

Robust Formation Control and Servoing using Switching Range Sensors

M. Karasalo and X. Hu

*Centre for Autonomous Systems
Royal Institute of Technology
100 44 Stockholm, Sweden*

Abstract

In this paper, control algorithms are presented for formation keeping and path following for non-holonomic platforms. The controls are based on feedback from onboard directional range sensors, and a switching Kalman filter is introduced for active sensing. Stability is analyzed theoretically and robustness is demonstrated in experiments and simulations.

Key words: Formation control, multi-agent systems, path following, sensor-based feedback control

1 Introduction

In recent years, the robotics community has become increasingly interested in multi-robot systems. A strong motivation for using multi-robot systems in applications such as surveillance and cleaning is that efficiency and robustness can be significantly improved. The drawback is, of course, that as more robots are added to a system the complexity of the control problem increases. This motivates the development of scalable, cascaded formation controls [8], [19]. Such controls are proposed in this paper, for agents with non-holonomic kinematics and limited sensing capabilities.

A variety of approaches to formation control have been developed, often inspired by flocking and schooling behavior observed in various animal species. The pioneering work on "boids" presented in [1] in the field of computer graphics is now well known.

Formation controls based on so called virtual vehicles and artificial potentials are proposed in [3], [6], [7], and [13]. This approach is robust and particularly advantageous if each agent in the formation should follow a predefined path. In this paper, however, we study a scenario where the majority of the agents has no a priori information on the desired formation path.

Graph theoretical approaches to formation control have become increasingly popular and are proposed in for instance [14] and [15]. In many such contributions, the focus is on formation structure and graph theoretical properties such as connectivity . In this paper, we focus on control design with constraints on sensors and actuators.

A common formation control strategy is behavior-based control. Such strategies are proposed in [2], [12], [18] and others. The idea is to merge a set of desired behaviors such as keeping a certain formation, following a specified path and avoiding obstacles. In this paper, we focus on formation keeping. The controls proposed here can be incorporated in a more complex behavior-based approach.

In various applications it is desired, mainly for cost reasons, that cheap and therefore also simple agents are used, when possible, to assist a few highly skilled but also expensive agents. In particular, in high risk operations such as mine sweeping or exploration of hostile environments, cheap and expendable agents are preferable.

In this paper, we consider teams of robots consisting of one leader and a number of followers. All robots are equipped with directional range sensors, but only the leader has advanced navigation skills and sufficient computational power to execute a more advanced strategy. We propose three cascaded tracking controls for the followers, which all assume input from directional range sensors. Complex formations can easily be achieved by altering control parameters. Results from initial experiments and simulations are included at the end of the paper to illustrate the theoretical results.

2 Related Work and Contributions

For mobile robots, navigation and target tracking are basic capabilities and well studied in the literature. Tracking by means of a virtual vehicle was proposed in [10] and implemented in for instance [13]. This is a robust approach but a drawback is that some kind of planned path is needed to design the dynamics for the virtual vehicle.

The controls developed in this paper assume platforms with non-holonomic kinematic constraints and input from directional sensors. On the issue of sensor and actuator limitations, the literature in the field of mobile robotics is somewhat sparse. In [11] and [17] omnidirectional visual sensors are used for formation control. A tracking control for directional sensors is presented in [13]. For AUVs recent contributions include [20], [21] and the references therein.

The main contributions of the paper are three cascaded formation controls for follower agents. The control input is obtained from low-resolution directional range sensors using active sensing by means of a switching Kalman filter [16]. No inter-agent communication is needed to execute the controls.

A similar formation control problem is studied in [5]. While this paper presents a behavior-based approach with two separate controls for heading and bearing, our approach uses only one control that achieves both objectives simultaneously. Some of the results in this paper have been previously provided in [4] and are briefly reviewed in [19]. This paper extends the rigid formation control introduced in [4] to scenarios with dynamic control parameter values. Further, it is shown that a slight modification of the control yields monotonic convergence for some of the formation errors.

3 Outline

The paper is organized as follows: in Section 4, notation is introduced and assumptions on the platform and range sensor model are reviewed. Section 5 is dedicated to the cascaded formation control in its original design. A proof of global stability is included. Section 6 discusses refinements of the control. Two modified versions of the first control are provided, one that drives the heading of the agents monotonically toward their reference values and one that introduces dynamic formation parameters. The controls presented in this paper assume that input is provided from directional sensors, which calls for active sensing as the tracked target agent may only be visible from a subset of the range sensors at any given time. A switching Kalman filter for active sensing is presented in Section 7. The results are demonstrated in experiments and simulations with the Khepera II platform in Section 8, and a concluding discussion is included in Section 9.

4 Kinematic Constraints and Notation

In this section, we discuss the assumptions on platform kinematics and sensors, and introduce notation. Consider a group of n agents $R_i, i = 1, \dots, n$. The control objective is for R_i to track R_{i-1} at the distance $d_{0,i}$ and angle $\beta_{0,i}$. The notation is illustrated in Figure 1. We assume that the kinematics of each

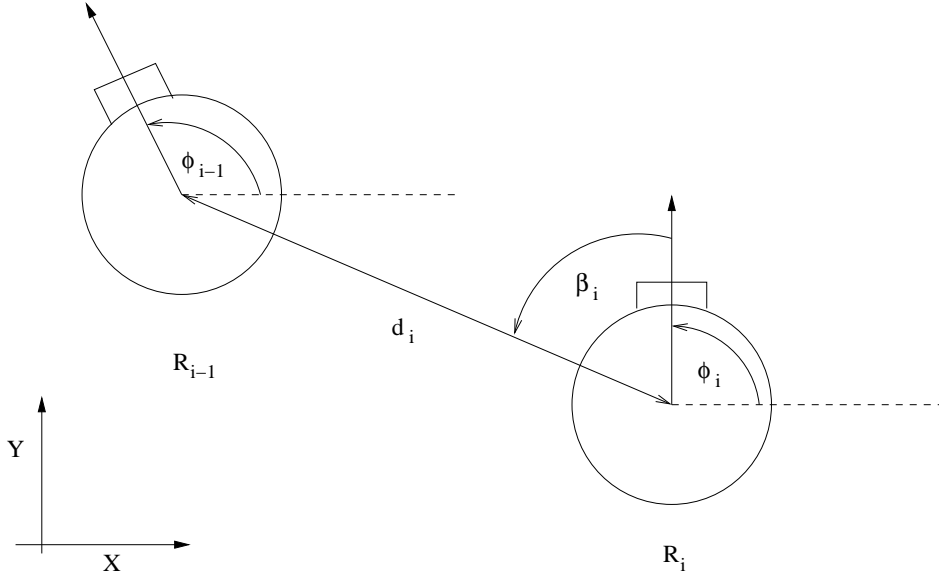


Fig. 1. Notation of angles and distances.

robot can be modeled by the set of ODEs

$$\begin{aligned}\dot{x}_i &= v_i \cos \phi_i \\ \dot{y}_i &= v_i \sin \phi_i \\ \dot{\phi}_i &= \omega_i,\end{aligned}\tag{1}$$

where (x_i, y_i) is the position and ϕ_i is the heading of R_i . v_i and ω_i are control inputs. The kinematics (1) is common for mobile platforms. Examples include the PowerBot from ActivMedia, the Magellan Pro platform from iRobot and the Khepera II from K-Team, which is used in the experiments and simulations in Section 8. Further, we assume the agents have only directional range sensors, and keep track of their own position by means of odometry. The Khepera II has directional IR proximity sensors, positioned as shown in Figure 2.

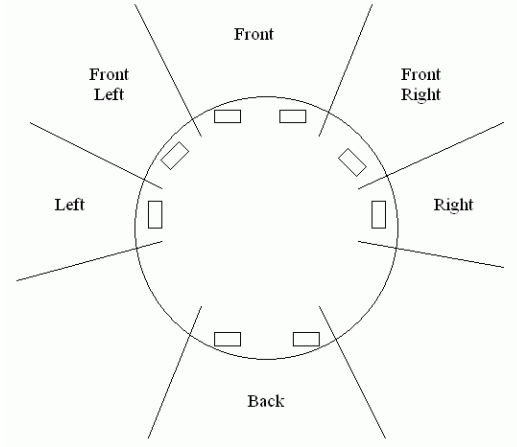


Fig. 2. Left: The Khepera II. Right: Positions of IR sensors.

5 Basic Formation control

In this section we derive the formation control that constitutes the foundation of controls derived in later sections. The tracking control is an easily implemented cascaded control algorithm that makes a robot track a leader or a target at a specified distance and bearing angle. Let d_i denote the actual distance between R_i and R_{i-1} and let β_i be the actual relative angle from the orientation of R_i to R_{i-1} (see Figure 1). The reference values of the formation parameters are denoted $d_{0,i}$ and $\beta_{0,i}$. Finally define $\gamma_i = \phi_i - \phi_{i-1}$. Now we can express the kinematics of the system (1) as

$$\dot{d}_i = -v_i \cos \beta_i + v_{i-1} \cos(\gamma_i + \beta_i) \quad (2)$$

$$\dot{\beta}_i = -\omega_i + \frac{v_i}{d_i} \sin \beta_i - \frac{v_{i-1}}{d_i} \sin(\gamma_i + \beta_i) \quad (3)$$

$$\dot{\gamma}_i = \omega_i - \omega_{i-1}. \quad (4)$$

Equation (4) is trivial and Equation (2) is easily derived from the observation that d_i decreases as the projection of the velocity of R_i on the distance vector $(x_{i-1} - x_i, y_{i-1} - y_i)^T$ increases, and increases with the corresponding projection of the velocity of R_{i-1} . To understand Equation (3), note that the second term, $\frac{v_i}{d_i} \sin \beta_i$, is approximately the change of β_i if R_i moves and R_{i-1} stands still, and conversely for the third term.

This is a cascaded system due to the presence of v_{i-1} and ω_{i-1} . The control objective can now be stated as:

Given $v_1(t)$ and $\omega_1(t)$, find controls $v_i(t)$ and $\omega_i(t)$ such that for $i = 2, \dots, n$, as $t \rightarrow \infty$,

$$d_i \rightarrow d_{0,i} \tag{5}$$

$$\beta_i \rightarrow \beta_{0,i} \tag{6}$$

$$\gamma_i \rightarrow 0. \tag{7}$$

When $\beta_{0,i} = 0$, $i = 2, \dots, n$, the robots should follow straight behind each other. We refer to this as a serial formation. If instead $\beta_{0,i} = \pi/2$, $i = 2, \dots, n$, the robots should move on a line parallel to each other. For reasons soon to be disclosed, we treat the case where $|\beta_{0,i}| < \pi/2$. Thus, the achievable formations range from serial to “sloping parallel”, which is often seen, for example, in formations of aircraft and convoys of tanks and battleships. The sloping parallel formation allows for an overlap in ground coverage as seen in Figure 3, which is desirable for applications such as mine sweeping or when constructing a detailed map. Derivation of the controls is facilitated

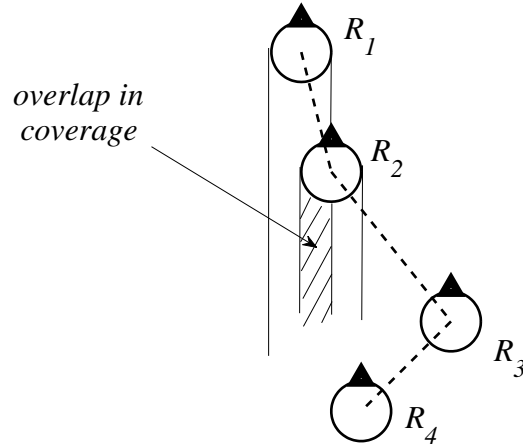


Fig. 3. An example formation. Overlap is illustrated between R_1 and R_2 .

by defining a reference point $(x_{0,i}, y_{0,i})$, chosen on R_i 's axis of orientation at a distance $d_{0,i}$ from the center of R_i :

$$\begin{aligned}
x_{0,i} &= x_i + d_{0,i} \cos(\phi_i + \beta_{0,i}) \\
y_{0,i} &= y_i + d_{0,i} \sin(\phi_i + \beta_{0,i}).
\end{aligned} \tag{8}$$

With this particular choice of reference point we obtain a point that can be driven arbitrarily close to R_{i-1} without causing the angle to target to be undefined. The first formation control is presented in the following theorem:

Theorem 1 *As $t \rightarrow \infty$, and with $\beta_{0,i} \neq \pi/2 + m\pi$ for any integer m , $(x_{0,i}(t), y_{0,i}(t))$ converges to $(x_{i-1}(t), y_{i-1}(t))$ with the following control:*

$$\textbf{Control 1 : } \begin{cases} v_i = \frac{k}{\cos \beta_{0,i}} \left(d_i \cos(\beta_i - \beta_{0,i}) - d_{0,i} + \frac{v_{i-1}}{k} \cos(\gamma_i + \beta_{0,i}) \right) \\ \omega_i = \frac{k}{d_{0,i} \cos \beta_{0,i}} \left(d_i \sin \beta_i - d_{0,i} \sin \beta_{0,i} - \frac{v_{i-1}}{k} \sin \gamma_i \right). \end{cases} \tag{9}$$

Proof The proof is straightforward. In fact, let $x_e = x_{0,i} - x_{i-1}$, $y_e = y_{0,i} - y_{i-1}$, then we have

$$\begin{aligned}
x_e &= x_i + d_{0,i} \cos(\phi_i + \beta_{0,i}) - x_{i-1} \\
y_e &= y_i + d_{0,i} \sin(\phi_i + \beta_{0,i}) - y_{i-1},
\end{aligned} \tag{10}$$

so that

$$\begin{aligned}
\begin{pmatrix} \dot{x}_e \\ \dot{y}_e \end{pmatrix} &= \begin{pmatrix} v_i \cos \phi_i - d_{0,i} \omega_i \sin(\phi_i + \beta_{0,i}) - v_{i-1} \cos \phi_{i-1} \\ v_i \sin \phi_i + d_{0,i} \omega_i \cos(\phi_i + \beta_{0,i}) - v_{i-1} \sin \phi_{i-1} \end{pmatrix} \\
&= \begin{pmatrix} \cos \phi_i & -d_{0,i} \sin(\phi_i + \beta_{0,i}) \\ \sin \phi_i & d_{0,i} \cos(\phi_i + \beta_{0,i}) \end{pmatrix} \begin{pmatrix} v_i \\ \omega_i \end{pmatrix} - v_{i-1} \begin{pmatrix} \cos \phi_{i-1} \\ \sin \phi_{i-1} \end{pmatrix}. \tag{11}
\end{aligned}$$

Now, by solving

$$\begin{pmatrix} \cos \phi_i & -d_{0,i} \sin(\phi_i + \beta_{0,i}) \\ \sin \phi_i & d_{0,i} \cos(\phi_i + \beta_{0,i}) \end{pmatrix} \begin{pmatrix} v_i \\ \omega_i \end{pmatrix} = -k \begin{pmatrix} x_e \\ y_e \end{pmatrix} + v_{i-1} \begin{pmatrix} \cos \phi_{i-1} \\ \sin \phi_{i-1} \end{pmatrix} \tag{12}$$

for v_i and ω_i we obtain Control 1, which obviously yields the error dynamics

$$\begin{aligned}\dot{x}_e &= -kx_e \\ \dot{y}_e &= -ky_e.\end{aligned}\tag{13}$$

■

As seen from Equation (13), the control gain k determines the rate of convergence to the desired configuration. In practice, of course, physical limitations will determine the upper bound for k . Also, due to the discretization of the control in implementation, a too large gain may lead to overshoot. The value of k must therefore be tuned with respect to platform and discretization method. By construction, Control 1 drives d_i to $d_{0,i}$ and β_i to $\beta_{0,i}$, satisfying the control objectives (5) and (6). For the third control objective, consider the case $\omega_{i-1} = 0$. With Control 1, the formation is driven to the steady state $d_i = d_{0,i}$, $\beta_i = \beta_{0,i}$. Then (4) becomes

$$\dot{\gamma}_i = -\frac{v_{i-1}}{d_{0,i} \cos \beta_{0,i}} \sin \gamma_i, \tag{14}$$

with equilibria at $\gamma_i = 0 \pm \pi$. For $v_{i-1} > 0$ and $|\beta_{0,i}| < \pi/2$ we get

$$\dot{\gamma}_i < 0 \text{ if } \gamma_i \in (0, \pi) \tag{15}$$

$$\dot{\gamma}_i > 0 \text{ if } \gamma_i \in (-\pi, 0). \tag{16}$$

In other words, with $\omega_{i-1} = 0$, $\dot{\gamma}_i$ has a stable equilibrium at $\gamma_i = 0$ and unstable equilibria at $\gamma_i = \pm\pi$. Therefore, for bounded angular velocity ω_{i-1} , γ_i stays close to zero.

By varying the tracking distance and bearing angle a variety of differently shaped formations can be obtained, such as the one shown in Figure 3. An example of a complex formation is given in [4]. Clearly, for any given rate of convergence, the control actions of Control 1 will grow large as $\beta_{0,i}$ approaches $\pi/2$ and the control becomes singular for $\beta_{0,i} = \pi/2$. To the extent of our knowledge, no known controller can handle the whole range from 0 to $\pi/2$. Thus for reference angles close to $\pi/2$ one should switch to another controller to avoid singularity [8].

6 Further Refinement of the Formation Control

In this section refinements of Control 1 are proposed and analyzed. The issues of keeping agent R_{i-1} in the field of view and allowing for dynamic control parameters are discussed.

6.1 Driving $\beta_i(t)$ to $\beta_{0,i}$

Equation (13) shows that Control 1 is globally exponentially stable in terms of position errors. However, the control cannot guarantee that the angular error $|\beta_i(t) - \beta_{0,i}|$ decreases monotonically. Thus, if the sensors have only limited angular field of view, we may risk losing track of the target. Therefore, some strategy for active sensing is needed when using Control 1. Such a strategy is further discussed in Section 7, where a switching Kalman filter is used as an observer whose input depends on which sensors of agent R_i currently view agent R_{i-1} .

However, a slight modification of Control 1 does make $|\beta_i(t) - \beta_{0,i}|$ decrease monotonically. The ability to control β_i in this manner means that agent R_i can choose to keep agent R_{i-1} in the field of view of some particular sensors at all times, unless the reference value $\beta_{0,i}$ changes rapidly. The price we pay is the loss of globality. The modified control is presented in the following theorem.

Theorem 2 *Let*

$$\text{Control 2 : } \begin{cases} v_i = \frac{k}{\cos \beta_i} \left(d_i - d_{0,i} \cos(\beta_i - \beta_{0,i}) + \frac{v_{i-1}}{k} \cos(\gamma_i + \beta_i) \right) \\ \omega_i = \frac{k}{d_i \cos \beta_i} \left(d_i \sin \beta_i - d_{0,i} \sin \beta_{0,i} - \frac{v_{i-1}}{k} \sin \gamma_i \right), \end{cases} \quad (17)$$

where $k > 0$, $d_{0,i} > 0$, $|\beta_{0,i}| < \frac{\pi}{2}$. Then for any $\beta_i(0) \in (-\frac{\pi}{2}, \frac{\pi}{2})$ and $d_i(0) > 0$, $|\beta_i(t) - \beta_{0,i}|$ tends to zero monotonically and $d_i(t) - d_{0,i}$ tends to zero.

Proof Define $\hat{d}_i = d_i \cos \beta_i - d_{0,i} \cos \beta_{0,i}$ and $\hat{\beta}_i = \beta_i - \beta_{0,i}$. Then we have, from Equations (2) - (3)

$$\begin{aligned} \dot{\hat{d}}_i &= \dot{d}_i \cos \beta_i - d_i \dot{\beta}_i \sin \beta_i = -v_i + v_{i-1} \cos \gamma_i + d_i \omega_i \sin \beta_i \\ \dot{\hat{\beta}}_i &= \dot{\beta}_i = -\omega_i + \frac{v_i}{d_i} \sin \beta_i - \frac{v_{i-1}}{d_i} \sin(\gamma_i + \beta_i). \end{aligned} \quad (18)$$

Now, plugging in Control 2 into (18) yields

$$\begin{aligned} \dot{\hat{d}}_i &= -k \hat{d}_i \\ \dot{\hat{\beta}}_i &= -\frac{k d_{0,i} \cos \beta_i}{d_{0,i} \cos \beta_{0,i} + \hat{d}_i} \sin \hat{\beta}_i. \end{aligned} \quad (19)$$

Since $|\hat{d}_i(t)|$ decreases monotonically, $d_i(t) > 0 \forall t \geq 0$ due to the continuity of the solution. Since $\dot{\hat{\beta}}_i = -k \frac{d_{0,i}}{d_i(t)} \sin \hat{\beta}_i$, it is easy to see that as long as $|\beta_i(0)| < \frac{\pi}{2}$, $|\beta_i(t) - \beta_{0,i}|$ decreases monotonically.

■

When $\beta_{0,i}$ is constant or slowly varying, Control 2 ensures that R_{i-1} stays within the field of view of some of the sensors of R_i . However, some scenarios may require a flexible formation, where $\beta_{0,i}$ changes rapidly to adapt to conditions in the environment. This would motivate the introduction of some active sensing scheme. In the next section, a formation control with dynamic control parameters is proposed.

6.2 Dynamic Control Parameters

A variety of formations can be obtained by altering the parameters $d_{0,i}$ and $\beta_{0,i}$. As Control 1 is globally stable for $|\beta_{0,i}| < \pi/2$, switching between different parameter values can be performed independently on-line for each robot. Under the assumption of directional sensors, a control with switching reference values should be combined with active sensing. This is further discussed in Section 7, where a switching Kalman filter is proposed for target tracking applications.

In this section the stability of Control 1 with dynamic parameters $d_{0,i}(t), \beta_{0,i}(t)$ is analyzed. According to Theorem 1, Control 1 is globally stable for constant $d_{0,i}$ and $\beta_{0,i}$. For time varying $d_{0,i}(t)$ and $\beta_{0,i}(t)$, using the same control input v_i and ω_i yields an additional nonlinear term in the error dynamics due to the nonzero time derivatives of $d_{0,i}(t)$ and $\beta_{0,i}(t)$. Recall that $x_e = x_{0,i} - x_{i-1}, y_e = y_{0,i} - y_{i-1}$. The derivatives are now

$$\begin{pmatrix} \dot{x}_e \\ \dot{y}_e \end{pmatrix} = -k \begin{pmatrix} x_e \\ y_e \end{pmatrix} + \begin{pmatrix} \cos(\phi_i + \beta_{0,i}) & -d_{0,i} \sin(\phi_i + \beta_{0,i}) \\ \sin(\phi_i + \beta_{0,i}) & d_{0,i} \cos(\phi_i + \beta_{0,i}) \end{pmatrix} \begin{pmatrix} \dot{d}_{0,i} \\ \dot{\beta}_{0,i} \end{pmatrix}. \quad (20)$$

We see that rapid changes in $d_{0,i}(t)$ and $\beta_{0,i}(t)$ might affect stability of the error dynamics. Denote by $\Phi(t)$ the nonlinear term of Equation (20). Then the error expression is

$$\begin{pmatrix} x_e \\ y_e \end{pmatrix} = e^{-kt} \begin{pmatrix} x_e(0) \\ y_e(0) \end{pmatrix} + \int_0^t e^{-k(t-s)} \Phi(s) ds. \quad (21)$$

Note that, with $\|\cdot\|$ denoting the euclidean norm,

$$\|\Phi(t)\| = \left\| \begin{pmatrix} \cos(\phi_i + \beta_{0,i}) & -\sin(\phi_i + \beta_{0,i}) \\ \sin(\phi_i + \beta_{0,i}) & \cos(\phi_i + \beta_{0,i}) \end{pmatrix} \begin{pmatrix} \dot{d}_{0,i} \\ d_{0,i}\dot{\beta}_{0,i} \end{pmatrix} \right\| = \sqrt{\dot{d}_{0,i}^2 + d_{0,i}^2\dot{\beta}_{0,i}^2}. \quad (22)$$

Therefore, if there exists an M such that $\sqrt{\dot{d}_{0,i}^2 + d_{0,i}^2\dot{\beta}_{0,i}^2} \leq M, \forall t \geq 0$, it holds that

$$\begin{pmatrix} x_e \\ y_e \end{pmatrix} \leq \begin{pmatrix} x_e(0) \\ y_e(0) \end{pmatrix} e^{-kt} + \frac{M}{k} \begin{pmatrix} 1 \\ 1 \end{pmatrix} (1 - e^{-kt}), \quad (23)$$

so that as $t \rightarrow \infty$ the right hand side of Equation (23) tends to M/k . Thus, with Control 1 and a given value of k , the error is bounded if $\dot{d}_{0,i}(t)$, $d_{0,i}(t)$ and $\dot{\beta}_{0,i}(t)$ are chosen so that M/k is smaller than the sensor range. To eliminate the static error, the following modified control should be used:

Theorem 3 For v_i and ω_i defined by Equation (9), the control

$$\textbf{Control 3 : } \begin{cases} \tilde{v}_i = v_i + \Delta v \\ \tilde{\omega}_i = \omega_i + \Delta \omega \end{cases} \quad (24)$$

with

$$\begin{pmatrix} \Delta v \\ \Delta \omega \end{pmatrix} = -\frac{1}{d_{0,i} \cos \beta_{0,i}} \begin{pmatrix} \dot{d}_{0,i} d_{0,i} \\ \dot{d}_{0,i} \sin \beta_{0,i} + \dot{\beta}_{0,i} d_{0,i} \cos \beta_{0,i} \end{pmatrix} \quad (25)$$

results in the error dynamics (13).

Proof The proof is straightforward and follows from the fact that, with dynamic control parameters $d_{0,i}(t)$ and $\beta_{0,i}(t)$, the error dynamics are

$$\begin{aligned} \dot{x}_e &= v_i \cos \phi_i - v_{i-1} \cos \phi_{i-1} + \dot{d}_{0,i} \cos(\phi_i + \beta_{0,i}) - d_{0,i}(\dot{\beta}_{0,i} + \omega_i) \sin(\phi_i + \beta_{0,i}) \\ \dot{y}_e &= v_i \sin \phi_i - v_{i-1} \sin \phi_{i-1} + \dot{d}_{0,i} \sin(\phi_i + \beta_{0,i}) + d_{0,i}(\dot{\beta}_{0,i} + \omega_i) \cos(\phi_i + \beta_{0,i}). \end{aligned} \quad (26)$$

Replacing (v_i, ω_i) with $(v_i + \Delta v, \omega_i + \Delta \omega)$ yields

$$\begin{aligned} \dot{x}_e &= (v_i + \Delta v) \cos \phi_i - v_{i-1} \cos \phi_{i-1} + \dot{d}_{0,i} \cos(\phi_i + \beta_{0,i}) - d_{0,i}(\dot{\beta}_{0,i} + \omega_i + \Delta \omega) \sin(\phi_i + \beta_{0,i}) \\ \dot{y}_e &= (v_i + \Delta v) \sin \phi_i - v_{i-1} \sin \phi_{i-1} + \dot{d}_{0,i} \sin(\phi_i + \beta_{0,i}) + d_{0,i}(\dot{\beta}_{0,i} + \omega_i + \Delta \omega) \cos(\phi_i + \beta_{0,i}). \end{aligned} \quad (27)$$

Now define

$$\begin{pmatrix} \Delta \dot{x}_e \\ \Delta \dot{y}_e \end{pmatrix} = \begin{pmatrix} \Delta v \cos \phi + \dot{d}_{0,i} \cos(\phi_i + \beta_{0,i}) - d_{0,i}(\dot{\beta}_{0,i} + \Delta \omega) \sin(\phi_i + \beta_{0,i}) \\ \Delta v \sin \phi + \dot{d}_{0,i} \sin(\phi_i + \beta_{0,i}) + d_{0,i}(\dot{\beta}_{0,i} + \Delta \omega) \cos(\phi_i + \beta_{0,i}) \end{pmatrix} = \\ \begin{pmatrix} \cos \phi_i & -d_{0,i} \sin(\phi_i + \beta_{0,i}) \\ \sin \phi_i & d_{0,i} \cos(\phi_i + \beta_{0,i}) \end{pmatrix} \begin{pmatrix} \Delta v \\ \Delta \omega \end{pmatrix} + \begin{pmatrix} \dot{d}_{0,i} \cos(\phi_i + \beta_{0,i}) - d_{0,i} \dot{\beta}_{0,i} \sin(\phi_i + \beta_{0,i}) \\ \dot{d}_{0,i} \sin(\phi_i + \beta_{0,i}) + d_{0,i} \dot{\beta}_{0,i} \cos(\phi_i + \beta_{0,i}) \end{pmatrix}, \quad (28)$$

so that

$$\begin{pmatrix} \dot{x}_e \\ \dot{y}_e \end{pmatrix} = -k \begin{pmatrix} x_e \\ y_e \end{pmatrix} + \begin{pmatrix} \Delta \dot{x}_e \\ \Delta \dot{y}_e \end{pmatrix}. \quad (29)$$

Now, since the control $\Delta v, \Delta \omega$ defined by (25) solves

$$\begin{pmatrix} \cos \phi_i & -d_{0,i} \sin(\phi_i + \beta_{0,i}) \\ \sin \phi_i & d_{0,i} \cos(\phi_i + \beta_{0,i}) \end{pmatrix} \begin{pmatrix} \Delta v \\ \Delta \omega \end{pmatrix} = - \begin{pmatrix} \dot{d}_{0,i} \cos(\phi_i + \beta_{0,i}) - d_{0,i} \dot{\beta}_{0,i} \sin(\phi_i + \beta_{0,i}) \\ \dot{d}_{0,i} \sin(\phi_i + \beta_{0,i}) + d_{0,i} \dot{\beta}_{0,i} \cos(\phi_i + \beta_{0,i}) \end{pmatrix}, \quad (30)$$

the terms $\Delta \dot{x}_e, \Delta \dot{y}_e$ vanish and the resulting error dynamics are exponentially stable. Note that the derivatives of $d_{0,i}(t)$ and $\beta_{0,i}(t)$ are completely known since they are control inputs.

■

Next, the switching Kalman filter is derived.

7 Active Sensing Through Switching Kalman Filters

To use Control 1 for agent R_i , we need to estimate the distance d_i , the angle β_i , and the velocity v_{i-1} of R_{i-1} from sensor measurements. When using directional range sensors, the only information available is the distance from each sensor to R_{i-1} . d_i and β_i may be computed directly from sensor data, but this is not the case for v_{i-1} . In [19] a nonlinear observer is proposed for this purpose. In this section, an alternative approach is introduced, enabling the use of a linear observer in the form of a Kalman filter.

As we shall see, even though this is not required explicitly by the formation controls, the Kalman filter needs an estimate of the global position of the target in order to estimate v_{i-1} . Although this introduces extra computations, the method is appealing due to the desirable properties of the Kalman filter, such as linearity, optimality and convergence.

As Control 1 does not guarantee monotonic convergence of β_i , the observer must be robust to the scenario that agent R_{i-1} moves out of the field of view for some of the sensors of agent R_i . Using dynamic parameters, Control 2 may also loose tracking for some directional sensors. We propose a switching Kalman filter for active sensing, and introduce the concept of *active sensor pairs* to obtain appropriate input to the Kalman filter.

The Khepera II, with its directional IR proximity sensors, is used as an example platform to illustrate the method. For localization, Khepera II uses odometry. Further down, expressions for d_i , β_i , γ_i and v_{i-1} are computed from sensor measurements. First, a brief review of notation and recursion formulas for the Kalman filter is provided.

7.1 The Discrete Kalman Filter

In this section we introduce notation and review the Kalman recursions. Consider a discrete, time-invariant system with a state space vector \mathbf{x}_t governed by the linear stochastic difference equation

$$\mathbf{x}_t = A\mathbf{x}_{t-1} + B\mathbf{w}_t \quad (31)$$

and a measurement $\mathbf{y}_t \in \mathbb{R}^m$ governed by

$$\mathbf{y}_t = C\mathbf{x}_t + \mathbf{v}_t. \quad (32)$$

The signals \mathbf{w}_t and \mathbf{v}_t are random process noise and measurement noise, respectively, and are assumed to be independent, gaussian noise with covariance matrices

$$\begin{aligned} E(\mathbf{w}_t \mathbf{w}_t^T) &= Q_t \\ E(\mathbf{v}_t \mathbf{v}_t^T) &= S_t. \end{aligned} \quad (33)$$

Here, $E(\cdot)$ is the expectation value of (\cdot) . The discrete Kalman filter is an observer that gives an estimate $\hat{\mathbf{x}}_t$ of the state \mathbf{x}_t at time step t , given the estimate $\hat{\mathbf{x}}_{t-1}$ and the observation \mathbf{y}_t , in a least squares sense. In other words, if we define the estimation error as

$$\mathbf{e}_t = \mathbf{x}_t - \hat{\mathbf{x}}_t, \quad (34)$$

then the Kalman filter will give the estimate $\hat{\mathbf{x}}_t$ that minimizes $E(\mathbf{e}_t^T \mathbf{e}_t)$. Now let

$$P_t \triangleq E(\mathbf{e}_t \mathbf{e}_t^T) \quad (35)$$

denote the covariance matrix of $\hat{\mathbf{x}}_t$. Then, given initial estimates $\hat{\mathbf{x}}_0$ and P_0 , the discrete Kalman filter is the recursive process

$$\begin{aligned}
\hat{\mathbf{x}}_t^- &= A\hat{\mathbf{x}}_{t-1} \\
P_t^- &= AP_{t-1}A^T + B^T Q_{t-1}B \\
K_t &= P_t^- C^T (CP_t^- C^T + R_t)^{-1} \\
\hat{\mathbf{x}}_t &= \hat{\mathbf{x}}_t^- + K_t (\mathbf{y}_t - C\hat{\mathbf{x}}_t^-) \\
P_t &= (I - K_t C) P_t^-.
\end{aligned} \tag{36}$$

Next, the application of the Kalman filter to target tracking is discussed.

7.2 Target tracking using the Kalman filter

Consider the problem of estimating the motion of the target R_{i-1} . This should be done locally by R_i using odometry and the information available from the sensor pairs p_j , $j = 1, \dots, 6$, shown in Figure 4. A linear model is needed for the kinematics of R_{i-1} in order to apply a Kalman filter. Therefore, let the position (x_{i-1}, y_{i-1}) and motion $(\dot{x}_{i-1}, \dot{y}_{i-1})$ of R_{i-1} constitute the state vector:

$$\mathbf{x} = \begin{pmatrix} x_{i-1} & y_{i-1} & \dot{x}_{i-1} & \dot{y}_{i-1} \end{pmatrix}^T. \tag{37}$$

Since we assume that the follower R_i has no a priori information about the kinematics or trajectory of R_{i-1} , it is natural to represent the motion with an ordinary random walk model:

$$\dot{\mathbf{x}} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \mathbf{w}, \tag{38}$$

where $\mathbf{w} = (\ddot{x}_{i-1} \ddot{y}_{i-1})^T$ is the unknown acceleration of R_{i-1} , which is modeled as gaussian noise. As (x_{i-1}, y_{i-1}) can be measured in the sense that they can be computed from sensor measurements, we define the measurement \mathbf{y} of the dynamic system (38) as

$$\mathbf{y} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \mathbf{x} + \mathbf{v}. \tag{39}$$

where \mathbf{v} is unknown, gaussian measurement noise. To enable application of the discrete Kalman filter, the system (38) - (39) is discretized using an sample time h . Denoting the discrete time steps with t , we get

$$\mathbf{x}_{t+1} = \begin{pmatrix} 1 & 0 & h & 0 \\ 0 & 1 & 0 & h \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{x}_t + \begin{pmatrix} \frac{h^2}{2} & 0 \\ 0 & \frac{h^2}{2} \\ h & 0 \\ 0 & h \end{pmatrix} \mathbf{w}_t \quad (40)$$

$$\mathbf{y}_t = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \mathbf{x}_t + \mathbf{v}_t.$$

It is easy to show that this system is observable. The output of the Kalman filter is

$\hat{\mathbf{x}} = (\hat{x}_{i-1} \ \hat{y}_{i-1} \ \hat{\dot{x}}_{i-1} \ \hat{\dot{y}}_{i-1})^T$, from which the velocity estimate $v_{i-1} = \sqrt{\hat{\dot{x}}_{i-1}^2 + \hat{\dot{y}}_{i-1}^2}$ is finally obtained. In the next section, \mathbf{y} is derived from sensor measurements.

7.3 Computation of Kalman Filter Output

\mathbf{y} in (39) can be derived from sensor data, using triangulation for pairs of directional sensors, as follows. For the Khepera II we define 6 neighboring sensor pairs (Figure 4):

$$\begin{aligned} p_1 &= \{s_1 \ s_2\}, \ p_2 = \{s_2 \ s_3\}, \ p_3 = \{s_3 \ s_4\}, \\ p_4 &= \{s_4 \ s_5\}, \ p_5 = \{s_5 \ s_6\}, \ p_6 = \{s_7 \ s_8\}. \end{aligned} \quad (41)$$

When there is no risk of confusion, we will use the notation s_q to specify both the sensor indexed $q \in [1, 8]$, and the global position of that sensor. For an explanation of the notation used in this section, see Figure 6.

The IR sensors of the Khepera II measure reflected light intensity. The relation between light intensity and distance for the IR sensors depends on factors such as ambient light and reflecting material. For the initial experiments presented in Section 8, the measured relation is shown in Figure 5. Let $\tilde{\delta}_q$ be the measured distance to the target from sensor s_q , and define the

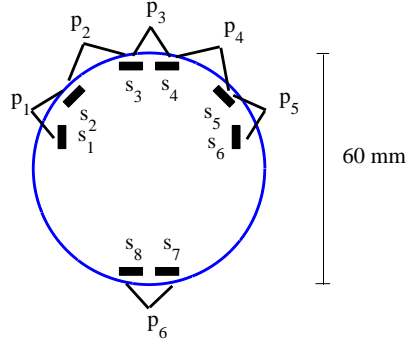


Fig. 4. The sensor pairs of Khepera II.

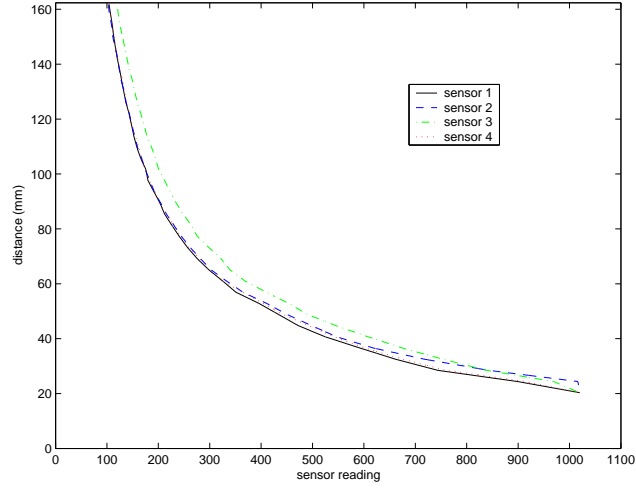


Fig. 5. The relation between sensor output and distance, as measured by the experimental setup in Section 8

distance δ_q from sensor s_q to (x_{i-1}, y_{i-1}) :

$$\delta_q \triangleq \tilde{\delta}_q + r. \quad (42)$$

Here, r denotes the radius of R_{i-1} . Let $p_j = \{s_q, s_{q-1}\}$, $\Delta s_j = \|s_q - s_{q-1}\|$ and define, using the law of cosines, the angle ψ_j such that

$$\delta_{q-1}^2 = \Delta s_j^2 + \delta_q^2 - 2\Delta s_j \delta_q \cos \psi_j. \quad (43)$$

Denote by \mathbf{y}_j the target position estimate associated with p_j . All variables are defined in a local coordinate system O' with x -axis parallel to the difference vector $s_q - s_{q-1}$ as shown in Figure 6. Expressed in O' , we obtain

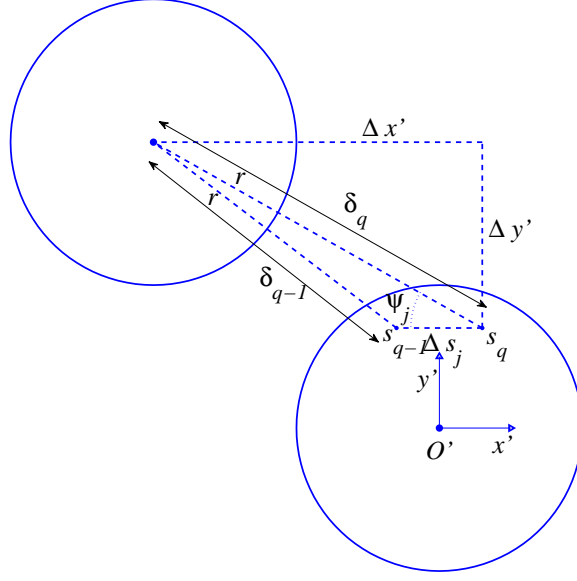


Fig. 6. Notation for computing the position of R_{i-1} from sensor measurements. Here s_q and s_{q-1} form the sensor pair p_j .

$$\begin{aligned}\Delta x'_j &= \delta_q \cos \psi_j \\ \Delta y'_j &= \delta_q \sin \psi_j.\end{aligned}\tag{44}$$

This means that the position of the target, as measured by p_j , in the frame O' is

$$\mathbf{y}'_j = \begin{pmatrix} x'_q - \Delta x'_j & y'_q + \Delta y'_j \end{pmatrix}^T = (x'_{i-1} \ y'_{i-1})^T\tag{45}$$

where (x'_q, y'_q) are the coordinates of the sensor s_q in the local frame. Expressed in the global coordinate system, we get

$$\mathbf{y}_j = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x'_{i-1} \\ y'_{i-1} \end{pmatrix} + \begin{pmatrix} x_i \\ y_i \end{pmatrix},\tag{46}$$

where θ is the rotation of O' compared to the global frame. θ and (x_i, y_i) are obtained from the odometry of R_i . Before computing control inputs, a critical issue is to decide from which sensor pairs to record measurements. This is discussed in the next section, and a formalization of the procedure is provided.

7.4 Switching Kalman Filters

In this section we derive a switching Kalman filter suitable for target tracking using directional sensors. It is clear from Figure 2 that each sensor has

a limited field of view. We say that a sensor is *active* if the target is in its field of view. If a Kalman filter F_j is designed for each sensor pair p_j and they are allowed to act on the discretized system (40), F_j will only give a fair estimation if both sensors in p_j are active. Further, note that (43) and (44) yield

$$\Delta y'_j = \sqrt{\delta_q^2 - \Delta x_j'^2} = \delta_q \sqrt{1 - \left(\frac{\Delta s_j^2 + \delta_q^2 - \delta_{q-1}^2}{2\Delta s_j \delta_q} \right)^2} \quad (47)$$

which is imaginary for $(\Delta s_j^2 + \delta_q^2 - \delta_{q-1}^2 / 2\Delta s_j \delta_q)^2 > 1$. Naturally, the true distance $\Delta y'_j$ is a real number. However, the estimate computed from (47) can be imaginary if too much measurement noise is present in the sensor readings δ_q and δ_{q-1} . We will therefore use the following definition.

Definition 1 *A filter F_j is active at time step t if and only if*

- (1) *Both sensors in the associated sensor pair p_j are active*
- (2) *For measurements from the sensor pair p_j , it holds that*

$$\left(\frac{\Delta s_j^2 + \delta_q^2 - \delta_{q-1}^2}{2\Delta s_j \delta_q} \right)^2 < 1. \quad (48)$$

In order to estimate the movement of R_{i-1} , R_i should only pay attention to the output of active filters. Which filters F_j that are active at each time step will vary with the path of the target and the tracking parameters. This motivates the use of a switching Kalman filter.

The switching Kalman filter consists of the subfilters F_j , which all run in parallel to each other. At each time step, the estimates $\hat{\mathbf{x}}_{j,t}$ from the *active* subfilters are merged into an estimate $\hat{\mathbf{x}}_t$ for the switching filter. $\hat{\mathbf{x}}_t$ can be either a linear combination of the state estimates from the active filters, or the best estimate $\hat{\mathbf{x}}_{j,t}$ with respect to some criterion.

A summary of the switching Kalman filter scheme follows in Algorithm 7.1.

Algorithm 7.1 Target Tracking with a Switching Kalman Filter

For each follower agent R_i , at each time step t :

- (1) *Record, using odometry, the agent's own position at time t : $(x_{i,t}, y_{i,t}, \phi_{i,t})$.*
- (2) *Record range measurements from the IR sensors. If s_{q-1} and s_q both get readings, the associated sensor pair p_j is added to the list of candidates for active filters.*
- (3) *Check condition (48). If (48) holds for p_j , this sensor pair is added to the final list of active sensor pairs.*

- (4) Compute $\mathbf{y}_{j,t}$ for all active sensor pairs using (42) - (46).
- (5) Apply the Kalman filter (36) to the system (40) using $\mathbf{y}_{j,t}$ to obtain estimates $\hat{\mathbf{x}}_{j,t}$ for all active filters.
- (6) Compute or choose the final state estimate $\hat{\mathbf{x}}_t = (\hat{x}_{i-1,t} \ \hat{y}_{i-1,t} \ \hat{\dot{x}}_{i-1,t} \ \hat{\dot{y}}_{i-1,t})^T$ from the estimates $\hat{\mathbf{x}}_{j,t}$.
- (7) Compute

$$v_{i-1,t} = \sqrt{\hat{\dot{x}}_{i-1,t}^2 + \hat{\dot{y}}_{i-1,t}^2} \quad (49)$$

$$d_{i,t} = \sqrt{(\hat{x}_{i-1,t} - x_{i,t})^2 + (\hat{y}_{i-1,t} - y_{i,t})^2} \quad (50)$$

$$\beta_{i,t} = \tan^{-1} \frac{\hat{y}_{i-1,t} - y_{i,t}}{\hat{x}_{i-1,t} - x_{i,t}} - \phi_{i,t} \quad (51)$$

$$\phi_{i-1,t} = \tan^{-1} \frac{\hat{y}_{i-1,t} - \hat{y}_{i-1,t-1}}{\hat{x}_{i-1,t} - \hat{x}_{i-1,t-1}} \quad (52)$$

$$\gamma_{i,t} = \phi_{i,t} - \phi_{i-1,t}. \quad (53)$$

- (8) Compute and apply control inputs $v_{i,t}$ and $\omega_{i,t}$ from Control 1, Control 2 or Control 3.

In the next section, the controls discussed in previous sections are demonstrated in experiments and simulations.

8 Experiments and Simulations

In this section, initial results from experiments and simulations are provided. The results in this section serve to illustrate the theoretical results in previous sections. An extensive experimental evaluation remains for future work and will be further discussed in Section 9. First, the outcome of experiments using Control 1 and Algorithm 7.1 are reported. Then, simulation results for Control 2 and Control 3 are included to demonstrate advantages of refining the control.

8.1 Experiments: Control 1 and Algorithm 7.1

The platform used in the experiments is the Khepera II (Figure 2) which has unicycle kinematics and 8 directional IR sensors. The sensors have a range of up to 100 mm but the quality of data is highly sensitive to factors such as ambient light. For localization, the Khepera II uses odometry.

Algorithm 7.1 was used to obtain position and velocity estimates. The switching strategy was to choose $\hat{\mathbf{x}}_t$ as the estimate $\hat{\mathbf{x}}_{j,t}$ among the outputs

of the active filters F_j that differed the least from the previous estimate $\hat{\mathbf{x}}_{t-1}$. This is a rough and simple strategy, best suited for constant or slowly varying parameters. More sophisticated strategies should be investigated in future work.

The experiments were performed with a setup of one leader (R_1) following a predefined path, and one follower (R_2). The platforms were controlled through an interface in MATLAB. No communication occurred between the agents.

Results are shown for robustness of a rigid formation as well as for switching bearing angle. The figures in this section contain plots of Δd_2 and β_2 computed from the Kalman filter output. Note that the scaling differs somewhat between plots, depending on the magnitude of the deviations of the curves from the reference values.

Robustness of Control A.1

To investigate the robustness of Control 1, R_1 followed a predefined path at constant speed, and R_2 tracked R_1 with a given reference angle, $\beta_{0,2}$, and a reference distance of $d_{0,2} = 120$ mm (from center to center).

The formation error $\Delta d_2 \equiv d_2 - d_{0,2}$ and angle β_2 , together with the trajectories of the robots, are shown in Figure 7 - 8 for $\beta_{0,2} = 0$ rad and $\beta_{0,2} = \pi/6$ rad respectively. For $\beta_{0,2} = 0$ rad, β_2 converges nicely and stays at 0 rad after a transient of about 15 s. $\beta_{0,2} = \pi/6$ rad is a more challenging case for the Khepera II, as this angle lies at the edge of the field of view for two sensors (s_2 and s_3 in Figure 4). As seen in Figure 8, β_2 fluctuates around its reference value and does not converge. This is due to the fact that for this challenging angle, the active filter switches back and forth between F_2 , F_3 , and sometimes F_1 (see Figure 4). In addition, the drifting error in the odometry used for localization contributes to measurement uncertainties. To reduce oscillations, a more sophisticated switching strategy, such as a weighted mean estimate from several filters, should be applied. This is an issue for further study. However, despite fluctuations, R_2 is able to follow R_1 , R_1 remains within the sensor range of R_2 at all times, and the errors are bounded.

Robustness for switching β_0

To investigate the robustness of a switch of the reference angle, the same setup as in Section 8.1 was used. Again, R_1 followed a predefined path with constant speed, and R_2 tracked R_1 with reference angle $\beta_{0,2} = 0$ rad, and a reference distance of $d_{0,2} = 120$ mm. The reference angle was switched to $\beta_{0,2} = \pi/6$ rad at $t = 10$ s, and then switched back to $\beta_{0,2} = 0$ rad at $t = 60$ s.

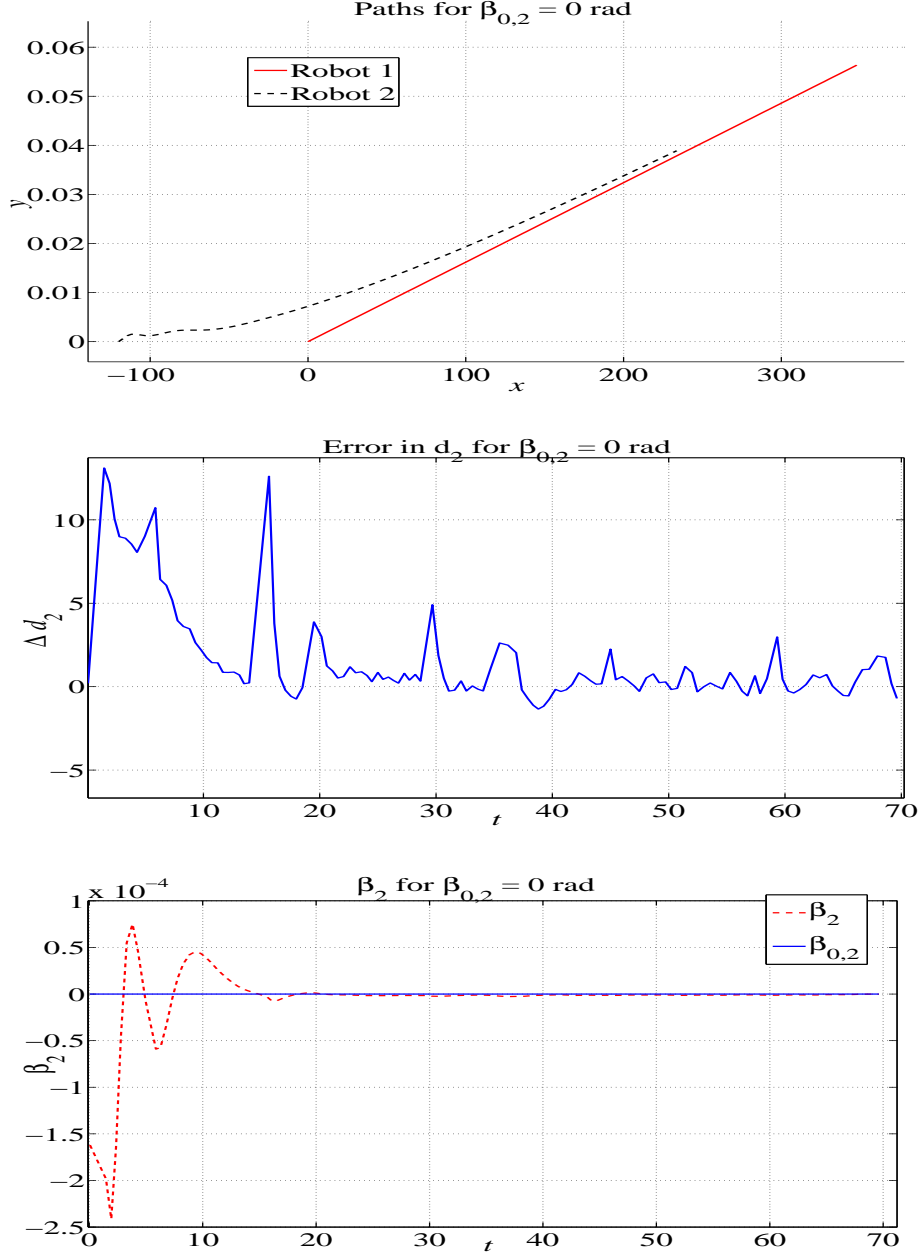


Fig. 7. **Robustness of Control A.1:** $\beta_0 = 0$ rad. Top: The trajectories of the platforms. Middle: The difference $\Delta d_2(t)$ [mm]. Bottom: The angle $\beta_2(t)$ rad and the bearing angle $\beta_{0,2}(t)$ rad.

The formation error Δd_2 and angle β_2 are shown, together with the trajectories, in Figure 9. After a transient of roughly 20 s, Δd_2 decreases and fluctuates around a small static error despite switch in $\beta_{0,2}$. R_2 responds quickly to the change in $\beta_{0,2}$ and although fluctuations are present, we observe no static error in β_2 . R_1 remains within the field of view of R_2 throughout the experiment, and the errors remain bounded.

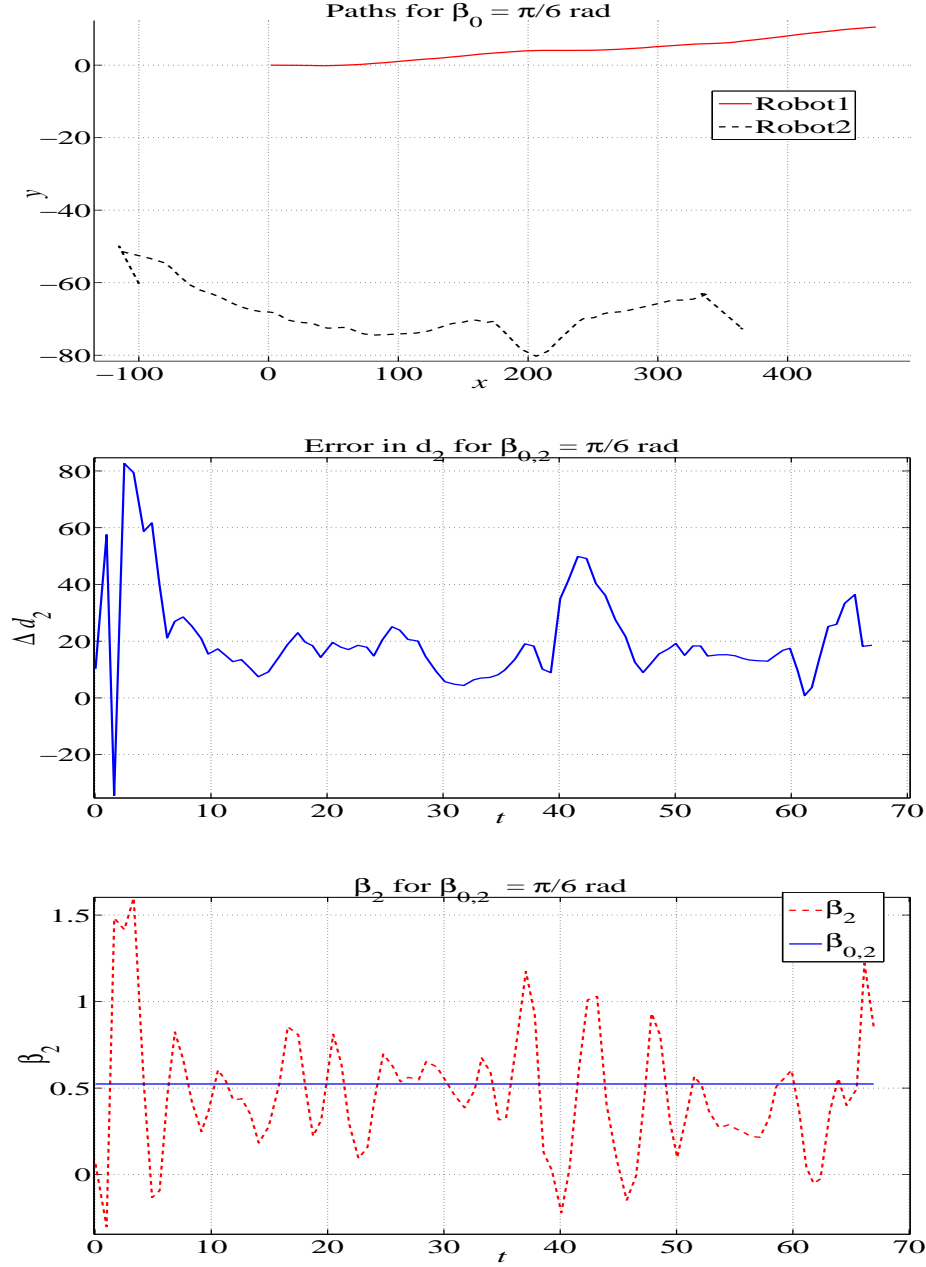


Fig. 8. **Robustness of Control A.1:** $\beta_0 = \pi/6$ rad. Top: The trajectories of the platforms. Middle: The difference $\Delta d_2(t)$ [mm]. Bottom: The angle $\beta_2(t)$ rad and the bearing angle $\beta_{0,2}(t)$ rad.

8.2 Simulations: Control 2

To demonstrate the usefulness of Control 2, simulations were carried out for two agents R_1 and R_2 using Control 1 and Control 2, which drives $\beta_i - \beta_{0,i}$ to zero monotonically. The agents were modeled as unicycle platforms. For $v_1 = 10$, $\omega_1 = \pi/3$, $k = 1$, $\beta_{0,2} = \pi/7$, $d_{0,2} = 1$,

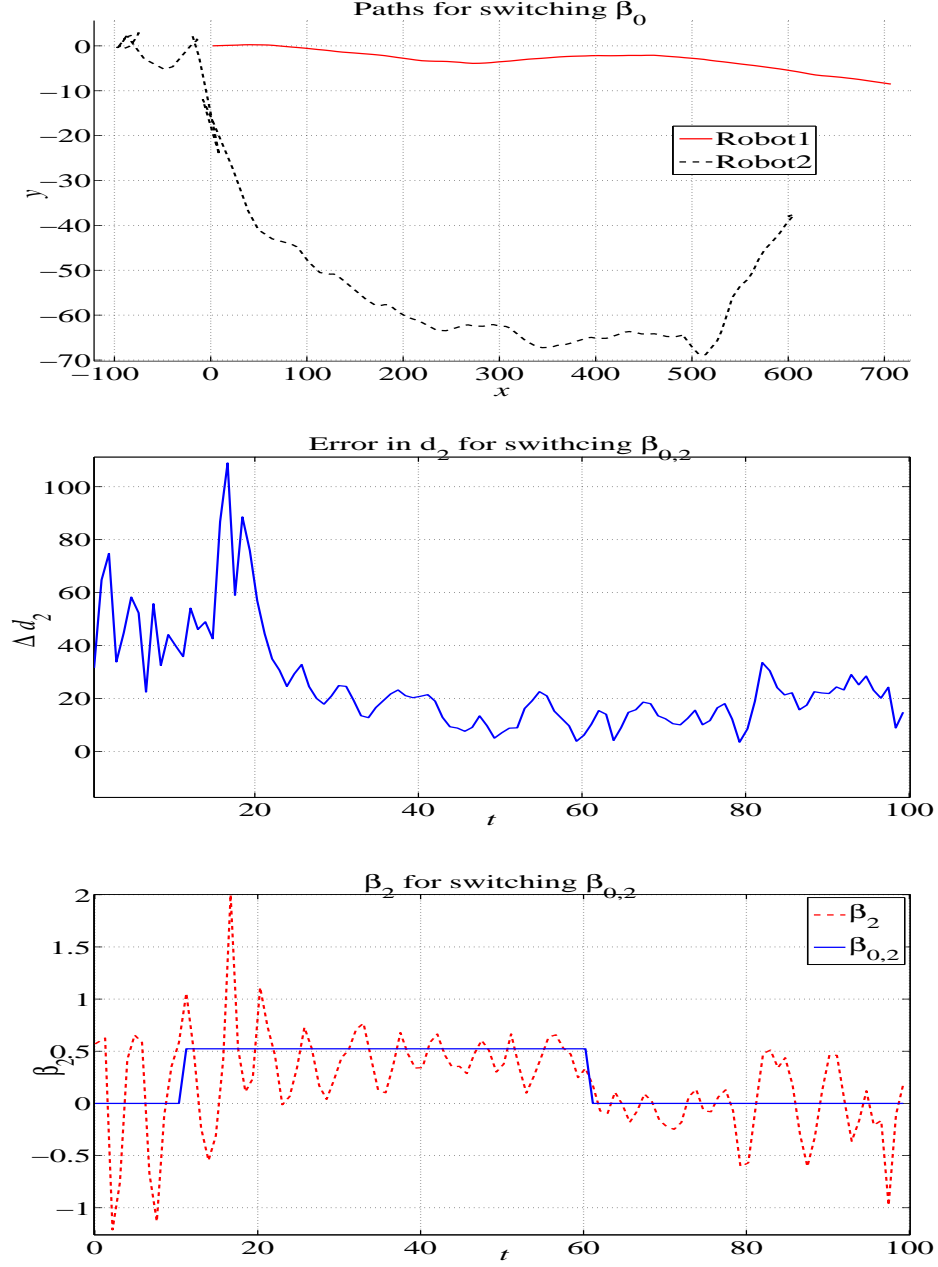


Fig. 9. **Robustness for switching β_0 :** Experiment with switch from $\beta_0 = 0$ rad to $\beta_0 = \pi/6$ rad. Top: The trajectories of the platforms. Note that, since R_2 tracks R_1 at a non-zero angle for $t \in [10, 60]$, its track deviates from that of R_1 . Middle: The difference $\Delta d_2(t)$ [mm]. Bottom: The angle $\beta_2(t)$ rad and the bearing angle $\beta_{0,2}(t)$ rad.

$(x_1, y_1, \phi_1) = (0, 0, 0)$, $(x_2, y_2, \phi_2) = (-0.97, -0.70, \pi/10)$ error convergence is studied. The measured entities are the deviations from reference values of control parameters ($d_2 - d_{0,2}$ and $\beta_2 - \beta_{0,2}$), and position error (x_e and y_e). Although all errors converge to 0, they do not do so monotonically for both controls. Results are plotted in Figures 10 - 12. Figure 10 shows the

resulting trajectories for R_1 and R_2 . Figure 11 shows convergence of d_2 and β_2 toward the reference values for both controls. Clearly, Control 2 yields a monotonic convergence while Control 1 does not. If R_2 is equipped with directional sensors, it may risk to loose tracking for $t \in (0,4)$ when using Control 1. On the other hand, as is seen in Figure 12, the position error converges monotonically with Control 1 but not with Control 2. It should however be noted that if the control gain k can be increased sufficiently, all errors will decrease monotonically for both controls.

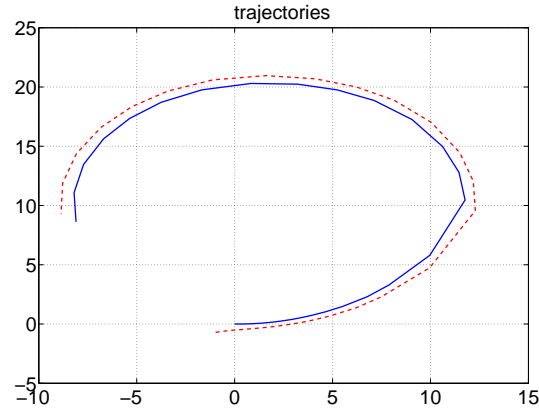


Fig. 10. **Controll A.2:** Trajectory of R_1 (solid) and R_2 (dashed).

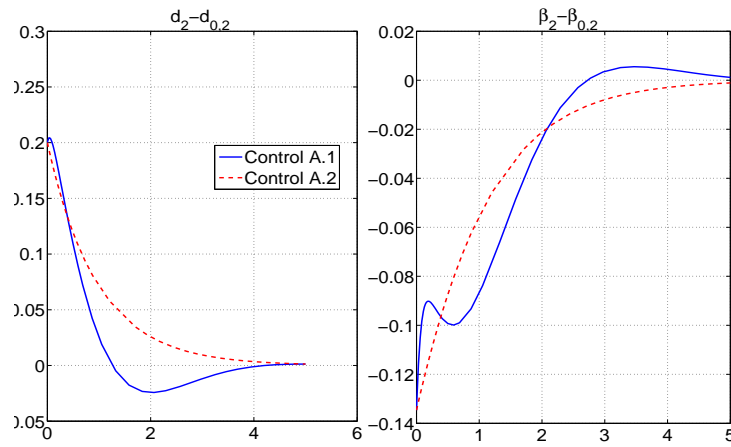


Fig. 11. **Controll A.2:** Convergence of $d_2 - d_{0,2}$ and $\beta_2 - \beta_{0,2}$ for Control 1 (solid) and Control 2 (dashed). It is clear that the convergence is monotonic for Control 2 but not for Control 1.

8.3 Simulations: Dynamic Parameters

Advantages of Dynamic Parameters

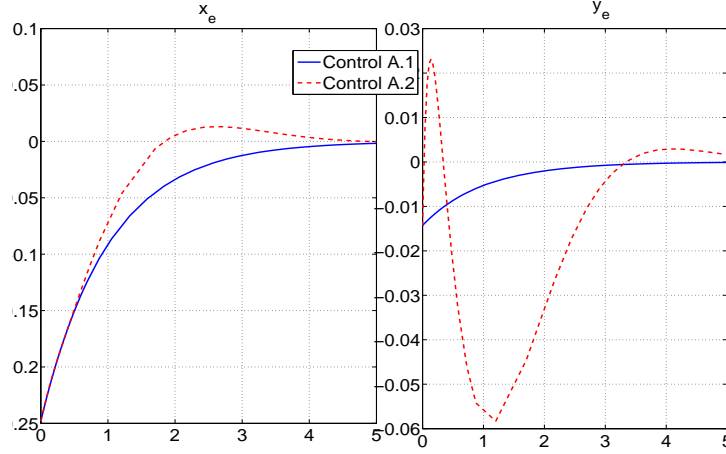


Fig. 12. **Controll A.2:** Convergence of x_e and y_e for Control 1 (solid) and Control 2 (dashed). It is clear that the convergence is monotonic for Control 1 but not for Control 2.

To demonstrate potential advantages of dynamic control parameters, we consider a scenario where the formation strives to stay on the same path as the leader at all times. We defined dynamic control parameters $d_{0,i}(t)$ and $\beta_{0,i}(t)$ such that, given the path of agent R_{i-1} up until time t , agent R_i should steer toward a position located on that path such that the distance $d_{0,i}(t)$ is maximized while the area shown in Figure 13 is smaller than a tolerance value A^{max} . Simulations were performed for a team of 5 agents using first

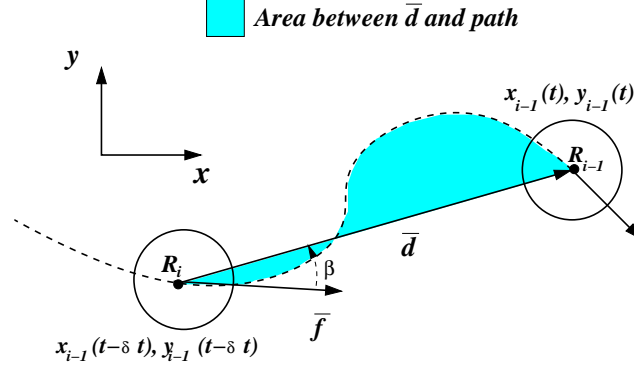


Fig. 13. **Advantages of Dynamic Parameters:** $d_{0,i}(t)$ is chosen as the maximum distance that keeps the tinted area smaller than A^{max} .

dynamic, and then constant, $d_{0,i}$ and $\beta_{0,i}$ and results were compared. For both cases we used Control 1. The leader agent R_1 traveled along a pre-defined path, which is unknown to the followers. The simulation setup is illustrated in Figure 14. For the dynamic parameters, letting $A^{max} = 0.005$ resulted in a mean inter-agent distance of 1.19 length units. In the case with constant parameters, we use reference angle 0 rad and reference distance equal for all agents. This distance was chosen as $d_{0,i} = 1.19$, $i = 2, \dots, 5$ to get as fair a comparison as possible. For both controls, we used the gain

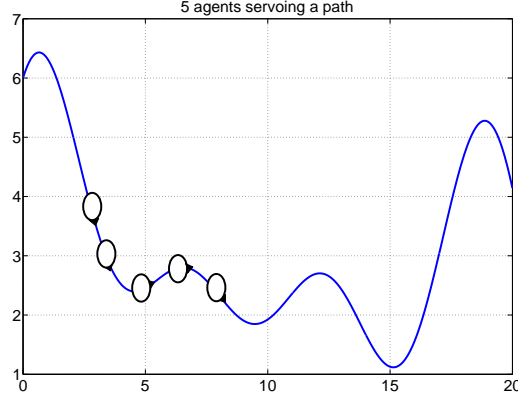


Fig. 14. **Advantages of Dynamic Parameters:** A team of one leader and 4 follower agents tracking a path.

$k = 1$. At each time step, the deviation for R_i from the path of R_{i-1} was measured as

$$dev_i = \min_{\tau \in (0, t]} \|R_{i-1}(\tau) - (x_i(t), y_i(t))\|. \quad (54)$$

Results are shown in Figures 15 - 18. Figures 15 - 16 show a significant improvement in path following using dynamic parameters. Figures 17 - 18 show how $d_{0,i}$ and $\beta_{0,i}$ vary for the dynamic parameter control. High curvature parts of the path cause the formation to contract to enable safer path following.

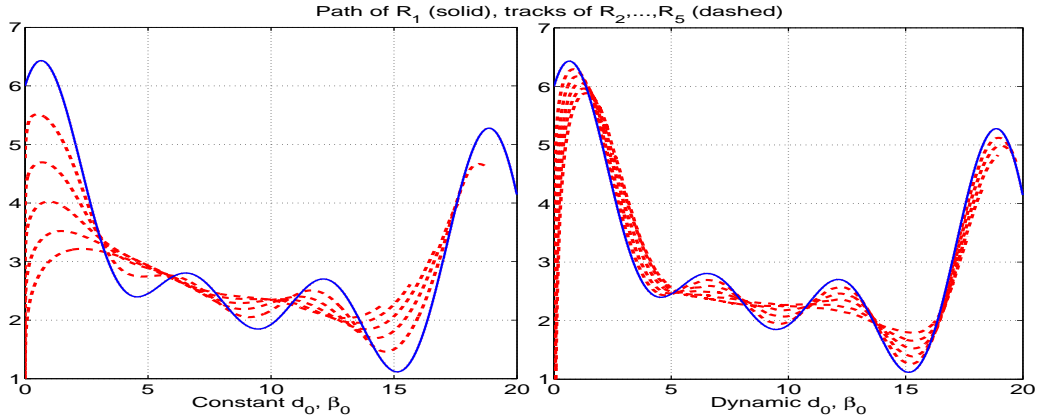


Fig. 15. **Advantages of Dynamic Parameters:** Tracks (dashed) for the follower agents. Left: Constant $d_{0,i}$ and $\beta_{0,i}$. Right: Dynamic $d_{0,i}$ and $\beta_{0,i}$.

Control 3

Finally, an example is included to illustrate the improvement of the error dynamics with Control 3. We used two agents and defined $d_{0,2}(t)$ and $\beta_{0,2}(t)$

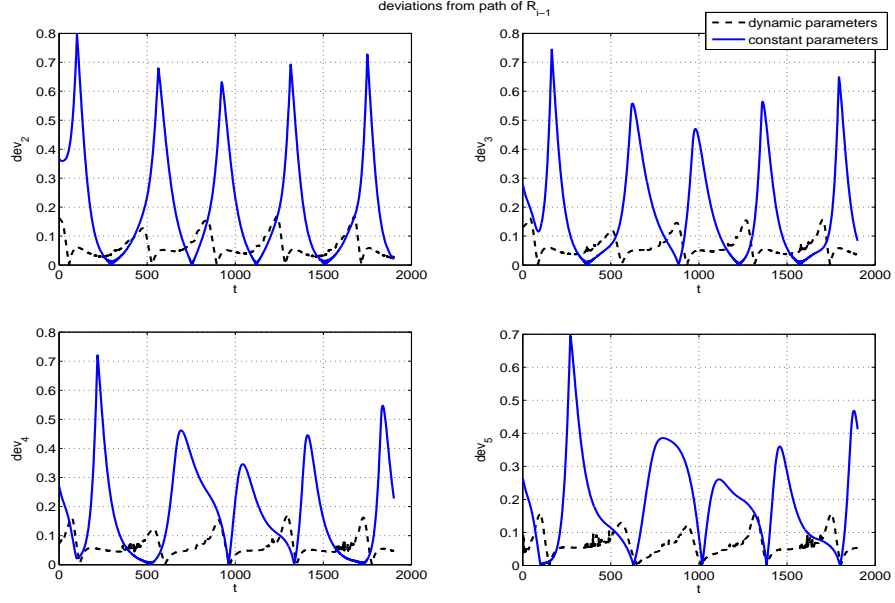


Fig. 16. **Advantages of Dynamic Parameters:** Deviations from the path of R_{i-1} for the four followers.

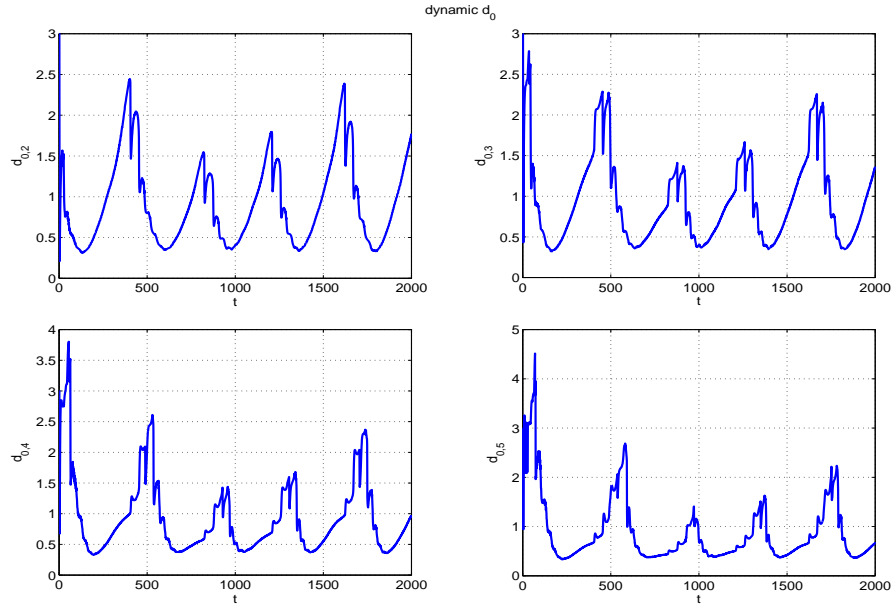


Fig. 17. **Advantages of Dynamic Parameters:** Dynamic $d_{0,i}$.

by

$$\begin{aligned}
 d_{0,2} &= 0.1 \cos(45\beta_{0,2}) \\
 \dot{\beta}_{0,2} &= 0.001 \\
 d_{0,2}(0) &= 0.5 \\
 \beta_{0,2}(0) &= \pi/7.
 \end{aligned} \tag{55}$$

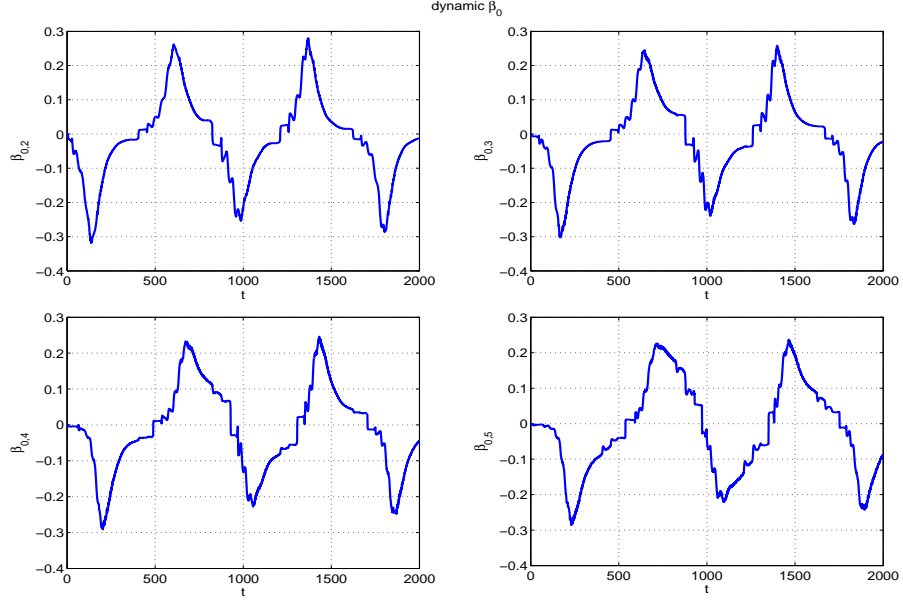


Fig. 18. Advantages of Dynamic Parameters: Dynamic $\beta_{0,i}$.

The resulting functions $d_{0,2}(t)$ and $\beta_{0,2}(t)$ are shown in Figure 19. For $v_1 =$

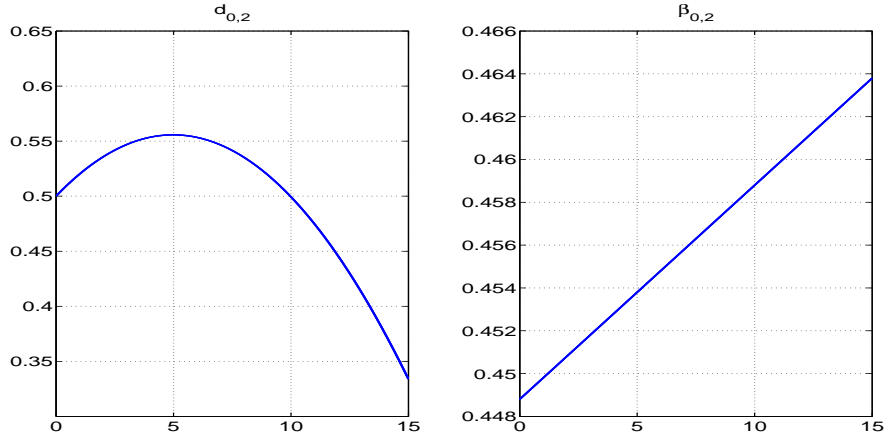


Fig. 19. **Control 3:** The dynamic parameters resulting from Equation (55). Left: $d_{0,2}(t)$. Right: $\beta_{0,2}(t)$.

$10, \omega_1 = \pi/3, k = 1, \beta_{0,2}(0) = \pi/7, d_{0,2}(0) = 1, (x_1, y_1, \phi_1) = (0, 0, 0), (x_2, y_2, \phi_2) = (-0.97, -0.70, \pi/10)$, we let R_2 track R_1 using Control 1, Control 2, and Control 3. The gain $k = 1$ was used for all three cases. Results are shown in Figures 20 - 21. It is clear from Figure 20 that both Control 1 and Control 2 yield static or even increasing errors in d_2 and β_2 , while Control 3 drives the errors to zero. The same behavior is apparent for the errors x_e and y_e , shown in Figure 21.

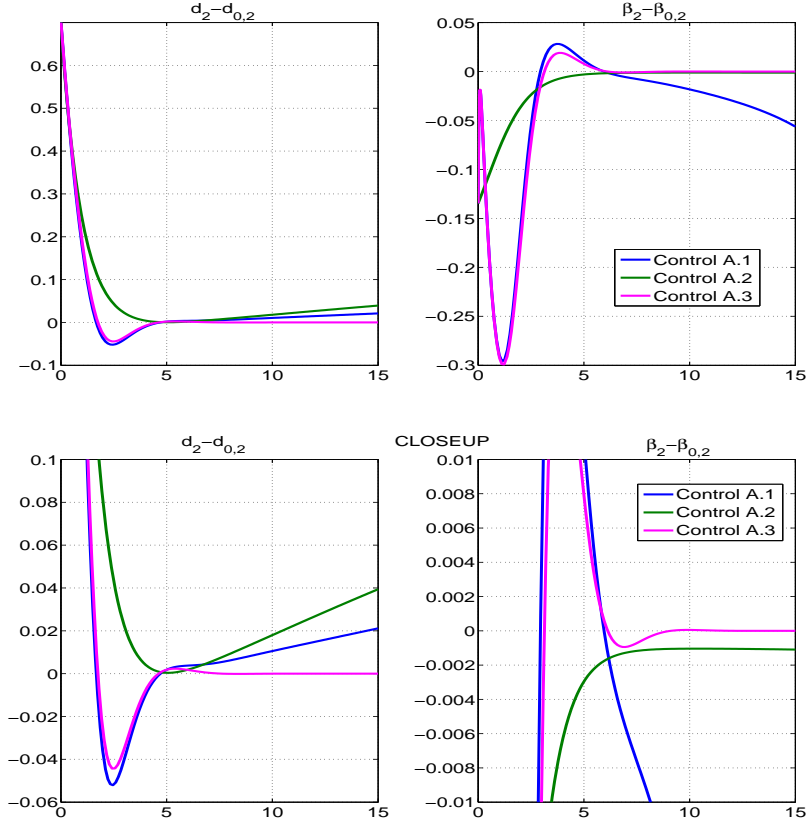


Fig. 20. **Control 3:** Left: $d_2 - d_{0,2}(t)$. Right: $\beta_2 - \beta_{0,2}(t)$. Top: full scale. Bottom: closeup view.

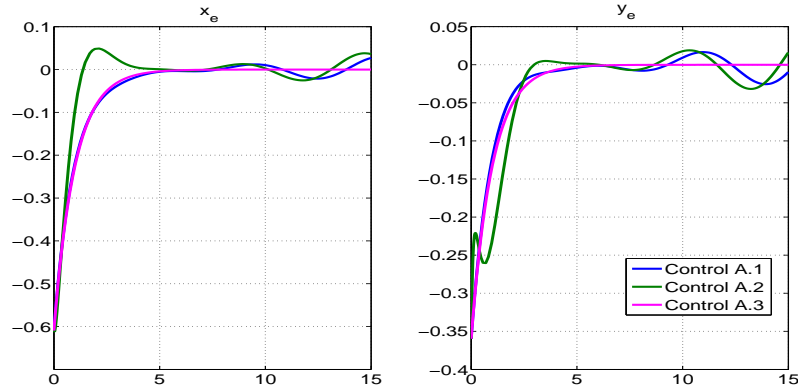


Fig. 21. **Control 3:** Left: $x_e(t)$. Right: $y_e(t)$.

9 Conclusion

In this paper, cascaded controls were proposed for formations of autonomous agents. The controls are based on feedback from directional and noise contaminated range sensors, and a switching Kalman filter was introduced to

enable target tracking for sensors with limited field of view. Theoretical results testified to stability and robustness, as well as flexibility, of the controls. Initial results from experiments and simulations were provided to illustrate the theoretical results. Combined with behaviors like obstacle avoidance and failure detection, the formation controls proposed in this paper are suitable for applications such as servoing, surveillance and mapping.

Future work includes a thorough experimental evaluation of the proposed control algorithms. Issues of interest are for instance scalability and robustness with respect to noisy measurements, and refinement of the switching criterion for the Kalman filter algorithm. A comparative study should also be made in order to evaluate the proposed controls in contrast to for instance the hybrid control proposed in [5], and the Kalman filter approach in contrast to the non-linear observer in [19].

References

- [1] C.W. Reynolds, Flocks, Herds, and Schools: A Distributed Behavioral Model, *Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.
- [2] T. Balch and R.C. Arkin, Behavior-based formation control for multirobot teams, *Robotics and Automation, IEEE Trans. on*, vol. 14, no. 6, pp. 926–939, 1998.
- [3] M. Egerstedt and Xiaoming Hu, Formation constrained multi-agent control, *Robotics and Automation, IEEE Trans. on*, vol. 17, no. 6, pp. 947–951, 2001.
- [4] T. Gustavi and X. Hu, Formation Control for Mobile Robots with Limited Sensor Information, *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 367–393, 2005.
- [5] David J. Naffin, Gaurav S. Sukhatme, and Mehmet Akar, Lateral and Longitudinal Stability for Decentralized Formation Control, *Proceedings of the International Symposium on Distributed Autonomous Robotic Systems*, pp. 421–430, June 2004.
- [6] N.E. Leonard and E. Fiorelli, Virtual leaders, artificial potentials and coordinated control of groups, *Proc. of the 40th IEEE Conference on Decision and Control (CDC)*, vol. 3, no. pp. 296–2973, 2001.
- [7] N.E. Leonard, P. Ögren, and E. Fiorelli, Formations with a mission: stable coordination of vehicle group maneuvers, *International Symposium on Mathematical Theory of Networks and Systems (MTNS)*, 2002.
- [8] M Karasalo T Gustavi, X Hu, Formation adaptation with limited sensor information, *invited paper, Proc. of Chinese Control Conference (CCC)*, 2005.

- [9] H.G. Tanner, G.J. Pappas, and V. Kumar, Leader-to-formation stability, *IEEE Trans. on Robotics and Automation*, vol. 20, no. 3, pp. 443–455, 2004.
- [10] M. Egerstedt, X. Hu, and A. Stotsky, Control of Mobile Platforms Using a Virtual Vehicle Approach, *IEEE Trans. Automatic Control*, vol. 46, no. 11, 2001.
- [11] A.K. Das, R. Fierro, V. Kumar, J.P. Ostrowski, J. Spletzer, and C.J. Taylor, A vision based formation control framework, *IEEE Trans. on Robotics and Automation*, vol. 18, pp. 813–825, 2002.
- [12] J. Lawton, R. Beard, and B. Young, A centralized Approach to Formation Manuevers, *IEEE Trans. on Robotics and Automation*, vol. 19, pp. 933–941, 2003.
- [13] M. Mazo, A. Speranzon, K.H. Johansson, and X. Hu, Multi-Robot Tracking of a Moving Object Using Directional Sensors, *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, 2004.
- [14] R. Olfati-Saber and R.M. Murray, Graph Rigidity and Distributed Formation Stabilization of Multi-Vehicle Systems, *Proc. of IEEE Conference on Decision and Control (CDC)*, 2002.
- [15] A. Jadbabaie, J. Lin and A. S. Morse, Coordination of groups of mobile autonomous agents using nearest neighbor rules, *IEEE Trans. Automat. Control*, vol. 48, no. 6, pp. 988–1001, 2003.
- [16] K.D. Murphy, Switching Kalman Filter, *Technical Report, Compaq Cambridge Research Laboratory*, 1998.
- [17] R. Vidal, O. Shakernia, and S. Sastry, Formation Control of Nonholonomic Mobile Robots with Omnidirectional Visual Servoing and Motion Segmentation, *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 584–589, 2003.
- [18] E. Bicho and S. Monteiro, Formation control for multiple mobile robots: a non-linear attractor dynamics approach, *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2016–2022, 2003.
- [19] T. Gustavi and X.Hu, Observer Based Leader-Following Formation Control using On-Board Sensor Information, *IEEE Trans. on Robotics*, vol. 24, no. 6, pp. 1457–1462, 2008.
- [20] C.A. Reeder, D.L. Odell, A. Okamoto, M.J. Anderson, and D.B. Edwards, Two-hydrophone heading and range sensor applied to formation-flying for AUVs, *Proc. of MTTT/IEEE TECHNO-OCEAN '04*, pp. 517–523, 2004.
- [21] S. Kim, C. Ryoo, K. Choi, and C. Park, Multi-vehicle formation using range-only measurement, *Proc. of IEEE International Symposium on Circuits and Systems (ICCAS)*, pp. 2104–2109, 2007.