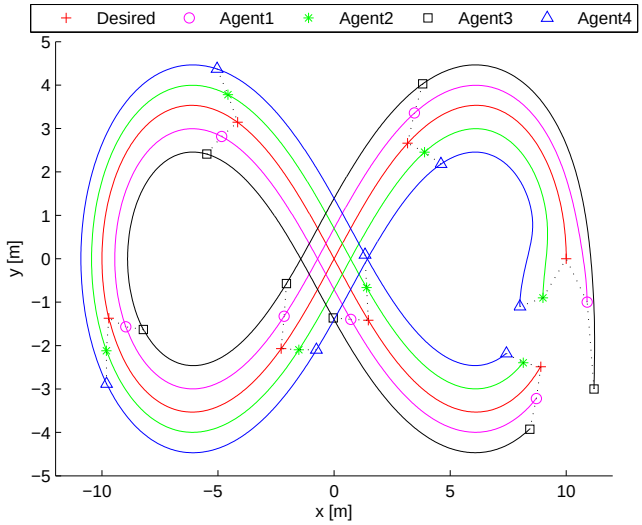


Figure 10: Four-agent formation tracking a lemniscate trajectory.

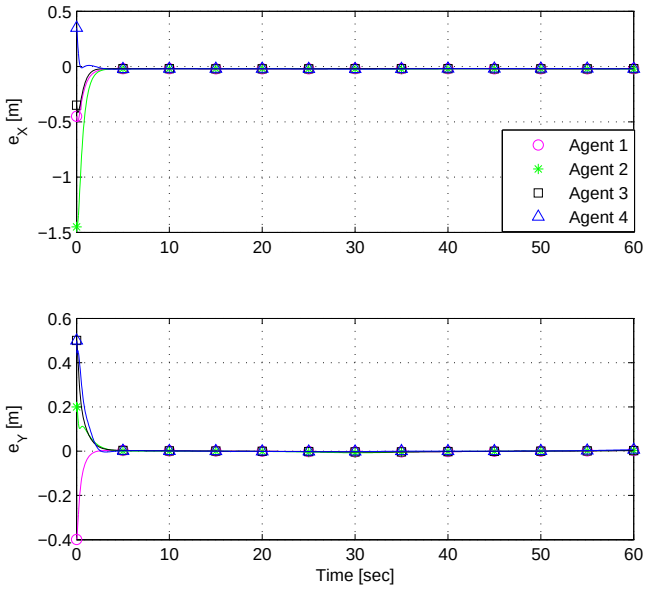


Figure 11: Time evolution of the position error for the lemniscate reference.

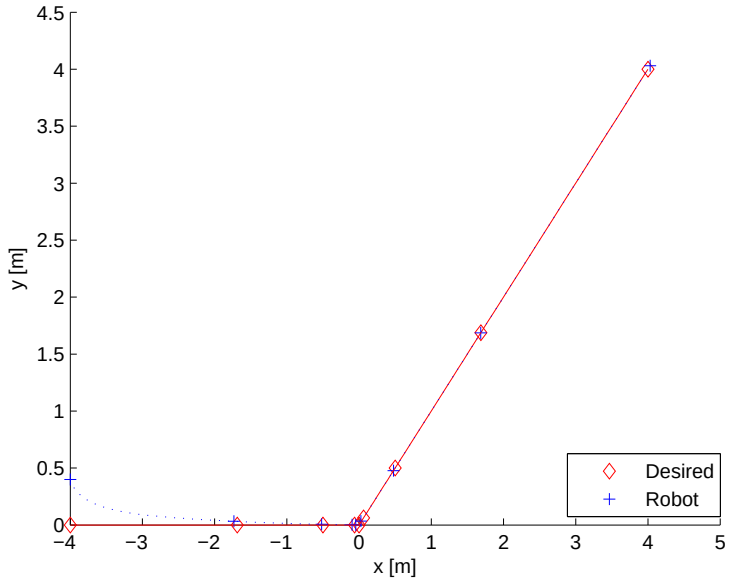


Figure 12: Desired and actual trajectory of the robot for a non-feasible trajectory

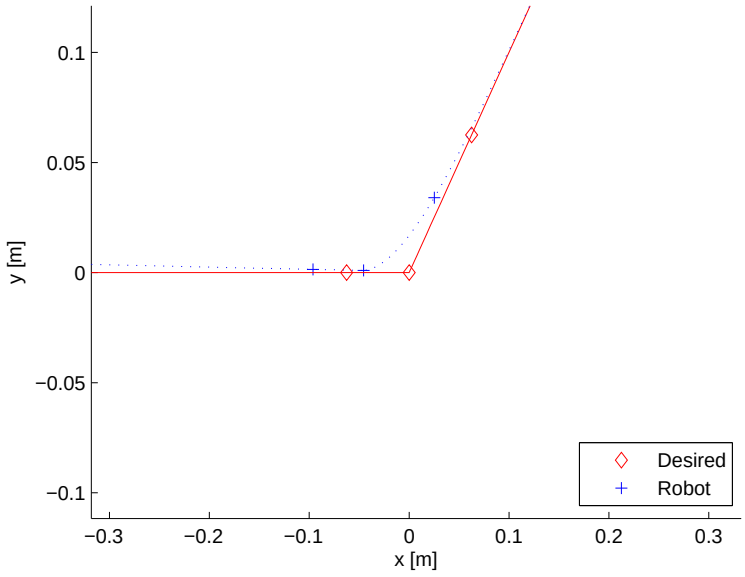


Figure 13: Zoomed section of sudden change of the desired and actual paths near $t = 4$ sec.

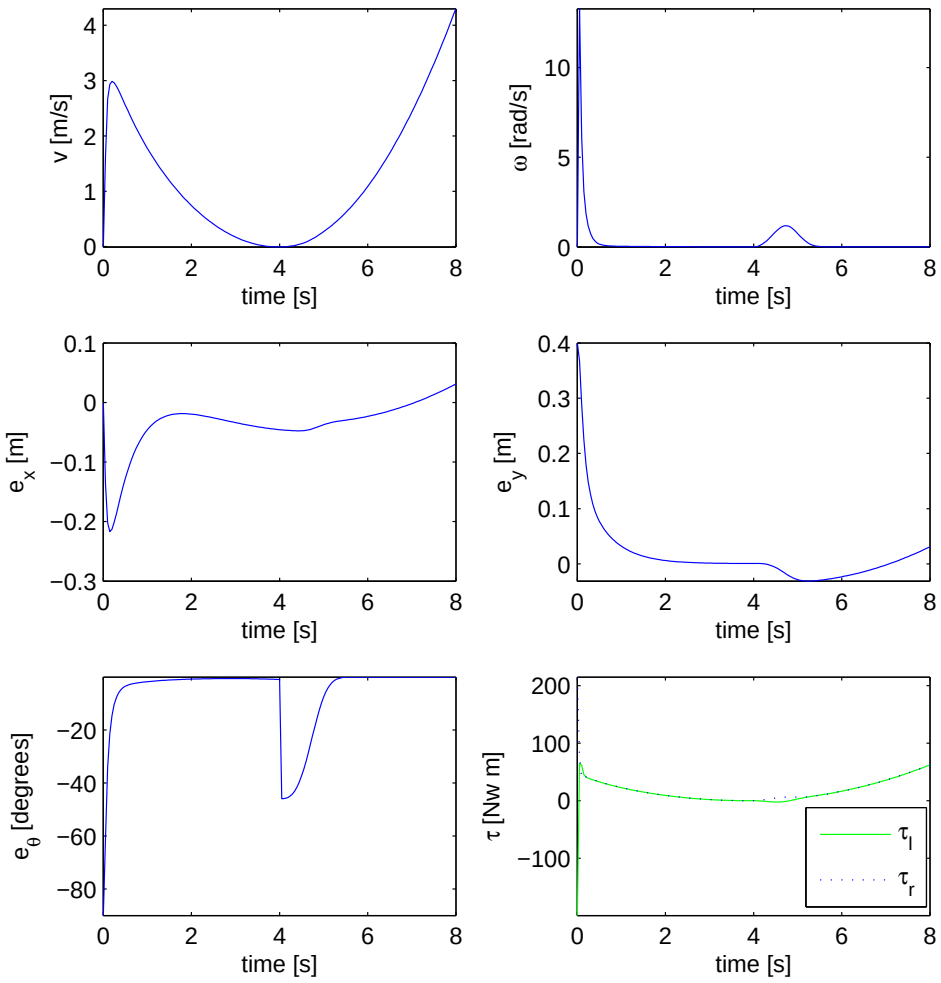


Figure 14: Time plots of the speeds, errors and torques for the non-feasible trajectory simulation

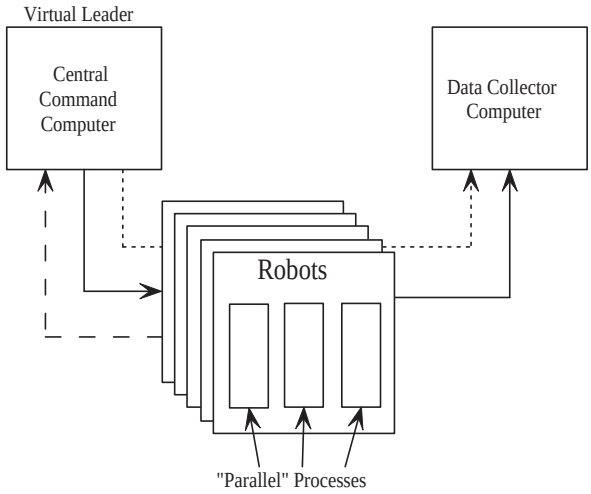


Figure 15: Experimental implementation structure.

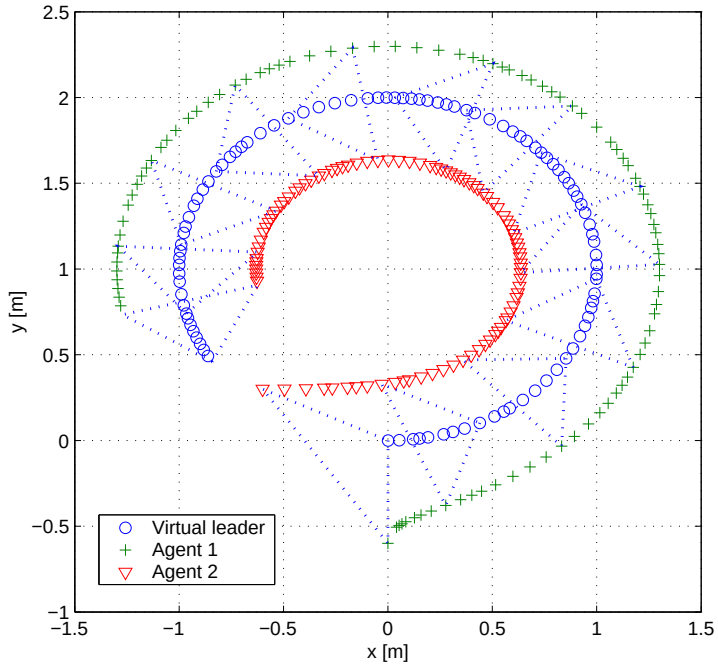


Figure 16: Experimental results for robot paths along a circular trajectory.

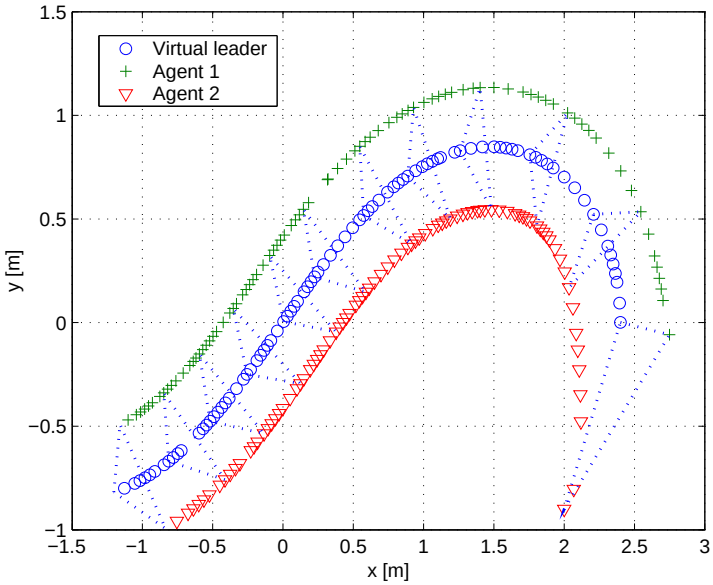


Figure 17: Experimental results for robot paths along a section of a lemniscate trajectory.