# Formation Control and Collision Avoidance for Multi-agent Non-holonomic Systems: Theory and Experiments

Silvia Mastellone, Dusan M. Stipanovic, Christopher R. Graunke, Koji A. Intlekofer and Mark W. Spong

*The International Journal of Robotics Research* 2008 27: 107
DOI: 10.1177/0278364907084441

**Silvia Mastellone**
**Dušan M. Stipanović**
**Christopher R. Graunke**
**Koji A. Intlekofer**
**Mark W. Spong**

Coordinated Science Laboratory, University of Illinois,
1308 West Main Street, Urbana, IL 61801, USA
{smastel2,dusan,graunke,intlekof,mspong}@uiuc.edu

# Formation Control and Collision Avoidance for Multi-agent Non-holonomic Systems: Theory and Experiments

## Abstract

*In this paper we present a theoretical and experimental result on the control of multi-agent non-holonomic systems. We design and implement a novel decentralized control scheme that achieves dynamic formation control and collision avoidance for a group of non-holonomic robots. First, we derive a feedback law using Lyapunov-type analysis that guarantees collision avoidance and tracking of a reference trajectory for a single robot. Then we extend this result to the case of multiple non-holonomic robots, and show how different multi-agent problems, such as formation control and leader–follower control, can be addressed in this framework. Finally, we combine the above results to address the problem of coordinated tracking for a group of agents. We give extensive experimental results that validate the effectiveness of our results in all three cases.*

KEY WORDS—autonomous agents, wheeled robots, distributed robot systems, control of non-holonomic systems, formation control, collision avoidance

## 1. Introduction

The technological revolution that came along in the last century with the advent of wireless communication brought a breadth of innovation and provided ways to efficiently share information between systems. Interacting systems are no longer constrained to be physically connected. Thus, in several applications a single complex system can be replaced by interacting multi-agent systems with simpler structure. In fact, a group of small non-holonomic robots (unicycles, car-like robots or unmanned aerial vehicles (UAVs)) with simple structures can achieve more complex tasks at a lower cost than a single complex robot owing to their modularity and flexibility. In this framework a new set of problems needs to be addressed for groups of robots with non-holonomic dynamics such as coordination and formation control while guaranteeing collision avoidance. In this paper we present a novel approach that addresses these problems.

### *1.1. Background*

The problem of coordination of multiple agents has been addressed using different approaches, various stability criteria and numerous control techniques. Some of the existing approaches, as highlighted by Tanner (2004), include the behavior-based approaches of Balch and Arkin (1998), where an interaction law between the subsystems is defined that leads to the emergence of a collective behavior. The leader–follower approach of Tanner et al. (2004) defines a hierarchy between the agents where one or more leaders drive the configuration scheme generating commands, while the followers follow the commands generated by the leaders. Another approach focuses on maintaining a certain group configuration and forces each agent to behave as a particle in a rigid virtual structure (Desai et al. 1999, 2001; Egerstedt and Hu 2001). As we are considering a group of interacting systems, classical stability needs to be redefined to include the interconnection aspects between the systems. To this end the classical concept of stability has been extended, for example, by Tanner et al. (2002) (where formation stability is analyzed using input-to-state stability), by Swaroop and Hedrick (1996) (where string stability is adopted to analyze the system's behavior) and by Šiljak (1991) (where connective stability for interconnected systems has been studied). In a recent survey by Murray (2007), two main methods were identified to solve formation control problems: an

The International Journal of Robotics Research
Vol. 27, No. 1, January 2008, pp. 107–126
DOI: 10.1177/0278364907084441

Figures 1, 3–8, 10, 16–20 appear in color online: http://ijr.sagepub.com

optimization-based method (Parker 1993; Dunbar and Murray 2006) and a potential fields method (Leonard and Fiorelli 2001b; Atkar et al. 2002; Justh and Krishnaprasad 2004; Ögren et al. 2004).

When one considers systems with non-holonomic constraints, the formation control problem becomes more challenging. Several techniques both for open- and closed-loop control can be found in Canudas de Wit et al. (1996) and Murray et al. (1994). Some representative papers in the area of formation control for non-holonomic dynamics are those of Leonard and Fiorilli (2001a) and Sepulchre et al. (2004). Stipanović et al. (2004), using dynamic extension, linearized the model locally for non-zero velocities and used a decentralized overlapping scheme to control the formation. Some of the existing results addressing tracking problems for non-holonomic systems are reported by Micaelli and Samson (1993), Bloch and Drakunov (1995) and Dong et al. (2000), where sliding-mode control is adopted to achieve tracking. In addition, Wu et al. (2005) achieved tracking by means of adaptive control and Morgansen (2001) used amplitude modulated sinusoids. Finally, Tanner et al. (2001) achieved stabilization for non-holonomic dynamics as well as collision avoidance using global barrier functions.

### *1.2. Contributions*

The main contributions of our work are the design of a controller which guarantees coordinated tracking and collision avoidance for groups of robots with non-holonomic dynamics and a set of experiments which implement the controller on robotic testbed.

In the theoretical part of this paper, we design a controller that guarantees tracking with bounded error and obstacle/vehicle collision avoidance for non-holonomic systems. We assume that each robot knows its position and can detect the presence of any object within a given range. We apply this result to formation control for multi-agent systems. Finally, we address the problem of coordinated tracking, where the objective is to drive a group of robots according to a trajectory provided by a supervisor while maintaining a specific formation. We use a decentralized architecture in which the controllers are implemented locally on each agent. The tracking part of the controller uses local information about the agent's current position as well as the position of the leader or virtual leader (center of mass of the formation). On the other hand, the avoidance part of the controller for each agent uses position information of obstacles as well as other mobile agents as they enter the detection region. More specifically, the collision avoidance control acts in real time and uses locally defined potential functions which can take different shapes and only require each agent to detect objects in its neighborhood. This is an advantage that distinguishes our method from some other potential field approaches such as that of Khatib (1986). Moreover, we show how this approach can be generalized to multiple robots to achieve formation control and mutual avoidance.

We couple the theoretical results with extensive experiments performed on a mobile robotic platform that was built at the College of Engineering Mechatronics Laboratory at the University of Illinois at Urbana-Champaign. The experimental setup consists of an enclosed workspace and a color camera mounted on the ceiling above the workspace to detect the positions and orientations of the differential drive robots. Each of the robots has wireless communication and onboard processing capability.

Each robot receives its position update information that is determined by the overhead camera and vision software. Moreover, each robot utilizes position information from local encoders mounted on the drive motors. Two color swatches are placed on top of each robot and are used to determine its position and orientation.

The experimental results show that the controllers we designed are easily implementable and are robust with respect to communication unreliability, such as delays, quantization, communication dropout and bounded disturbances. As such, the experimental implementation helped us identify practical issues that constitute points of possible improvement for future studies.

The rest of the paper is organized as follows. In Section 2 a controller is designed that guarantees tracking and collision avoidance for a single non-holonomic robot. We address the cases of static obstacle avoidance and cooperative avoidance between agents. In Section 3 we formulate the formation control problem as a tracking problem and solve it. In Section 4 we combine the previous results to achieve velocity tracking for a group of non-holonomic robots while maintaining a desired formation. In Section 5 we describe the experimental implementation of the formation control and collision avoidance on differential drive robots. Finally, in Section 6 we draw some conclusions and provide future directions of research. At the end of each section we provide simulation data that illustrate the main result presented in the section.

## 2. Trajectory Tracking and Collision Avoidance

In this section we consider a non-holonomic mobile robot for which we want to design a controller that guarantees tracking with bounded error of a reference trajectory while avoiding collisions with static objects as well as other mobile robots in a cooperative fashion. We address the obstacle avoidance and collaborative collision avoidance separately.

The robot is modeled by the following non-linear ordinary differential equations (ODEs)

$$\begin{aligned} \dot{x} &= v\cos(\theta), \\ \dot{y} &= v\sin(\theta), \\ \dot{\theta} &= u, \end{aligned} \tag{1}$$

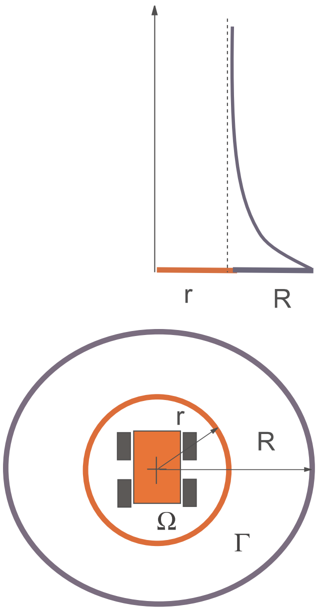


Fig. 1. Top: Side section of the avoidance function around the robot. Bottom: Avoidance (radius $r$) and detection (radius $R$) regions around the robot.

where $x \in \mathbb{R}$ and $y \in \mathbb{R}$ are the Cartesian coordinates, $\theta \in [0, 2\pi)$ is the orientation of the robot with respect to the world frame and $v, u$ are the linear and angular velocity inputs, respectively. We are also given a reference trajectory, $x_{\mathrm{d}}$, $y_{\mathrm{d}}$, with bounded derivative. Define the position errors as $e_x = x - x_{\mathrm{d}}$ and $e_y = y - y_{\mathrm{d}}$, the coordinates of the object to be avoided as $x_{\mathrm{a}}$, $y_{\mathrm{a}}$ and a distance function

$$d_{\mathrm{a}} = \sqrt{\left(\frac{x - x_{\mathrm{a}}}{\alpha}\right)^2 + \left(\frac{y - y_{\mathrm{a}}}{\beta}\right)^2}$$

for $\alpha, \beta > 0$. We address the obstacle avoidance problem using the following avoidance function (cf. Stipanović et al. (2007), Leitmann and Skowronski (1977) and Leitmann (1980))) defined for $\alpha = \beta = 1$

$$V_{\mathrm{a}} = \left(\min\left\{0, \frac{d_{\mathrm{a}}^2 - R^2}{d_{\mathrm{a}}^2 - r^2}\right\}\right)^2,$$

where $r > 0$ and $R > 0$, with $R > r$, are the radii of the avoidance and detection regions, respectively, which are defined as

$$\begin{aligned}\Omega &= \{[x \quad y] : (x, y) \in \mathbb{R}^2, \|[x \quad y]^{\mathrm{T}} - [x_{\mathrm{a}} \quad y_{\mathrm{a}}]^{\mathrm{T}}\| \leq r\}\\ \Gamma &= \{[x \quad y] : [x \quad y] \notin \Omega, r < \|[x \quad y]^{\mathrm{T}} - [x_{\mathrm{a}} \quad y_{\mathrm{a}}]^{\mathrm{T}}\| \leq R\}\end{aligned}$$

and represented in Figure 1.

This function is infinite at the boundary of the avoidance region and is zero outside the detection region. To break the symmetry when it occurs different shapes of the potential function, for example ellipsoids, can be obtained by choosing different values for the coefficients $\alpha, \beta$. Upon taking the partial derivatives of $V_{\mathrm{a}}$ with respect to the $x$ and $y$ coordinates, we obtain

$$\frac{\partial V_{\mathrm{a}}}{\partial x} = \begin{cases} 0, & \text{if} \quad d_{\mathrm{a}} \geq R \\ 4\dfrac{(R^2 - r^2)(d_{\mathrm{a}}^2 - R^2)}{(d_{\mathrm{a}}^2 - r^2)^3}(x - x_{\mathrm{a}}), & \text{if} \quad R > d_{\mathrm{a}} > r \\ 0, & \text{if} \quad d_{\mathrm{a}} < r \end{cases}$$

and

$$\frac{\partial V_{\mathrm{a}}}{\partial y} = \begin{cases} 0, & \text{if} \quad d_{\mathrm{a}} \geq R \\ 4\dfrac{(R^2 - r^2)(d_{\mathrm{a}}^2 - R^2)}{(d_{\mathrm{a}}^2 - r^2)^3}(y - y_{\mathrm{a}}), & \text{if} \quad R > d_{\mathrm{a}} > r \\ 0, & \text{if} \quad d_{\mathrm{a}} < r. \end{cases}$$

Let us define

$$\begin{aligned} E_x &= e_x + \frac{\partial V_{\mathrm{a}}}{\partial x} \\ E_y &= e_y + \frac{\partial V_{\mathrm{a}}}{\partial y}, \end{aligned}$$

for $(E_x, E_y) \neq (0, 0)$, the desired orientation

$$\theta_{\mathrm{d}} = \mathrm{Atan2}(-E_y, -E_x) \tag{2}$$

and the orientation error $e_\theta = \theta - \theta_{\mathrm{d}}$. Note that $\theta_{\mathrm{d}}$ defines a desired direction of motion that depends on the reference trajectory, the robot position and the obstacle to be avoided by the robot. However, some configurations might lead to singular directions as explained later. In order to avoid singular cases, we assume the following throughout the paper:

**Assumption 1.** The reference trajectory is smooth and satisfies

$$|e_\theta| \neq \frac{\pi}{2}. \tag{3}$$

**Assumption 2.** The reference trajectory remains constant inside the detection region, i.e. $\dot{x}_{\mathrm{d}} = \dot{y}_{\mathrm{d}} = 0$, for $r \leq d_{\mathrm{a}} < R$. This means that as the robot detects an obstacle in its path, it momentarily freezes its reference to the last data received, while trying to resolve the collision. Once it is outside the collision region, it updates the reference to the new values. The reason for this choice is that collision avoidance has a higher priority than tracking, as collision among robots could lead to system damage, which is more critical than temporary degeneration of tracking performance.

**Assumption 3.** Define $\hat{\dot{\theta}}_{\mathrm{d}}$ as

$$\hat{\dot{\theta}}_{\mathrm{d}} = \frac{E_x(t)\hat{\dot{E}}_y - E_y(t)\hat{\dot{E}}_x}{D^2},$$

where

$$\begin{aligned} \hat{\dot{E}}_x &= \frac{E_x(t+T) - E_x(t)}{T} \\ \hat{\dot{E}}_y &= \frac{E_y(t+T) - E_y(t)}{T} \end{aligned}$$

for some small $T > 0$. As such, $\hat{\dot{\theta}}_{\mathrm{d}}$ is a sufficiently smooth estimate of

$$\dot{\theta}_{\mathrm{d}} = \frac{E_x\dot{E}_y - \dot{E}_x E_y}{D^2},$$

where $D = \sqrt{E_x^2 + E_y^2} > 0$. Then, we assume that

$$\left|\hat{\dot{\theta}}_{\mathrm{d}} - \dot{\theta}_{\mathrm{d}}\right| \leq \epsilon_\theta$$

for some small positive $\epsilon_\theta$. Note that most of the variables in $\dot{\theta}_{\mathrm{d}}$ can be measured, in fact we have that

$$\left|\hat{\dot{\theta}}_{\mathrm{d}} - \dot{\theta}_{\mathrm{d}}\right| = \frac{E_x\left(\dot{E}_y - \hat{\dot{E}}_y\right) - E_y\left(\dot{E}_x - \hat{\dot{E}}_x\right)}{D^2},$$

where $E_y, E_x, D$ can be computed using the state measurements and desired values. Then, as $E_y, E_x$ are smooth almost everywhere, we have that $(\dot{E}_x - \hat{\dot{E}}_x) \simeq (\dot{E}_y - \hat{\dot{E}}_y) \simeq o(T)$ and we can pick $\epsilon_\theta \simeq o(T)$.

**Remark 1.** Assumption 1 on the reference trajectory implies the following two conditions.

1. Outside the detection region ($d_{\mathrm{a}} \geq R$) and for $(e_x, e_y) \neq (0, 0)$ we have $\theta_{\mathrm{d}} = \mathrm{Atan2}(-e_y, -e_x)$. The reference trajectory is such that it does not initiate sharp turns of 90° with respect to the current orientation of the robot. Note that this condition is not too restrictive since the robot can reorient itself in place if the condition is not satisfied, and smoothness of the reference trajectory is a reasonable assumption in the case of robots subject to non-holonomic constraints.

2. Inside the detection region ($r \leq d_{\mathrm{a}} < R$) we have

$$\theta_{\mathrm{d}} = \mathrm{Atan2}\left(-e_y - \frac{\partial V_{\mathrm{a}}}{\partial y}, -e_x - \frac{\partial V_{\mathrm{a}}}{\partial x}\right).$$

The combination of obstacle position and reference trajectory might drive the robot in a singular configuration where Assumption 1 does not hold. One solution is to consider a perturbed desired orientation $\bar{\theta}_{\mathrm{d}}$, instead of the desired orientation $\theta_{\mathrm{d}}$, whenever (Assumption 1) is not satisfied. In particular, each robot can modify the desired orientation so that, whenever Assumption 1 is not satisfied, $\theta_{\mathrm{d}}$ is replaced with the following perturbed version:

$$\bar{\theta}_{\mathrm{d}} = \theta_{\mathrm{d}} + \bar{\epsilon}_\theta,$$

where $\bar{\epsilon}_\theta \neq 0$ is some small perturbation value. This condition guarantees that the system avoids singularities and deadlocks.

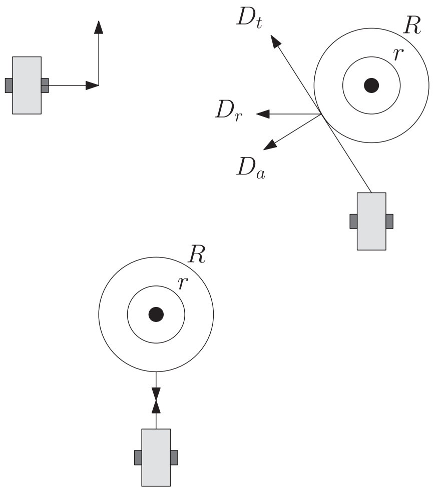


Fig. 2. Examples of non-admissible trajectories which lead to violation of the non-holonomic constraints (top) and a deadlock (bottom).

In Figure 2 two examples of non-admissible (singular) directions are shown. In the top-left corner a non-holonomic agent is shown in an obstacle free space the direction of motion defined by the arrow is not admissible because it violates the non-holonomic constraints. The top-right figure shows a non-holonomic agent approaching an obstacle: $D_{\mathrm{t}}$ is the direction required by the reference trajectory, $D_{\mathrm{a}}$ is the avoidance direction and $D_{\mathrm{r}}$ is the resulting direction which is not admissible since it violates the non-holonomic constraints.

**Remark 2.** The singularity condition $E_x = E_y = 0$ can occur in the following two cases. The first case occurs outside the detection region where

$$\frac{\partial V_{\mathrm{a}}}{\partial y} = \frac{\partial V_{\mathrm{a}}}{\partial x} = 0,$$

which corresponds to $e_x = e_y = 0$ and this case can easily be handled using zero controllers $u = v = 0$. The second case occurs inside the detection region where the condition corresponds to a singularity in which the reference direction vector for tracking is opposite but of equal magnitude to the direction vector for avoiding collision: this results in a deadlock as shown in the bottom of Figure 2. This case can be handled by changing the reference trajectory to drive the robot out of the singularity. We do not investigate this case further in this paper.

### 2.1. Static Obstacle Avoidance

First we study the case in which the obstacle is a static object in the plane and hence $x_\mathrm{a}$, $y_\mathrm{a}$ are constant values. In the setting described above we design a control law to achieve tracking of the reference trajectory while avoiding collision with the object.

**Theorem 1.** *Consider system (1) and the reference trajectory described by* $(x_\mathrm{d}, y_\mathrm{d})$ *that satisfies Assumptions 1–3. Consider also a static object to be avoided that is located at* $(x_\mathrm{a}, y_\mathrm{a})$. *Define the desired orientation as in (2). Then tracking with bounded error, outside the detection region, and collision avoidance are guaranteed if the following controller is applied*

$$\begin{aligned} u &= -K_\theta e_\theta + \hat{\dot{\theta}}_\mathrm{d} && (4)\\ v &= -K\cos(e_\theta)D && (5)\end{aligned}$$

*for all gains* $K, K_\theta > 0$ *and* $D = \sqrt{E_x^2 + E_y^2} > 0$. *Moreover, the tracking error can be reduced by increasing the value of the gains.*

Note that the singular case $D = 0$ is handled as explained in Remark 2.

**Proof.** Consider the error dynamics

$$\begin{aligned} \dot{e}_x &= v(\cos(e_\theta)\cos(\theta_\mathrm{d}) - \sin(e_\theta)\sin(\theta_\mathrm{d})) - \dot{x}_\mathrm{d},\\ \dot{e}_y &= v(\sin(e_\theta)\cos(\theta_\mathrm{d}) + \cos(e_\theta)\sin(\theta_\mathrm{d})) - \dot{y}_\mathrm{d},\\ \dot{e}_\theta &= u - \dot{\theta}_\mathrm{d}.\end{aligned}$$

Using the expressions

$$\cos(\theta_\mathrm{d}) = \frac{E_x}{D} \quad \text{and} \quad \sin(\theta_\mathrm{d}) = \frac{E_y}{D}$$

and applying the controller (4)–(6) we obtain

$$\begin{aligned} \dot{e}_x &= K(-E_x\cos^2(e_\theta) + E_y\cos(e_\theta)\sin(e_\theta)) - \dot{x}_\mathrm{d}\\ \dot{e}_y &= K(-E_x\cos(e_\theta)\sin(e_\theta) - E_y\cos^2(e_\theta)) - \dot{y}_\mathrm{d}\\ \dot{e}_\theta &= -K_\theta e_\theta + \hat{\dot{\theta}}_\mathrm{d} - \dot{\theta}_\mathrm{d}.\end{aligned}$$

Let us pick the Lyapunov-like function candidate as

$$\begin{aligned} V &= V_\mathrm{t} + V_\mathrm{a}\\ &= \frac{1}{2}(e_x^2 + e_y^2 + e_\theta^2) + \left(\min\left\{0, \frac{d_\mathrm{a}^2 - R^2}{d_\mathrm{a}^2 - r^2}\right\}\right)^2.\end{aligned}$$

Since $V$ is differentiable almost everywhere except at a measure zero set on the boundary of the avoidance region, we can evaluate the derivative along the trajectories of the error dynamics in the region $\Gamma$ as follows:

$$\begin{aligned} \frac{\mathrm{d}V}{\mathrm{d}t} &= e_x\dot{e}_x + e_y\dot{e}_y + e_\theta\dot{e}_\theta + \frac{\partial V_a}{\partial x}\dot{x} + \frac{\partial V_a}{\partial y}\dot{y}\\ &\le -K\cos^2(e_\theta)(E_x^2 + E_y^2) - e_x\dot{x}_\mathrm{d} - e_y\dot{y}_\mathrm{d}\\ &- \|e_\theta\|(K_\theta\|e_\theta\| - \epsilon_\theta). \qquad (6)\end{aligned}$$

When the robot is outside the detection region ($d_\mathrm{a} > R$), we have

$$\frac{\partial V_\mathrm{a}}{\partial x} = \frac{\partial V_\mathrm{a}}{\partial y} = 0,$$

and the previous inequality becomes

$$\begin{aligned} \frac{\mathrm{d}V}{\mathrm{d}t} &\le -K\cos^2(e_\theta)(e_x^2 + e_y^2) - e_x\dot{x}_\mathrm{d} - e_y\dot{y}_\mathrm{d}\\ &- \|e_\theta\|(K_\theta\|e_\theta\| - \epsilon_\theta),\\ &= -\begin{bmatrix} e_x\\ e_y\\ e_\theta\end{bmatrix}^\mathrm{T} M \begin{bmatrix} e_x\\ e_y\\ e_\theta\end{bmatrix} - \begin{bmatrix} e_x\\ e_y\\ \|e_\theta\|\end{bmatrix}^\mathrm{T} \begin{bmatrix} \dot{x}_\mathrm{d}\\ \dot{y}_\mathrm{d}\\ -\epsilon_\theta\end{bmatrix},\\ &\le -\begin{bmatrix} e_x\\ e_y\\ e_\theta\end{bmatrix}^\mathrm{T} M \begin{bmatrix} e_x\\ e_y\\ e_\theta\end{bmatrix} + \left\|\begin{bmatrix} e_x\\ e_y\\ e_\theta\end{bmatrix}\right\| \left\|\begin{bmatrix} \dot{x}_\mathrm{d}\\ \dot{y}_\mathrm{d}\\ -\epsilon_\theta\end{bmatrix}\right\|,\end{aligned}$$

where

$$M = \begin{bmatrix} K\cos^2(e_\theta) & 0 & 0\\ 0 & K\cos^2(e_\theta) & 0\\ 0 & 0 & K_\theta\end{bmatrix}$$

and from Assumption 1 we have that $\cos^2(e_\theta) > 0$. Hence, $\mathrm{d}V/\mathrm{d}t < 0$ whenever

$$\|e\| > \frac{\|d\|}{\lambda_{\min}(M)}, \qquad (7)$$

where $e = [e_x \;\; e_y \;\; e_\theta]^\mathrm{T}$ and $d = [\dot{x}_\mathrm{d} \;\; \dot{y}_\mathrm{d} \;\; \epsilon_\theta]$. Therefore, the stability of the error dynamics, and hence tracking with bounded error, are guaranteed outside the detection region. Moreover, by increasing the gains $K, K_\theta$ we can decrease the tracking error.

When the robot is inside the detection region ($r \le d_\mathrm{a} < R$), Assumption 2 implies $\dot{x}_\mathrm{d} = \dot{y}_\mathrm{d} = 0$. Inequality (6) becomes

$$\frac{\mathrm{d}V}{\mathrm{d}t} \le -K\cos^2(e_\theta)D^2 - \|e_\theta\|(K_\theta\|e_\theta\| - \epsilon_\theta),$$

which is negative definite for

$$\|e_\theta\| > \frac{\epsilon_\theta}{K_\theta}.$$

Hence, as shown by Stipanović et al. (2007), since $\mathrm{d}V/\mathrm{d}t$ is negative definite, then $V$ is non-increasing inside the detection region. Since

$$\lim_{\|z-z_{\mathrm{a}}\|\to r^+} V_a = \infty, \tag{8}$$

where $z = [x \quad y]^{\mathrm{T}}$, $z_{\mathrm{a}} = [x_{\mathrm{a}} \quad y_{\mathrm{a}}]^{\mathrm{T}}$, then collision avoidance is guaranteed. ■

**Remark 3.** The previous theorem can be extended to include multiple obstacles by defining avoidance functions for each obstacle and appending them to the total Lyapunov-like function $V$. The total avoidance and detection regions are defined as the union of avoidance and detection regions of all of the obstacles, see Stipanović et al. (2007) for more details. As for the case of single robot, we can address the issue of local minimum, which leads to a deadlock, by perturbing the reference signal.

### *2.2. Cooperative Collision Avoidance*

Next we consider a scenario in which two robots are tracking a reference trajectories and meanwhile avoiding collision with each other in a cooperative fashion. The robot state coordinates are denoted by $(x_1, y_1, \theta_1)$, $(x_2, y_2, \theta_2)$ respectively. The obstacle coordinates for each robot are the coordinates of the other robot and are described by $x_{\mathrm{a}i} = x_j$, $y_{\mathrm{a}i} = y_j$, $j, i = 1, 2, i \neq j$. Each robot has circular detection and avoidance regions defined previously and characterized by radii $R$ and $r$, respectively. The next theorem shows how the same control law as in the static case achieves tracking and collision avoidance between robots.

**Theorem 2.** *Consider two systems of the form (1) and their reference trajectories described by* $(x_{\mathrm{d}i}, y_{\mathrm{d}i})$, $i = 1, 2$ *that satisfy Assumptions 1–3. Define the desired orientation for each system as in (2). Then, tracking with bounded error outside the detection region and collision avoidance are guaranteed if the following controllers are applied:*

$$\begin{aligned} u_i &= -K_{\theta i} e_{\theta i} + \hat{\dot{\theta}}_{\mathrm{d}i} \qquad (9)\\ v_i &= -K_i \cos(e_{\theta i}) D_i \qquad (10) \end{aligned}$$

*for all gains* $K_i, K_{\theta i} > 0$ *and* $D_i = \sqrt{E_{xi}^2 + E_{yi}^2} > 0$, $i = 1, 2$. *Moreover the tracking error can be reduced by increasing the value of the gains.*

**Proof.** Consider the error dynamics for $i = 1, 2$

$$\begin{aligned} \dot{e}_{xi} &= v_i(\cos(e_{\theta i})\cos(\theta_{\mathrm{d}i}) - \sin(e_{\theta i})\sin(\theta_{\mathrm{d}i})) - \dot{x}_{\mathrm{d}i},\\ \dot{e}_{yi} &= v_i(\sin(e_{\theta i})\cos(\theta_{\mathrm{d}i}) + \cos(e_{\theta i})\sin(\theta_{\mathrm{d}i})) - \dot{y}_{\mathrm{d}i},\\ \dot{e}_{\theta i} &= u_i - \dot{\theta}_{\mathrm{d}i}. \end{aligned}$$

Applying the controller (9)–(10), we can rewrite the error dynamics as follows:

$$\begin{aligned} \dot{e}_{xi} &= K_i(-E_{xi}\cos^2(e_{\theta i}) + E_{yi}\cos(e_{\theta i})\sin(e_{\theta i})) - \dot{x}_{\mathrm{d}i},\\ \dot{e}_{yi} &= K_i(-E_{xi}\cos(e_{\theta i})\sin(e_{\theta i}) - E_{yi}\cos^2(e_{\theta i})) - \dot{y}_{\mathrm{d}i},\\ \dot{e}_{\theta i} &= -K_{\theta i} e_{\theta i} + \hat{\dot{\theta}}_{\mathrm{d}i} - \dot{\theta}_{\mathrm{d}i}. \end{aligned}$$

We define the avoidance function for each robot as follows:

$$V_{\mathrm{a}i} = \left(\min\left\{0, \frac{d_{\mathrm{a}i}^2 - R^2}{d_{\mathrm{a}i}^2 - r^2}\right\}\right)^2,$$

where

$$d_{\mathrm{a}i} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

for $i, j = 1, 2, i \neq j$.

We pick the Lyapunov-like function candidate for the system comprised of two robots as follows:

$$\begin{aligned} V &= \frac{1}{2}(V_{t1} + V_{\mathrm{a}1} + V_{t2} + V_{\mathrm{a}2}),\\ &= \frac{1}{2}\left((e_{x1}^2 + e_{y1}^2 + e_{\theta 1}^2) + \left(\min\left\{0, \frac{d_{\mathrm{a}1}^2 - R^2}{d_{\mathrm{a}1}^2 - r^2}\right\}\right)^2\right)\\ &+ \frac{1}{2}\left((e_{x2}^2 + e_{y2}^2 + e_{\theta 2}^2) + \left(\min\left\{0, \frac{d_{\mathrm{a}2}^2 - R^2}{d_{\mathrm{a}2}^2 - r^2}\right\}\right)^2\right). \end{aligned}$$

Note that

$$\frac{\partial V_{\mathrm{a}1}}{\partial x_1} = -\frac{\partial V_{\mathrm{a}1}}{\partial x_2} = -\frac{\partial V_{\mathrm{a}2}}{\partial x_2} = \frac{\partial V_{\mathrm{a}2}}{\partial x_1}.$$

Then, the derivative along the trajectories of the error dynamics in the region $\Gamma = \Gamma_1 \bigcup \Gamma_2$ is

$$
\begin{aligned}
\frac{\mathrm{d}V}{\mathrm{d}t} &= e_{x1}\dot{e}_{x1} + e_{y1}\dot{e}_{y1} + e_{\theta 1}\dot{e}_{\theta 1} \\
&+ \frac{1}{2}\left(\frac{\partial V_{\mathrm{a1}}}{\partial x_1}\dot{x}_1 + \frac{\partial V_{\mathrm{a1}}}{\partial x_2}\dot{x}_2 + \frac{\partial V_{\mathrm{a1}}}{\partial y_1}\dot{y}_1 + \frac{\partial V_{\mathrm{a1}}}{\partial y_2}\dot{y}_2\right) \\
&+ e_{x2}\dot{e}_{x2} + e_{y2}\dot{e}_{y2} + e_{\theta 2}\dot{e}_{\theta 2} \\
&+ \frac{1}{2}\left(\frac{\partial V_{\mathrm{a2}}}{\partial x_2}\dot{x}_2 + \frac{\partial V_{\mathrm{a2}}}{\partial x_1}\dot{x}_1 + \frac{\partial V_{\mathrm{a2}}}{\partial y_2}\dot{y}_2 + \frac{\partial V_{\mathrm{a2}}}{\partial y_1}\dot{y}_1\right) \\
&= \left(e_{x1} + \frac{\partial V_{\mathrm{a1}}}{\partial x_1}\right)\dot{x}_1 + \left(e_{y1} + \frac{\partial V_{\mathrm{a1}}}{\partial y_1}\right)\dot{y}_1 \\
&+ \left(e_{x2} + \frac{\partial V_{\mathrm{a2}}}{\partial x_2}\right)\dot{x}_2 + \left(e_{y2} + \frac{\partial V_{\mathrm{a2}}}{\partial y_2}\right)\dot{y}_2 \\
&- e_{x1}\dot{x}_{\mathrm{d1}} - e_{y1}\dot{y}_{\mathrm{d1}} - e_{x2}\dot{x}_{\mathrm{d2}} - e_{y2}\dot{y}_{\mathrm{d2}} \\
&+ e_{\theta 1}\dot{e}_{\theta 1} + e_{\theta 2}\dot{e}_{\theta 2}.
\end{aligned}
$$

Substituting the closed-loop robots dynamics, we obtain

$$
\begin{aligned}
\frac{\mathrm{d}V}{\mathrm{d}t} &\leq -K_1\cos^2(e_{\theta 1})\left(E_{x1}^2 + E_{y1}^2\right) - e_{x1}\dot{x}_{\mathrm{d1}} - e_{y1}\dot{y}_{\mathrm{d1}} \\
&- \|e_{\theta 1}\|(K_{\theta 1}\|e_{\theta 1}\| - \epsilon_{\theta 1}) \\
&- K_2\cos^2(e_{\theta 2})\left(E_{x2}^2 + E_{y2}^2\right) - e_{x2}\dot{x}_{\mathrm{d2}} - e_{y2}\dot{y}_{\mathrm{d2}} \\
&- \|e_{\theta 2}\|(K_{\theta 2}\|e_{\theta 2}\| - \epsilon_{\theta 2}). \qquad (11)
\end{aligned}
$$

When the robots are outside each other's detection regions ($d_{\mathrm{a}i} > R, i = 1, 2$), we have

$$
\frac{\partial V_{\mathrm{a}i}}{\partial x_j} = \frac{\partial V_{\mathrm{a}i}}{\partial y_j} = 0, \quad i, j = 1, 2,
$$

i.e. $E_{xi} = e_{xi}$, $E_{yi} = e_{yi}$, and the previous inequality becomes

$$
\begin{aligned}
\frac{\mathrm{d}V}{\mathrm{d}t} &\leq -e_1^T M_1 e_1 + \|e_1\|\|d_1\| - e_2^T M_2 e_2 + \|e_2\|\|d_2\| \\
&= -\begin{bmatrix} e_1 \\ e_2 \end{bmatrix}\begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix}\begin{bmatrix} e_1 \\ e_2 \end{bmatrix} + \left\|\begin{bmatrix} e_1 \\ e_2 \end{bmatrix}\right\|\left\|\begin{bmatrix} d_1 \\ d_2 \end{bmatrix}\right\|,
\end{aligned}
$$

where

$$
M_1 = \begin{bmatrix} K_1\cos^2(e_{\theta 1}) & 0 & 0 \\ 0 & K_1\cos^2(e_{\theta 1}) & 0 \\ 0 & 0 & K_{\theta 1} \end{bmatrix}
$$

$$
M_2 = \begin{bmatrix} K_2\cos^2(e_{\theta 2}) & 0 & 0 \\ 0 & K_2\cos^2(e_{\theta 2}) & 0 \\ 0 & 0 & K_{\theta 2} \end{bmatrix}
$$

and

$$
e_1 = \begin{bmatrix} e_{x1} \\ e_{y1} \\ e_{\theta 1} \end{bmatrix}, \quad e_2 = \begin{bmatrix} e_{x2} \\ e_{y2} \\ e_{\theta 2} \end{bmatrix},
$$

$$
d_1 = \begin{bmatrix} \dot{x}_{\mathrm{d1}} \\ \dot{y}_{\mathrm{d1}} \\ \epsilon_{\theta 1} \end{bmatrix}, \quad d_2 = \begin{bmatrix} \dot{x}_{\mathrm{d2}} \\ \dot{y}_{\mathrm{d2}} \\ \epsilon_{\theta 2} \end{bmatrix}.
$$

Hence, $\mathrm{d}V/\mathrm{d}t < 0$ whenever

$$
\|[e_1 \quad e_2]^{\mathrm{T}}\| > \frac{\|[d_1 \quad d_2]^{\mathrm{T}}\|}{\min\{\lambda_{\min}(M_1), \lambda_{\min}(M_2)\}}
$$

Therefore stability of the error dynamics, and hence tracking with bounded error, is guaranteed outside the detection region.

When the robot is inside the detection region ($r \leq d_{\mathrm{a}} < R$), Assumption 2 implies $\dot{x}_{\mathrm{d}} = \dot{y}_{\mathrm{d}} = 0$ hence, inequality (11) becomes

$$
\frac{\mathrm{d}V}{\mathrm{d}t} \leq \sum_{i=1}^{2} -K_i\cos^2(e_{\theta i})D_i^2 - \|e_{\theta i}\|(K_{\theta i}\|e_{\theta i}\| - \epsilon_{\theta i}),
$$

which is negative definite for

$$
\|[e_{\theta 1} \quad e_{\theta_2}]^{\mathrm{T}}\| > \frac{\|[\epsilon_{\theta 1} \quad \epsilon_{\theta 2}]^{\mathrm{T}}\|}{\min_i\{K_{\theta i}\}}.
$$

Hence, as shown by Stipanović et al. (2007), since $\mathrm{d}V/\mathrm{d}t$ is negative definite, then $V$ is non-increasing inside the detection region, and since

$$
\lim_{\|x_1 - x_2\| \to r^+} V_{\mathrm{a}i} = \infty, \quad i = 1, 2, \qquad (12)
$$

then collision avoidance is guaranteed. ■

### 2.3. Simulation Results: Obstacle and Collision Avoidance

To illustrate the result we consider a unicycle in the $X$–$Y$ plane for which the objective is to track a circle centered at (9, 9) of radius 6 while avoiding collision with an obstacle located in the plane. The initial conditions for the robot are $x_0 = y_0 = 1$, $\theta_0 = \pi$. The reference trajectory is given to the agents as

$$
\begin{aligned}
x_{\mathrm{d}} &= 9 + 6\cos(0.11t) \\
y_{\mathrm{d}} &= 9 + 6\sin(0.11t)
\end{aligned}
$$

The obstacle is at $x_{\mathrm{a}} = 4$, $y_{\mathrm{a}} = 4.5$ and the detection and avoidance radius are $R = 3$, $r = 1$, respectively. Two sequential frames of the unicycle trajectory resulting by applying the controller (4) are represented in Figures 3 and 4.

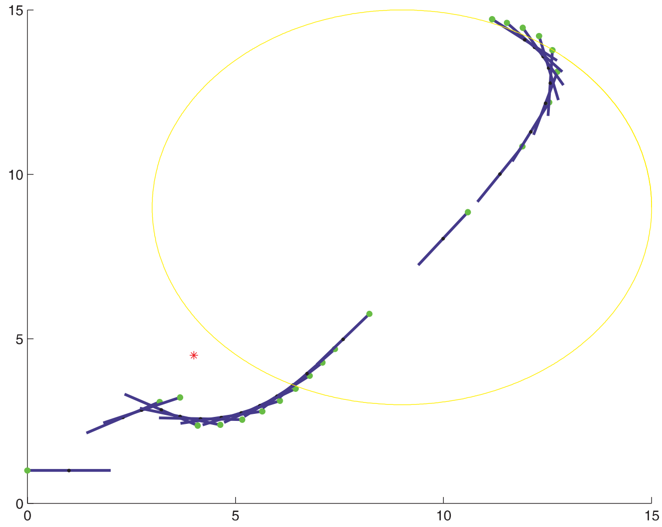


Fig. 3. Unicycle tracking a circular trajectory (time dependent) while avoiding collision with an obstacle (denoted by *) located at (4.5, 4.5). The coordinates of the robot initial conditions are (1,1).

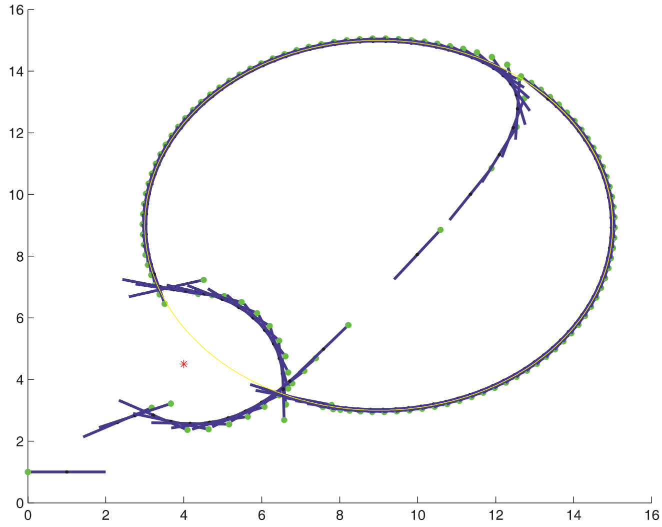


Fig. 4. After tracking the first part of the circle, the robot again detects the obstacle on its path and activates the collision avoidance control. Finally, after the obstacle is outside the robot's detection region the trajectory tracking is resumed.

In the second simulation we consider a scenario where the objective is to achieve collision avoidance between robots, when multiple agents are operating in the same area. In Figures 5 and 6 the trajectories of two robots tracking two circles of radius 4.5 centered at (5, 10) and (15, 10) are depicted, respectively. When the reference trajectories get close, the robots deviate from their paths in order to avoid collisions. The detection and avoidance radius are $R = 3$, $r = 1$, respectively.

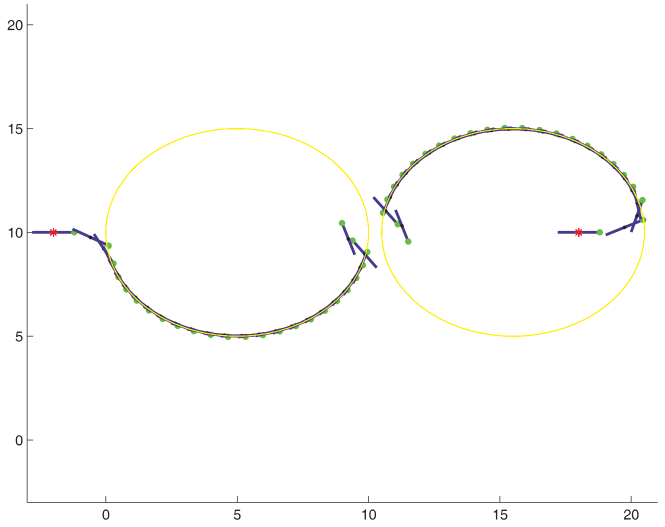


Fig. 5. Two unicycles initially tracking a given trajectory consisting of two circles, then entering each others detection regions and activating the avoidance control laws. The coordinates of the robots initial conditions are (−2, 10), (18, 10).

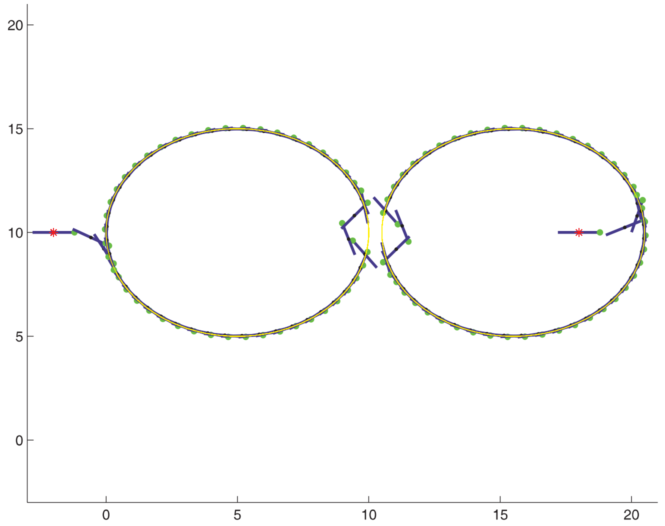


Fig. 6. Snapshot of the full trajectory with tracking and collision avoidance. The tracking is resumed after exiting the detection region.

## 3. Formation Control

In this section we study the problem of formation control for a group of non-holonomic agents. We consider a group of $N$

non-holonomic robots whose kinematic models are described by

$$\begin{aligned}\dot{x}_i &= u_i \cos(\theta_i),\\ \dot{y}_i &= u_i \sin(\theta_i), \qquad (13)\\ \dot{\theta}_i &= v_i,\end{aligned}$$

where $x_i \in \mathbb{R}$, $y_i \in \mathbb{R}$, $i = 1, \ldots, N$, describe the position of each robot in the plane, $\theta_i \in [0, 2\pi)$ is the heading angle and $u_i$ and $v_i$ are velocity inputs.

Given a desired formation and a desired trajectory for the center of mass of the formation, the objective is for the robots to converge to the formation and to follow the desired trajectory while maintaining the stability of the formation. Moreover we would like to achieve such an objective in a decentralized manner, i.e. the controller should be implemented locally on each agent. The desired formation can be described by a set of relative angles $\varphi_i$ and relative distances $d_i$, $i = 1, \ldots, N$, from each robot to the center of mass. Hence, a desired configuration for $N$ robots is described by $2N$ parameters $d_i, \varphi_i$, $i = 1, \ldots, N$. Then the formation control problem can be addressed in a trajectory tracking framework. Given a desired formation for $N$ robots, denoted by $(d_i, \varphi_i, i = 1, \ldots, N)$ and a desired trajectory $(x_{\mathrm{dc}}(t), y_{\mathrm{dc}}(t))$, $t \in R^+$, for the center of mass of the formation, we can define a desired trajectory for each robot as follows:

$$\begin{aligned}x_{\mathrm{d}i}(t) &= x_{\mathrm{dc}}(t) + d_i \cos(\varphi_i)\\ y_{\mathrm{d}i}(t) &= y_{\mathrm{dc}}(t) + d_i \sin(\varphi_i), \quad i = 1, \ldots, N.\end{aligned}$$

Then as each robot tracks its reference, i.e. $x_i(t) \to x_{\mathrm{d}i}(t)$, $y_i(t) \to y_{\mathrm{d}i}(t)$, $i = 1, \ldots, N$, and collision avoidance between robots is guaranteed, the robots will converge to the formation and follow the desired trajectory while maintaining the formation without colliding.

**Theorem 3.** *Consider a group of $N$ non-holonomic agents and a desired formation for the group so that the desired position of each robot is outside the detection regions. Consider also a desired trajectory for the center of mass of the formation that satisfies Assumptions 1–3. Then, the robots converge to the desired formation and track the desired trajectory for the center of mass while avoiding collisions, if the following controllers are applied:*

$$\begin{aligned}u_i &= -K_{\theta_i}(e_{\theta i}) + \hat{\dot{\theta}}_{\mathrm{d}i},\\ v_i &= -K_i \cos(e_{\theta i}) D_i, \qquad (14)\end{aligned}$$

*where $K_{\theta_i}$, $K_i$, $i = 1, 2, \ldots, N$, are arbitrary positive gains and $D_i = \sqrt{E_{x_i}^2 + E_{y_i}^2} > 0$. Moreover, the tracking and formation errors can be reduced by increasing the value of the gains.*

*The desired orientation is defined for $(|E_{x_i}|, |E_{y_i}|) \neq (0, 0)$ as*

$$\theta_{\mathrm{d}i} = \mathrm{Atan2}(-E_{y_i}, -E_{x_i}),$$

*where we define $E_{x_i}$ and $E_{y_i}$, as for the case of single robot. The avoidance functions are defined as*

$$V_{\mathrm{a}i} = \sum_{j=1,\, j\neq i}^{N-1} V_{\mathrm{a}ij}$$

*where*

$$V_{\mathrm{a}ij} = \left(\min\left\{0, \frac{d_{\mathrm{a}ij}^2 - R^2}{d_{\mathrm{a}ij}^2 - r^2}\right\}\right)^2$$

*and*

$$d_{\mathrm{a}ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, \quad j \neq i,\ j = 1, \ldots, N-1.$$

Note that each agent might potentially collide with the remaining $N - 1$ agents therefore, for each agent $i$ we define $N - 1$ avoidance functions $V_{ij}$, $j = 1, \ldots N - 1$, that correspond to the other $N - 1$ agents.

**Proof.** Consider the following Lyapunov-like function candidate

$$V = \sum_{i=1}^{N}\left[\left(e_{xi}^2 + e_{yi}^2 + e_{\theta i}^2\right) + \sum_{j\neq i, j=1}^{N-1} \frac{1}{2} V_{\mathrm{a}ij}\right].$$

If we calculate the derivative of the function $V$ along the trajectories of the system of robots we obtain

$$\begin{aligned}\frac{\mathrm{d}V}{\mathrm{d}t} &\leq \sum_{i=1}^{N}\left[-K_i \cos^2(e_{\theta_i})\left(E_{xi}^2 + E_{yi}^2\right) + e_{xi}\|\dot{x}_{\mathrm{d}i}\|\right.\\ &+ \left. e_{yi}\|\dot{y}_{\mathrm{d}i}\| - \|e_{\theta_i}\|(K_{\theta_i}\|e_{\theta i}\| - \epsilon_{\theta i})\right]. \qquad (15)\end{aligned}$$

When all of the robots are outside each other's detection regions the previous inequality becomes

$$\begin{aligned}\frac{\mathrm{d}V}{\mathrm{d}t} &= -\begin{bmatrix} e_1\\ \vdots\\ e_N\end{bmatrix}\begin{bmatrix} M_1 & 0 & \ldots & 0\\ \vdots & \vdots & \vdots & \vdots\\ 0 & \ldots & 0 & M_n\end{bmatrix}\begin{bmatrix} e_1\\ \vdots\\ e_N\end{bmatrix}\\ &+ \left\|\begin{bmatrix} e_1\\ \vdots\\ e_N\end{bmatrix}\right\|\left\|\begin{bmatrix} d_1\\ \vdots\\ d_N\end{bmatrix}\right\|, \qquad (16)\end{aligned}$$

where

$$M_i = \begin{bmatrix} K_i \cos^2(e_{\theta 1}) & 0 & 0 \\ 0 & K_i \cos^2(e_{\theta i}) & 0 \\ 0 & 0 & K_{\theta i} \end{bmatrix}$$

and

$$e_i = \begin{bmatrix} e_{xi} \\ e_{yi} \\ e_{\theta i} \end{bmatrix}, \quad d_i = \begin{bmatrix} \dot{x}_{\mathrm{d}i} \\ \dot{y}_{\mathrm{d}i} \\ \epsilon_{\theta i} \end{bmatrix}, \quad i = 1, \ldots, N.$$

Hence, $\mathrm{d}V/\mathrm{d}t < 0$ whenever

$$\|[e_1 \quad \cdots \quad e_N]^{\mathrm{T}}\| > \frac{\|[d_1 \quad \cdots \quad d_2]^{\mathrm{T}}\|}{\min_i\{\lambda_{\min i}(M_i)\}}.$$

Therefore stability of the error dynamics, and hence formation tracking, is guaranteed outside the detection region.

When two or more robots approach each other and enter each other's detection regions ($r \leq d_{\mathrm{a}i} < R$), Assumption 2 implies $\dot{x}_{\mathrm{d}i} = \dot{y}_{\mathrm{d}i} = 0$ hence, inequality (15) becomes

$$\frac{\mathrm{d}V}{\mathrm{d}t} \leq \sum_{i=1}^{N} -K_i \cos^2(\theta_i) D_i^2 - \|e_{\theta_i}\| \left( \|e_{\theta i}\| - \frac{\epsilon_{\theta i}}{K_{\theta_i}} \right),$$

which is negative definite for

$$\|[e_{\theta 1} \quad \cdots \quad e_{\theta N}]\| > \frac{\|[\epsilon_{\theta 1} \quad \cdots \quad \epsilon_{\theta N}\|]}{\min_i\{K_{\theta i}\}}.$$

Since $\mathrm{d}V/\mathrm{d}t$ is negative definite, then $V$ is non-increasing inside the detection region, and since

$$\lim_{\|x_i - x_j\| \to r^+} V_{\mathrm{a}ij} = \infty, \quad i \neq j, \tag{17}$$

then collision avoidance is guaranteed. ■

### 3.1. Simulation Results

We consider a group of three robots in the $X$–$Y$ plane. The objective is to reach a triangular formation whose center of mass is at $x_{\mathrm{c}} = 6$, $y_{\mathrm{c}} = 14.5$ and from there the center of mass must follow a trajectory $x_{\mathrm{d}} = t/5$, $y_{\mathrm{d}} = 30\sin(2\pi 0.01 x_{\mathrm{d}})$. We simulated the robots dynamics and the resulting trajectories for the group are shown in Figures 7 and 8. In Figure 7 the three robots, starting at initial conditions $(x_1, y_1, \theta_1) = (0, 0, \pi/4)$, $(x_2, y_2, \theta_2) = (5, 0, 3\pi/4)$ and $(x_3, y_3, \theta_3) = (0, 10, 3\pi/4)$, converge to the desired formation.

In Figure 8, after the robots reach the desired formation, they follow the desired trajectory while maintaining the formation stable.

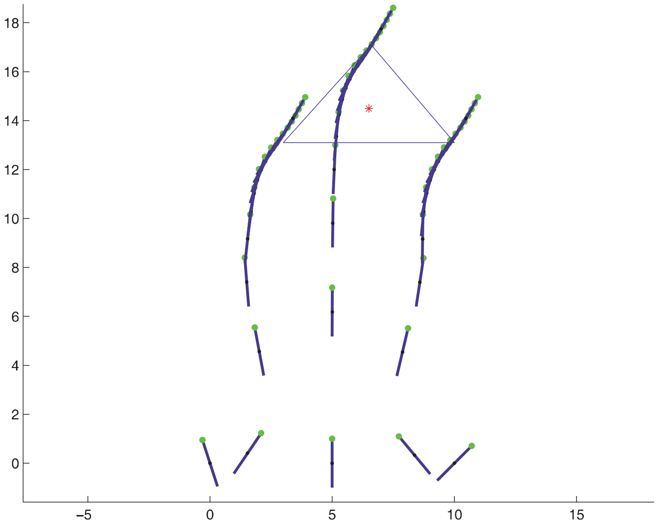


Fig. 7. Formation control for a group of three robots. Three robots starting from initial conditions: (0, 0), (5, 0) and (0, 10) first converge to the desired triangle formation centered at (6, 14).

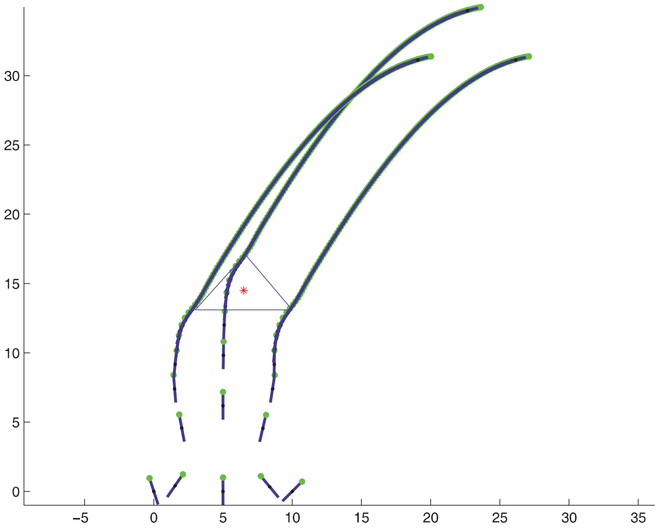


Fig. 8. Formation control for a group of three robots. Once the group reaches the formation they track, in a coordinated fashion, a specified trajectory.

**Remark 4.** Note that the leader–follower control problem is a special case of the formation control problem, in which we consider the leader instead of the formation center of mass as reference point. In fact, given a position for the leader and a relative desired position with respect to the leader, the follower can compute its desired position by properly scaling the leader position.

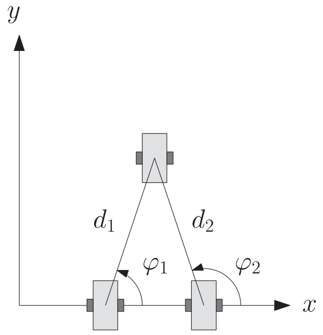


Fig. 9. Formation shape described as a relative distance and orientation of each follower with respect to the leader.

## 4. Formation Control and Coordinated Tracking

Our final objective is to have a supervisor drive, using a velocity command, a single leader robot and to have a group of follower robots maintain a desired formation with respect to the leader.

Consider a group of $N$ non-holonomic robots whose kinematic model is described by (13) with $i = 1, \ldots, N$. The set of $N$ robots is composed of one leader robot and a group of $N-1$ followers. A supervisor specifies a desired velocity and a desired shape (or formation) for the group. The desired bounded velocity, characterized by amplitude $V_\mathrm{d}$ and directional angle $\psi_\mathrm{d}$, is communicated to the leader from a supervisor. The desired formation is described by relative angle $\varphi_i$ and relative distance $d_i$ from each follower to the leader (see, for example, Figure 9). Each follower receives the formation parameters $(d_i, \varphi_i)$ from the supervisor and the leader position $x_0$, $y_0$ from the leader. In order to guarantee collision avoidance we also assume that each robot can sense the others in its proximity. In summary the supervisor provides the set of desired parameters $V_\mathrm{d}$, $\psi_\mathrm{d}$, $d_i$, $\varphi_i$, $i = 1, \ldots, N-1$.

We need to guarantee that the leader tracks the velocity commands $V_\mathrm{d}$, $\psi_\mathrm{d}$ given by the supervisor, and that the followers keep the desired formation specified by the parameters $d_i, \varphi_i, i = 1, \ldots, N-1$. The formation problem and collision avoidance can be addressed as in the previous section using the control laws (14). We address the velocity tracking in the following section.

### *4.1. Velocity Tracking*

Recall the kinematic model (1). Our objective in this section is to drive the velocity of the center of mass of the robot according to specified values of velocity magnitude and direction, which are given by $V_\mathrm{d}$ and $\psi_\mathrm{d}$ in polar coordinates or equivalently $\dot{x}_\mathrm{d}$, $\dot{y}_\mathrm{d}$ in Cartesian coordinates:

$$\begin{aligned} \dot{x}_\mathrm{d} &= V_\mathrm{d}\cos(\psi_\mathrm{d}), \\ \dot{y}_\mathrm{d} &= V_\mathrm{d}\sin(\psi_\mathrm{d}). \end{aligned}$$

Since the velocity input $u$ directly affects the output that we aim to drive $(\dot{x}, \dot{y})$ but not all of the directions are feasible owing to the non-holonomic constraints, then we need to guarantee that the robot orientation is aligned with the direction of desired motion, that is $\theta_\mathrm{d} = \psi_\mathrm{d}$, and define the error dynamics as

$$\begin{aligned} \dot{e}_\theta &= u - \dot{\theta}_\mathrm{d}, \\ \dot{e}_x &= v\cos(\theta) - V_\mathrm{d}\cos(\theta_\mathrm{d}), \\ \dot{e}_y &= v\sin(\theta) - V_\mathrm{d}\sin(\theta_\mathrm{d}). \end{aligned} \tag{18}$$

By applying the simple proportional controller

$$\begin{aligned} v &= V_\mathrm{d} \\ u &= -K(\theta - \theta_\mathrm{d}) \end{aligned} \tag{19}$$

with $\theta_\mathrm{d} = \psi_\mathrm{d}$ and some gain $K > 0$, we have the closed loop system

$$\begin{aligned} \dot{e}_x &= V_\mathrm{d}(\cos(e_\theta)\cos(\theta_\mathrm{d}) - \sin(e_\theta)\sin(\theta_\mathrm{d}) - \cos(\theta_\mathrm{d})), \\ \dot{e}_y &= V_\mathrm{d}(\sin(e_\theta)\cos(\theta_\mathrm{d}) + \cos(e_\theta)\sin(\theta_\mathrm{d}) - \sin(\theta_\mathrm{d})), \\ \dot{e}_\theta &= -K(\theta - \theta_\mathrm{d}) - \dot{\theta}_\mathrm{d}. \end{aligned} \tag{20}$$

It can be easily shown, using the ISS-Lyapunov function $V = \frac{1}{2}e_\theta^2$, that this system is input-to-state stable (ISS; see Sontag (2006)) with respect to the input $\dot{\theta}_\mathrm{d}$ and that the output is bounded for bounded values of the input $V_\mathrm{d}$. This guarantees that we can drive the velocity of the leader according to some desired bounded value using controllers (19).

**Remark 5.** Tanner (2004) proved a result that establishes the ISS property for non-holonomic systems of the form (1) with a dynamic extension, when tracking a given trajectory. However, since we are only interested in velocity tracking, then we only need it to be ISS with respect to the orientation.

### *4.2. Simulation Results*

We consider a group of unicycles composed of one master and two slaves. The master is driven by the following velocity command

$$V_\mathrm{d} = 1, \quad \psi = \begin{cases} \dfrac{\pi}{2}, & t \leq 5 \\ \dfrac{\pi}{4}, & 5 < t \leq 7 \cdot \\ 0, & t > 7 \end{cases} \tag{21}$$

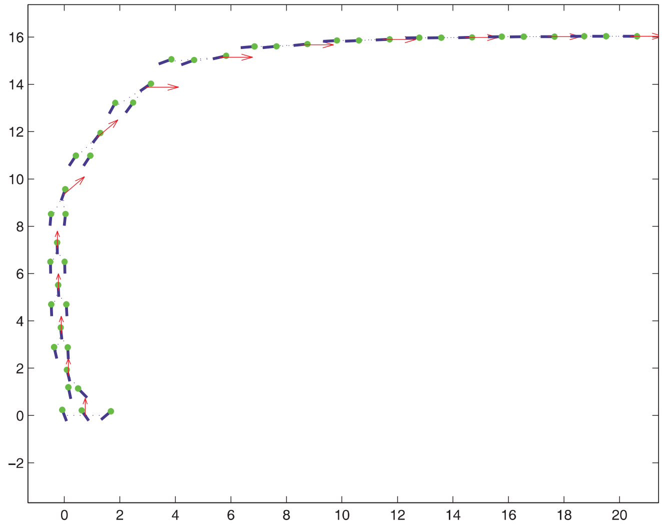


Fig. 10. Group of unicycles driven by velocity commands while changing formation.

A formation is specified so that in the first part of the trajectory ($t \leq 7$) the group has a "V" formation, while in the second part ($t > 7$) the robots are on a straight line.

The resulting trajectories of the unicycles are depicted in the upper part of Figure 10, where the arrow represents the velocity command.

## 5. Experimental Results

In order to validate the theoretical results, experiments were conducted on a robotic testbed. The completion of such experiments demonstrates the correctness and the ease with which the control algorithms can be implemented. In addition, successful results indicate a tolerance to disturbances such as numerical approximations and delays due to sampling or communications.

### *5.1. Experimental Setup*

Testing was carried out in the College of Engineering Mechatronics Laboratory at the University of Illinois at Urbana-Champaign. The workspace consists of a 3.66 m by 3.66 m enclosed region as shown in Figure 11. A color camera, located approximately 3.35 m above the floor, is used to detect robot positions and orientations. Image processing is carried out by a computer running a 3.4 GHz Pentium 4 processor with 1 GB of RAM. Communication between the robots and the computer uses an AeroComm 4490 wireless modem. These are capable of transferring data at a rate of 44 Kb/s. The robotic platform used in the experiments, called the SegBot, is shown in Figure 12. Onboard processing is carried out by a Texas Instruments 225 MHz TMS 320C6713 DSP. This is included as part of the C6713 Development Systems Kit, which also incorporates a motherboard with 8 MB of external RAM and 256 KB of flash RAM. The robots transmit their respective desired position and current position back to the computer, which is then recorded by the computer every 500 ms. Position update information, determined by the overhead camera and vision software, is broadcasted to the robots approximately every 500 ms. The flow of information from the computer vision to the robots and vice versa is illustrated in Figure 13. In addition, to improve tracking performance between position updates, each robot utilizes dead reckoning based on encoder information located on the drive motors. The use of two col-

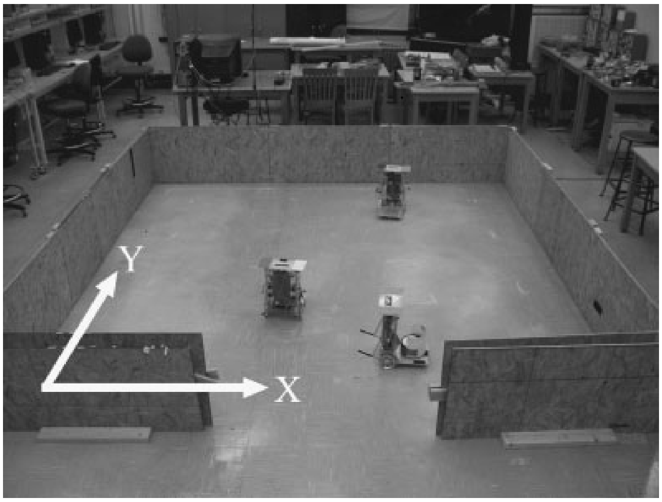


Fig. 11. The laboratory workspace.

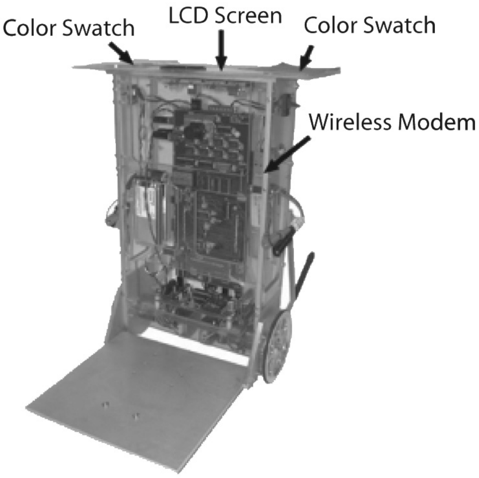


Fig. 12. SegBot.

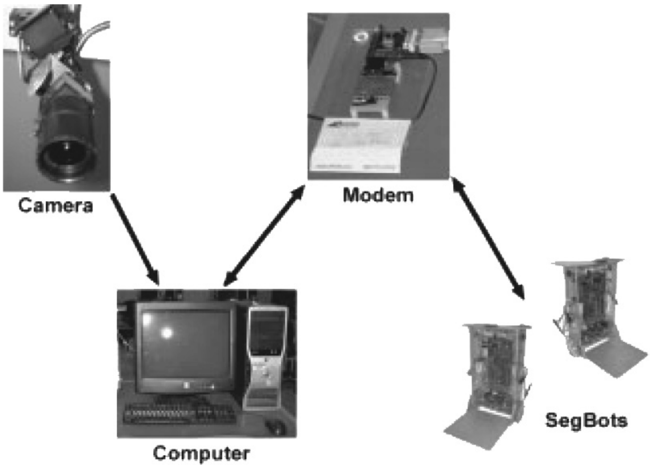


Fig. 13. Information flow chart.

ors for each robot allows the position and orientation to be calculated. Color swatches are placed on top of each SegBot and are detected by the vision software. By calculating the centroid of these colors the computer can determine position and orientation for each robot. Images captured by the camera are processed by the computer with the use of computer vision software developed for this task. This software provides a graphical user interface (GUI) allowing the user to process an image to search for individual colors, as well as transmit commands to the robots the GUI before and during processing is shown in Figures 14 and 15, respectively. Color selection can be carried out with the help of a "color picker" utility built into the program. This allows the user to determine colors captured by the camera based on hue and saturation value (HSV) values, which is used by the software to detect specific colors assigned to each SegBot.

An illustrative video of the following experiments 1–3 can be found in Extensions 1–3, respectively.

### 5.2. Experiment 1: Stationary Obstacle Avoidance

The first experiment discussed is stationary object avoidance. This involves a SegBot following a circular trajectory. A stationary object is placed in the path of the robot's trajectory. The object, in this case a cardboard box, also contains a unique color swatch pair, allowing it to be identified as an obstacle by the vision software. The position of this obstacle is then relayed to the robot, which uses this information along with its collision avoidance algorithm to determine an avoidance course. Trajectory and control values for this, and all experiments, are updated every 1 ms. The deviation of the robot from its desired trajectory, and its behavior within the avoidance boundaries is observed.

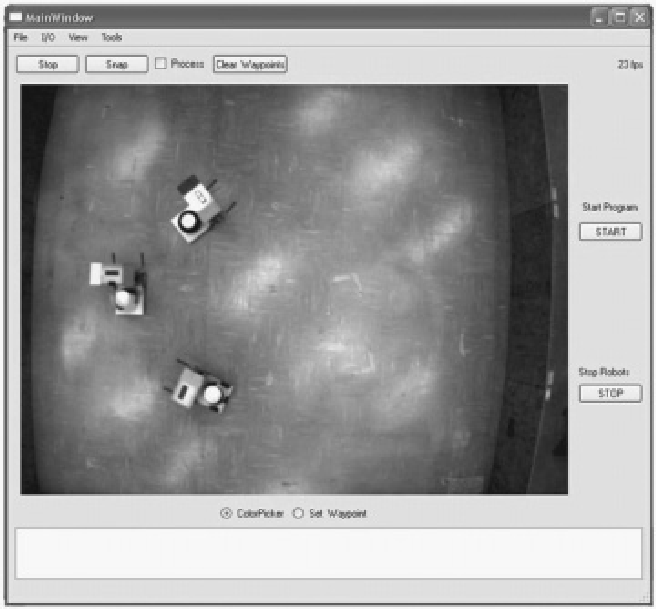
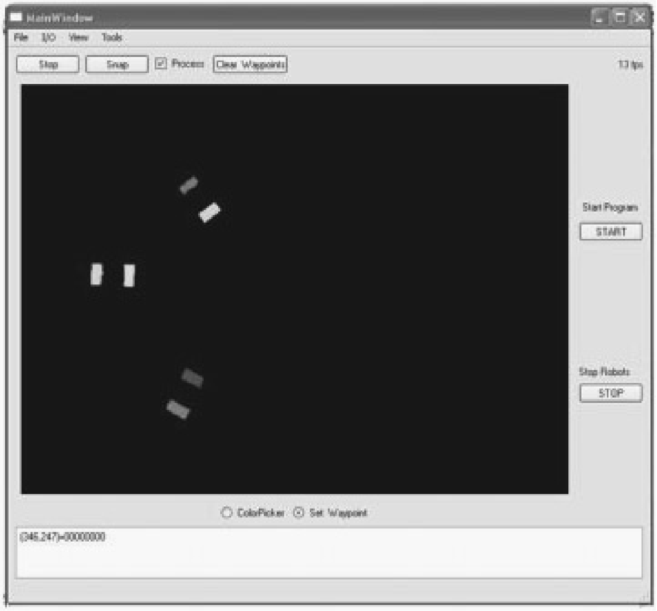


Fig. 14. GUI before processing.



(346,247)=00000000

Fig. 15. GUI while the images are being processed.

The data shown in Figures 16 and 17 illustrate the results of the stationary object avoidance test with an obstacle at position (1.1, 1.05) and detection and avoidance radius of 0.61 m and 0.21 m, respectively. The reference trajectory of the robot is defined by

$$
\begin{aligned}
x_\mathrm{d} &= 1.37 + 0.55\cos(0.3t),\\
y_\mathrm{d} &= 1.37 + 0.55\sin(0.3t).
\end{aligned}
$$

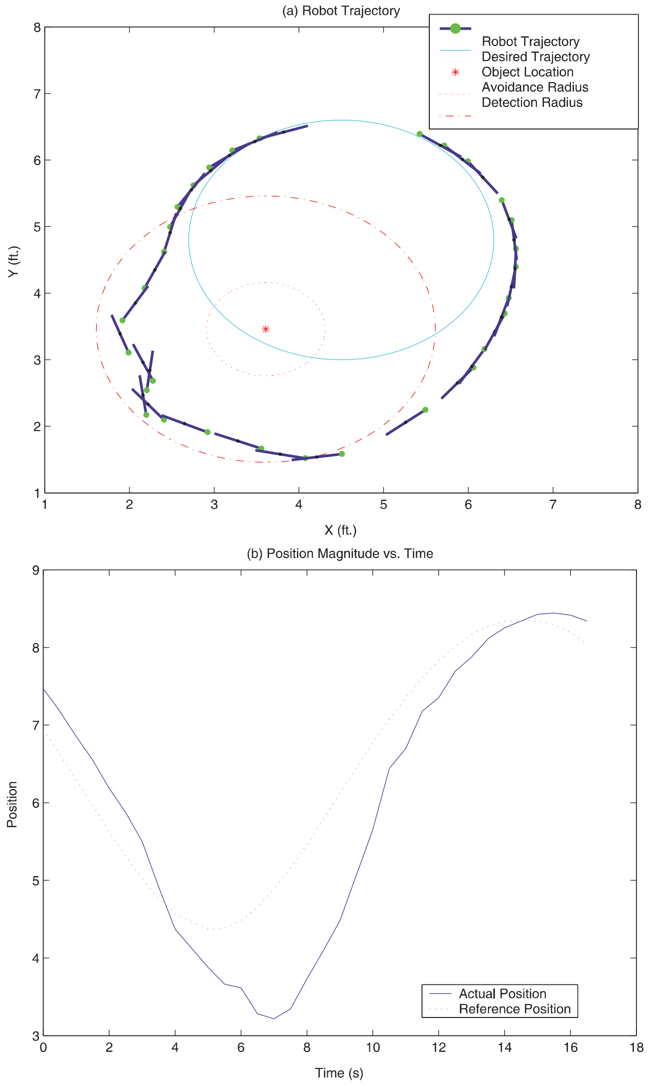


Fig. 16. Results of experiment 1 (the stationary object avoidance test): (a) robot trajectory in the $X$–$Y$ plane; (b) position magnitude versus time plot.

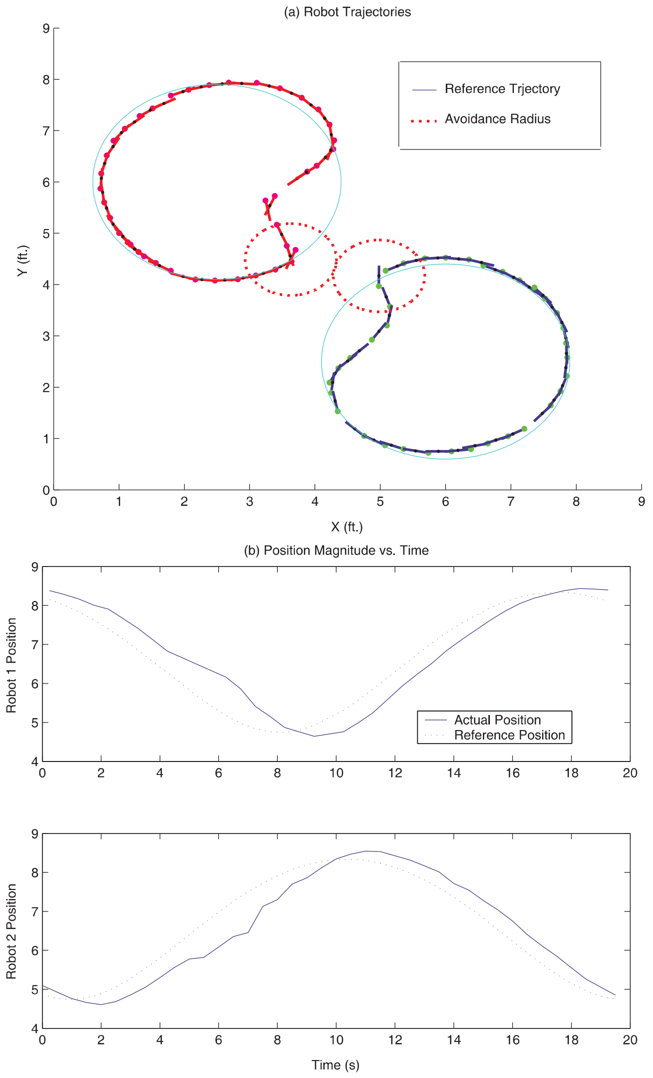


Fig. 17. Results of experiment 2 (the two-robot collision avoidance test): (a) robot trajectory in $X$–$Y$ plane; (b) position magnitude versus time plot.

The robot's position data, as shown in Figure 16(a), clearly indicate that the controller was successfully implemented. The robot tracks the reference trajectory around the circle, until it moves within the detection radius of the obstacle. The SegBot is able to navigate around the obstacle, without moving within the avoidance radius. Once the robot has moved far enough beyond the detection radius it resumes the trajectory tracking. As the trajectory has constantly progressed, the robot's desired position advances around the circle while the robot maneuvers to avoid the obstacle. When the robot has successfully navigated around the object and the avoidance radius no longer obstructs its convergence with the trajectory, the robot speeds up to regain its desired position. Figure 16(b) shows the SegBot's position as a function of time and further illustrates the robot's position compared with its trajectory. Note how the tracking error increases while the robot maneuvers to avoid the obstacle (i.e. when the obstacle enters the robot's detection region), followed by its re-convergence with the desired trajectory. When in normal trajectory tracking mode, a small constant error exists owing to the delays in the communication. Note that in our model we did not consider communication delays between the camera and the computer and between the computer and the agents. Even though such delays were not part of our model, the tracking error remains bounded, which shows the robustness of the controller with respect to such communication unreliability. Part of the future development is to include delays in the model and prove the robustness that we observed from the experiments. Extension 1 provides a video which shows the results of the stationary obstacle avoidance experiment.

### *5.3. Experiment 2: Dynamic Collision Avoidance*

Next, the behavior of two mobile robots, each operating with the collision avoidance algorithm, is tested. These robots are given circular trajectories and as the robots get closer they enter each other's detection region. The deviation from the desired trajectory during collision avoidance with a moving object and the robots convergence back to the trajectory post avoidance is observed.

For the second experiment, two SegBots denoted by Robot A and were set to follow offset circular trajectories of opposing direction. These circles were set so that the SegBots would enter each other's detection regions and trigger collision avoidance. The reference trajectories were defined as:

$$\begin{aligned} x_{\mathrm{d}} &= 6 + 0.55\cos(0.3t) \\ y_{\mathrm{d}} &= 0.79 + 0.55\sin(0.3t) \\ x_{\mathrm{d}Q} &= 0.79 + 0.55\cos(0.3t) \\ y_{\mathrm{d}Q} &= 6 + 0.55\sin(0.3t) \end{aligned}$$

The detection and avoidance radii were 0.61 and 0.21, respectively. Figure 17(a) illustrates the experimental results. As the robots approach each other and enter other's detection region they divert from their desired trajectories and successfully maneuver away from the avoidance regions. As each robot is moving away from the other during collision avoidance, they exit the opposing robot's detection region more rapidly than with a stationary object, allowing them to converge back to the desired trajectory faster. As seen in Figure 17(b), the robot's respective divergence from their desired trajectories are smaller in magnitude when compared with the corresponding graph of stationary object avoidance. Extension 2 provides an illustrative video of this experiment.

### *5.4. Experiment 3: Dynamic Formation Control*

The final experiment implements the leader–follower and formation control result. The objective is to have a group of three robots switching between leader–follower mode (line formation) and triangular formation. During this experimental setup the collision avoidance algorithms are not used. Given the close proximities between robots required to perform the formation controls, collision avoidance would have prevented the desired close spacing between robots in formation. A single robot is given a circular trajectory and is designated the *leader*. Two other robots are designated as *followers*. For the leader–follower control, the lead robot is programmed only to track a designated trajectory. The follower robots are programmed to track, at a desired distance, the leader robot in front of them. After a designated time has elapsed the reference formation command changes from line to a triangular shape through the reference trajectory. Here, the follower robots track a triangular formation. Each follower maintain a designated distance and angle behind the lead robot. In this case, each follower robot is tracking the lead robot. This formation is then maintained as the lead robot continues to track its circular trajectory. After a set amount of time, this control algorithm transitions smoothly back to the previous leader–follower control. For this experiment the leader robot, robot 1, is set to follow a constant circular trajectory defined by

$$\begin{aligned} x_{\mathrm{d}} &= 4.7 + 2.8\cos(0.3t), \\ y_{\mathrm{d}} &= 4.5 + 2.8\sin(0.3t). \end{aligned}$$

The designated follower robots, robots 2 and 3, are set to switch between leader–follower (line formation) and triangle formation controls at 40 s intervals.

Figures 18–20 show snapshots of the robots moving in a circle and changing formation from line to "V" shape. In particular, Figure 18 shows snapshots of the robots converging to a line formation and moving around a circular trajectory. First, in the line formation mode, robot 2 was set to track at a distance of 0.37 m behind the leader and robot 3 was set to follow at a distance of 0.21 m behind robot 2. Robot 2 was able to maintain a trajectory within an error of 0.26–0.73 m from its desired position, relative to robot 1. Robot 3 maintained a trajectory within an error of 0.42–1.08 m. In this experiment the desired trajectories for the followers are calculated from the leader position relayed from the computer. Those trajectories are only updated every 500 ms (the communication period between robots and the computer) for the follower robots, as opposed to every 1 ms for the leader, whose trajectory is directly specified from the computer and does not need to be calculated.

In Figure 19 a sequence of snapshots illustrates the transition from line to "V" formation. As the leader–follower control switches to "V" formation tracking, the control gains are adjusted so that the follower robots can quickly approach their desired following positions in the triangle formation. Once sufficiently close to the desired position, the gains are reset to allow a consistent formation to be maintained.

Robot 3 was set to follow at an angle of 35° inside the path of robot 1, at a distance of 0.52 m. Robot 2 was set to follow at an angle of 35° outside of the path of robot 1, also at a distance of 0.52 m. Robot 2 maintained a trajectory within an error of 0.66–1.27 m of its desired position, while robot 3 kept within an error of 0.3–0.55 m of its reference trajectory. Extension 3 is a video summarizing the dynamic formation control experiment.

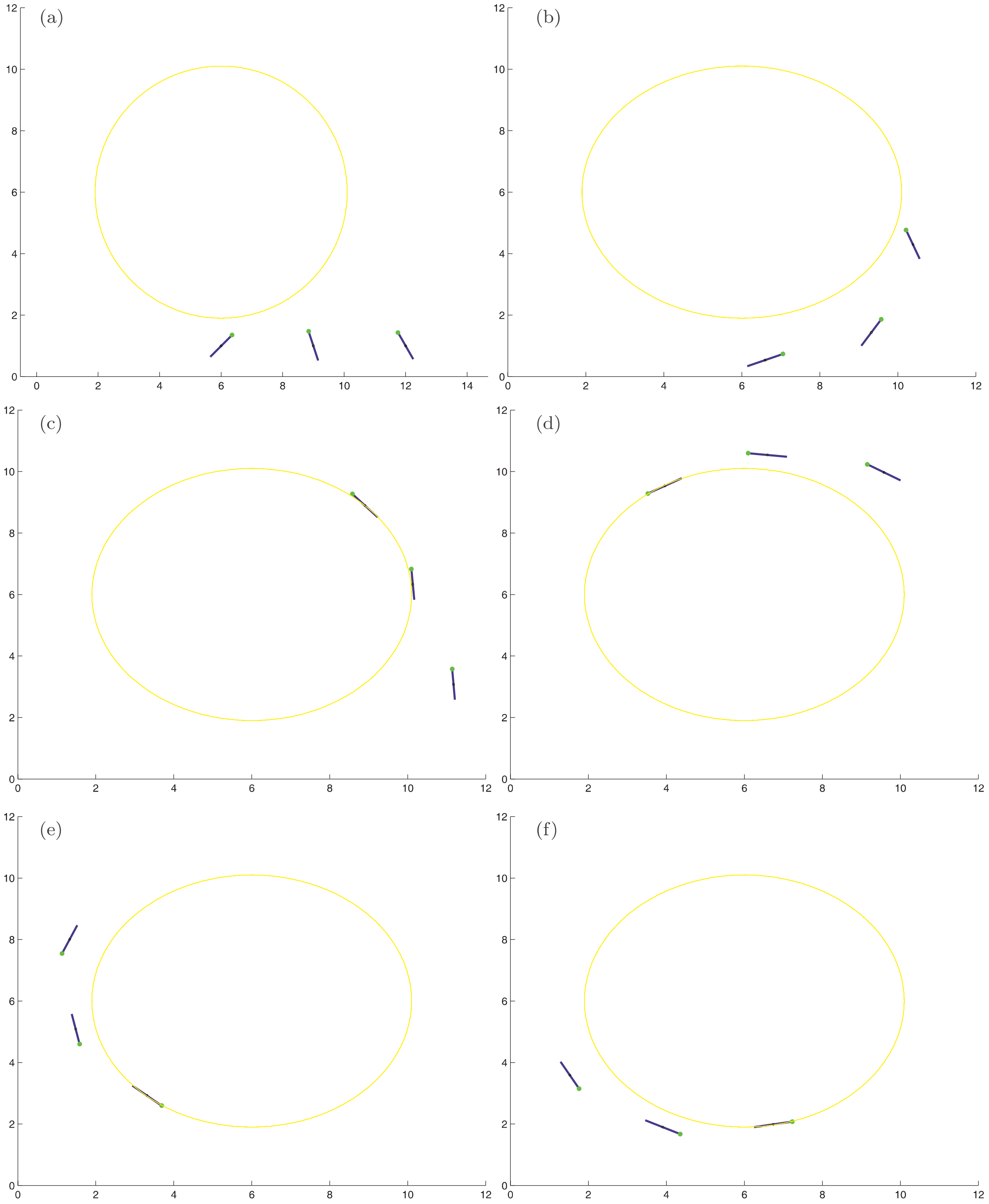


Fig. 18. A group of robots moving on a circle in a line formation. The robots begin by converging to a line formation and continue the sequence as the formation is held for an entire revolution.

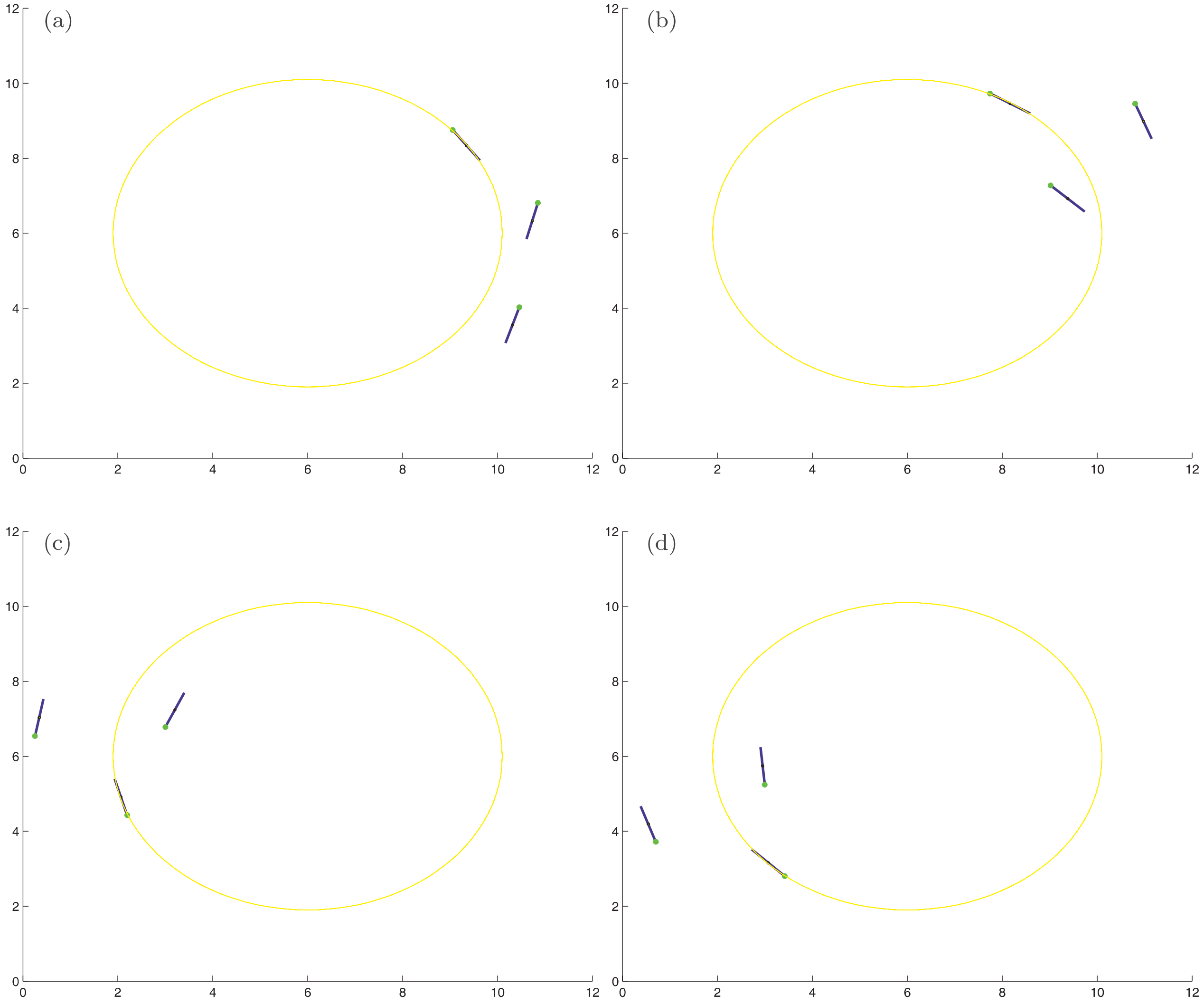


Fig. 19. A group of robots moving in a circle and changing formation from a line to a "V" shape. After some time, the robots transition to the "V" shape formation. This configuration is maintained for a period.

## 6. Conclusion

We have studied the problem of trajectory tracking and collision avoidance for non-holonomic systems. We have designed a controller that guarantees collision avoidance and tracking with bounded error outside the collision region. The design is based on Lyapunov-type approach and avoidance functions. We have used this basic framework to solve leader–follower and formation control problems for multi-agent non-holonomic systems. Finally, we have addressed the problem of velocity control of a group of robots while maintaining a certain formation using leader-tracking plus collision avoidance control laws. We have shown how most of the results that require cooperation and coordination of agents can be restated as a tracking problem. This allows us to utilize a distributed control scheme that achieves the control objectives while minimizing the communication among the agents. The experimental data confirm the effectiveness of our control design and show the robustness of the system with respect to communication unreliability. Future directions include considering a more complex dynamical model that allows extensions to a larger class of physical systems. Moreover, in order to improve the overall performance, we aim to consider a more complete model of the overall multi-agent system which includes delays in the communication channels and design controllers that mitigate the effect of the delays on the system performance.

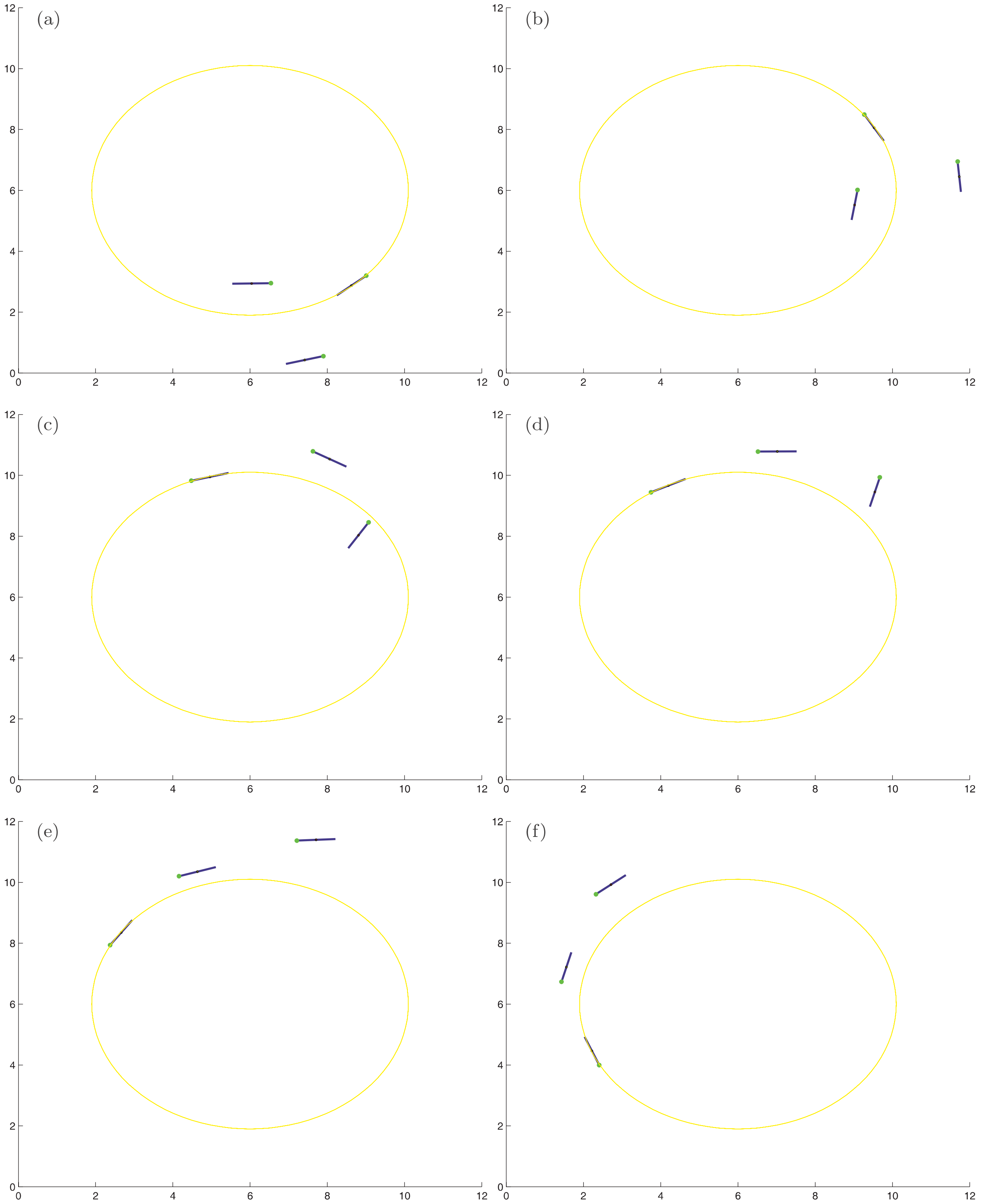


Fig. 20. A group of robots moving in a circle and changing formation from a "V" shape to a line. Finally the robots transition back to the triangle formation state, where the sequence began.

## Acknowledgement

This research was partially supported by the Office of Naval Research grant N00014-05-1-0186.

## Appendix: Index to Multimedia Extensions

The multimedia extension page is found at http://www.ijrr.org

**Table of Multimedia Extensions**

| Extension | Type | Description |
|---|---|---|
| 1 | Video | Summary of experiment 1 showing a robot tracking a circular trajectory while avoiding collision with an obstacle that is placed in its path. The obstacle is placed at different points of the path and each time the robot follows a different alternative trajectory in order to avoid collision and minimize the deviation from its original trajectory. |
| 2 | Video | Illustration of experiment 2 showing two robots moving on their path and avoiding collision with each other in a cooperative fashion. |
| 3 | Video | Illustration of the third experiment showing three robots following a circular trajectory and dynamically changing their formation from a line to a "V" shape. |

## References

Atkar, P. N., Choset, H. and Rizzi, A. A. (2002). Distributed cooperative control of multiple vehicle formations using structural potential functions. *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain, 21–26 July.

Balch, T. and Arkin, R. C. (1998). Behavior-based formation control for multirobot systems. *IEEE Transactions on Robotics and Automation*, **14**(12): 926–939.

Bloch, A. and Drakunov, A. 1995). Tracking in nonholonomic dynamic systems via sliding modes. *Proceedings of the IEEE Conference on Decision and Control*, New Orleans, LA, 13–15 December.

Canudas de Wit, C., Siciliano, B. and Bastine, G. (1996). *Theory of Robot Control*. London, Springer.

Desai, J. P., Kumar, V. and Ostrowski, J. P. (1999). Control of changes in formation for a team of mobile robots. *Proceedings of the IEEE International Conference on Robotics and Automation*, Detroit, MI, 10–15 May.

Desai, J. P., Kumar, V. and Ostrowski, J. P. (2001). Modeling and control of formations of nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation*, **17**(6): 905–908.

Dong, W., Huo, W., Tso, S. K. and Xu, W. L. (2000). Tracking control of uncertain dynamic nonholonomic system and its application to wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, **16**(6): 870–874.

Dunbar, W. B. and Murray, R. M. (2006). Distributed receding horizon control for multi-vehicle formation stabilization. *Automatica*, **42**(4): 549–558.

Egerstedt, M. and Hu, X. (2001). Formation constrained multiagent control. *IEEE Transactions on Robotics and Automation*, **17**(6): 947–951.

Justh, E. W. and Krishnaprasad, P. S. (2004). Equilibria and steering laws for planar formations. *Systems and Control Letters*, **52**: 25–38.

Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, **5**(1): 90–98.

Leitmann, G. (1980). Guaranteed avoidance strategies. *Journal of Optimization Theory and Applications*, **32**: 569–576.

Leitmann, G. and Skowronski, J. (1977). Avoidance control. *Journal of Optimization Theory and Applications*, **23**: 581–591.

Leonard, N. E. and Fiorilli, E. (2001a). Collective motion: bistability and trajectory tracking. *Proceedings of the IEEE Conference on Decision and Control*, Orlando, FL, 4–7 December.

Leonard, N. E. and Fiorilli, E. (2001b) Dec. 4-7. Virtual leaders, artificial potentials and coordinated control of groups. *Proceedings of the IEEE Conference on Decision and Control*, Orlando, FL, 4–7 December.

Micaelli, A. and Samson, C. (1993). Trajectory tracking for unicycle-type and two-steering-wheels mobile robots. *Techical Report*, INRIA.

Morgansen, K. (2001). Controllability and trajectory tracking for classes of cascade-form second order nonholonomic systems. *Proceedings of the IEEE Conference on Decision and Control*, Orlando, FL, 4–7 December.

Murray, R. M. 2007. Recent research in cooperative control of multi-vehicle systems. *Journal of Dynamic Systems, Measurement and Control* **129**(5): 571–583.

Murray, R. M., Li, Z. and Sastry, S. (1994). *A Mathematica Introduction to Robotic Manipulation*. Boca Raton, FL, CRC Press.

Ögren, P., Fiorelli, E. and Leonard, N. E. (2004). Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, **49**(8): 1292–1302.

Parker, L. E. (1993). Designing control laws for cooperative agent teams. *Proceedings IEEE International Conference on Robotics and Automation*, 2–6 May.

Sepulchre, R., Paley, D. and Leonard, N. E. (2004). Collective motion and oscillator synchronization. *Proceedings 16th Symposium on Mathematical Theorey of networks and Systems*, Leuven, Belgium, p. 308.

Sontag, E. (2006). *Input to State Stability: Basic Concepts and Results*. Berlin, Springer.

Stipanović, D. M., Hokayem, P. F., Spong, M. W. and Šiljak, D. D. (2007). Cooperative Avoidance Control for Multiagent Systems. *Journal of Dynamic Systems, Measurement and Control* **129**(5): 699–707.

Stipanović, D. M., Inalhan, G., Teo, R. and Tomlin, C. J. (2004). Decentralized overlapping control of a formation of unmanned aerial vehicles. *Automatica*, **40**: 1285–1296.

Swaroop, D. and Hedrick, J. K. 1996. String stability of interconnected systems. *IEEE Transactions on Automatic Control*, **41**(3): 349–357.

Tanner, H. G. 2004. ISS properties of nonholonomic vehicles. *Systems and Control Letters*, **53**(3–4): 229–235.

Tanner, H. G., Loizou, S. and Kyriakopoulos, K. J. (2001). Nonholonomic stabilization with collision avoidance for mobile robots. *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, HI, 29 October–3 November.

Tanner, H. G., Pappas, G. J. and Kumar, V. (2002). Input-to-state stability on formation graphs. *Proceedings of the IEEE Conference on Decision and Control*, Las Vegas, NV, 10–13 December.

Tanner, H. G., Pappas, G. J. and Kumar, V. (2004). Leader-to-formation stability. *IEEE Transactions on Robotics and Automation*, **20**(3): 433–455.

Šiljak, D. D. (1991). *Decentralized Control of Complex Systems*. Boston, MA, Academic Press.

Wu, Y., Wang, B. and Zong, G. D. (2005). Finite-time tracking controller design for nonholonomic systems with extended chained form. *IEEE Transactions on Circuits and Systems—II: Express Briefs*, **52**(11): 798–802.