Chapter 13
Decentralized Formation Control in Fleets of Nonholonomic Robots with a Clustered Pattern

Marcos Cesar Bragagnolo, Irinel-Constantin Morărescu, Lucian Buşoniu, and Pierre Riedinger

Abstract In this work we consider a fleet of non-holonomic robots that has to realize a formation in a decentralized and collaborative manner. The fleet is clustered due to communication or energy-saving constraints. We assume that each robot continuously measures its relative distance to other robots belonging to the same cluster. Due to this, the robots communicate on a directed connected graph within each cluster. On top of this, in each cluster there exists one robot called leader that receives information from other leaders at discrete instants. In order to realize the formation we propose a two-pronged strategy. First, the robots compute reference trajectories using a linear consensus protocol. Second, a classical tracking control strategy is used to follow the references. Overall, formation realization is obtained. Numerical simulations with small and large robot teams illustrate the effectiveness of this approach.

13.1 Introduction

In order to facilitate human tasks, during the last century many researchers focused on automation and robotics. The main goal was to design mechanisms and appropriate control laws that are able to operate in harsh environments and/or execute difficult or repetitive tasks. The control of single, stand-alone systems is extensively

Marcos Cesar Bragagnolo, Irinel-Constantin Morărescu, Pierre Riedinger
Université de Lorraine, CRAN, UMR 7039, 2 Avenue de la Forêt de Haye, France and CNRS, CRAN, UMR 7039, e-mail: marcos-cesar.bragagnolo@univ-lorraine.fr, constantin.morarescu@univ-lorraine.fr, pierre.riedinger@univ-lorraine.fr

Lucian Buşoniu
Automation Department, Technical University of Cluj-Napoca, Memorandumului 28, 400114 Cluj-Napoca, Romania, e-mail: lucian.busoniu@aut.utcluj.ro

studied and applied in industry, see e.g. (Moreau, 1988; McClamroch and Wang, 1988; Brogliato et al, 2007; Morărescu and Brogliato, 2010a,b).

New challenges related to networked control systems arose at the end of the 20th century. Since then, this domain is flourishing, because many current engineering problems require multiple systems with local sensing and actions, which have to collaborate in order to accomplish a global goal. One example is in the area of sensor networks, where nodes form a cluster and a cluster leader is selected. This leader communicates with other clusters to minimize energy consumption (Sun et al, 2006; Halgamuge et al, 2003). Issues related to communication between different devices in the network with minimal network use, while ensuring a desired level of performance, are solved by using time and event triggered controllers (Anta and Tabuada, 2010; Heemels et al, 2012; Postoyan et al, 2011). The global goal is often formulated in terms of agreement or consensus (Michiels et al, 2009) and the systems involved in the process are called agents. Beside the classical full or partial state consensus (Jadbabaie et al, 2003; Moreau, 2005; Martin and Girard, 2013), other applications can be handled by this framework, such as persistent monitoring (Lin and Cassandras, 2014) or control of power consumption in smart-grids (Ratliff et al, 2014). With slight modifications, the consensus problem can be reformulated to represent formation realization goals (Brinon Arranz et al, 2014; Ren and Beard, 2004).

An important type of such a network of agents, where formation realization is particularly relevant, is a team of mobile robots. Such a multi-robot system can adapt better to changes in the mission goals than a single robot, therefore providing a higher degree of flexibility and scalability. Robot formations have a wide variety of applications, both civilian and military (Bertuccelli et al, 2009), such as scanning an interesting area, protecting a given target (Ding et al, 2009; Mahacek et al, 2011), performing naval mine countermeasure missions (Sariel et al, 2006), or protecting convoys (Ding et al, 2009). For two surveys in the context of Unmanned Aerial Vehicles (UAVs), see (Scharf et al, 2003, 2004; Samad et al, 2007).

For the purposes of formation realization, the robots are often assumed to be very simple, being modeled as single or double integrators (Beard and Stepanyan, 2003; Bullo et al, 2009; Tanner et al, 2007; Su et al, 2011; Fiacchini and Morărescu, 2014). In reality, most mobile robots have nonholonomic dynamics, which are nontrivial to control, as noticed e.g. by (Kolmanovsky and McClamroch, 1995; Jiang and Nijmeijer, 1997; Samson and Ait-Abderrahim, 1991). Control strategies often rely on trajectory tracking of mobile robots modeled as unycicle dynamics (Jiang and Nijmeijer, 1997; Panteley et al, 1998). These are most representative for mobile wheeled robots, but can also be used to study UAVs or Autonomous Underwater Vehicles (AUVs).

In this chapter, we propose a solution to multi-robot formation realization for a fleet of non-holonomic mobile robots under communication constraints. To introduce and motivate the network structure, consider a group of mobile robots that communicates using bluetooth technology. While capable of reliably transmitting information, bluetooth often imposes restrictions on the number of connected robots. Therefore, in practice it is difficult to connect simultaneously a large number of robots, and one must instead use consider smaller networks, by partitioning the overall team into smaller clusters. To ensure the realization of a global, common goal despite this partitioning, we propose a solution inspired from Bragagnolo et al (2014), as follows. Robots are assumed to interact continuously (or very often) only with the robots inside their own cluster. However, sporadically one of the robots in each cluster, that we call leader, will interact with the leaders of the other clusters. This is done using consensus algorithms for linear impulsive systems, which results in desired reference trajectories for the robots. The second component of our approach is a standard control strategy that allows each non-holonomic robot to track this reference. Due to the impulsive nature of the algorithm, the reference trajectories may present discontinuous jumps; however, the real trajectories do not. Overall, by implementing this algorithm the robots are able to follow in a decentralized manner a trajectory that leads to formation realization.

The remainder of the chapter is organized as follows: in Section 13.2 we present the problem statement, and to this end introduce the non-holonomic dynamics of a mobile robot, as well as the network topology of the group of robots. In Section 13.3 we state some preliminaries, mainly the results of Bragagnolo et al (2014) where the convergence to a general consensus is proven, and the results of Panteley et al (1998) that show the tracking controller strategy. In Section 13.4 we present our overall control strategy, which is similar to the one described in Buşoniu and Morărescu (2014). Section 13.5 presents a set of numerical results with two distinct formations (ellipse and a three-leaved clover) and different initial conditions, showing that the formation can be realized despite the complexity of the network and the large number of agents. Finally, in Section 13.6 we present our conclusions and possibilities for future work.

List of symbols and notations

- $\mathbb{N}$: Set of nonnegative integers;
- $\mathbb{R}$: Set of real numbers;
- $\mathbb{R}_+$: Set of nonnegative real numbers;
- $\|x\|$: Euclidean norm of vector x;
- $A^\top$: Transpose of matrix A;
- $A > 0$ ($A \geq 0$): Symmetric matrix A is positive (semi-)definite;
- I_k: The $k \times k$ identity matrix;
- $\mathbf{1}_k$: Column vector of size k having all components equal to 1;
- $\mathbf{0}_k$: Column vector of size k having all components equal to 0;
- (q_i, y_i): Cartesian coordinates of the center of mass of robot i;
- θ_i: Angle between the heading direction and the x-axis of robot i;
- v_i: Linear velocity of robot i;
- ω_i: Angular velocity of robot i;
- (q_{ri}, y_{ri}): Cartesian coordinates of the reference trajectory for robot i;
- θ_{ri}: Angle of the reference trajectory for robot i;
- v_{ri}: Linear velocity of the reference trajectory for robot i;

- ω_{ri}: Angular velocity of the reference trajectory for robot i;
- $\mathcal{G} = (\mathcal{V}, \mathcal{E})$: Graph with vertex set $\mathcal{V}$ and edge set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$
- L: Laplacian matrix;
- P_l: Stochastic matrix (also called Perron matrix);
- $\mathbf{w}^\top L = 0$: $\mathbf{w}$ is the left eigenvector of Laplacian L associated to the eigenvalue 0;
- $u^\top P = u^\top$: u is the left eigenvector of Perron matrix P associated to the eigenvalue 1;
- $A_1 \otimes A_2$: The Kronecker product of matrices A_1 and A_2;

13.2 Problem Formulation and Preliminaries

In this section we introduce the problem of multi-robot formation of robots in clustered networks, and some preliminaries on the robot dynamics and network topology. We start with the overall problem description.

Problem statement: *Consider a network of mobile robots. Due to communication restrictions the network is not overall connected, instead it is separated into several clusters, or smaller networks. The connections inside each cluster happen continuously or very rapidly. Moreover, one robot of each cluster is allowed to communicate outside its cluster at discrete time instants. The goal is to realize an a priori specified formation in a decentralized manner.*

For the sake of clarity we give a simple example that illustrates this type of problem. In Figure 13.1 we consider two clusters, red and blue, of two robots that have to realize a given formation (a square, in this case). The only information that is provided to each robot is the state of its neighbors. In this case each robot accesses only the position and velocity of another robot of the same color. Due to the cluster pattern the robots would never be able to accomplish their task without communication between the clusters. Thus, at some discrete-time instants we allow a communication between one red and one blue robot (tagged in Figure 13.1 as L), that we call leaders in the sequel.

The approach we take separates the formation realization problem in two subproblems: the trajectory tracking of a reference under non-holonomic dynamics; and the consensus approach used to generate this reference. Before we can formally define and solve these subproblems, some preliminaries about the robot dynamics and the network topology need to be presented.

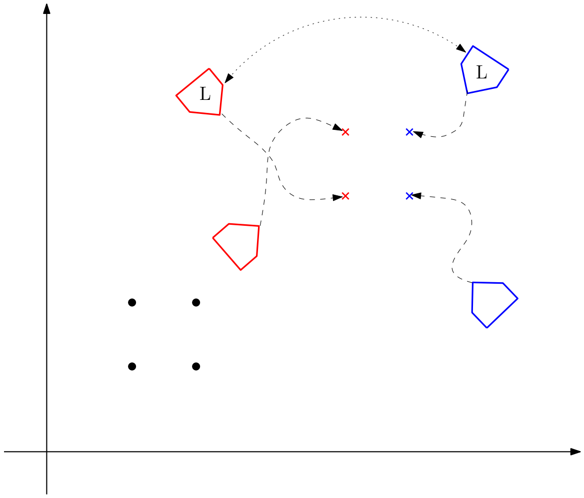

Fig. 13.1: Formation problem. The bottom formation (black dots) shows the intended formation while the top formation (crosses) is the final formation. Agents that can communicate outside the cluster (leaders) are tagged by L, and communication outside the clusters is shown by the dotted line.

13.2.1 Robot Dynamics and Tracking Error

The dynamics of each individual robot are described by the following differential equation

$$\begin{aligned} \dot{q}_i(t) &= v_i \cos(\theta_i), \\ \dot{y}_i(t) &= v_i \sin(\theta_i), \\ \dot{\theta}_i(t) &= \omega_i \end{aligned} \tag{13.1}$$

where v_i is the linear velocity and ω_i is angular velocity of robot i; (q_i, y_i) are the Cartesian coordinates of the center of mass of the robot, and θ_i is the angle between the heading direction and the x-axis (see Figure 13.2).

In the sequel we will have to solve a trajectory tracking problem, so we wish to find control laws for v_i and ω_i such that the robot follows a reference position $p_{ri} = (q_{ri}, y_{ri}, \theta_{ri})$ with velocities v_{ri} and ω_{ri}. We now define the error coordinates from Kanayama et al (1990) (see Figure 13.3).

$$\begin{bmatrix} q_{ei} \\ y_{ei} \\ \theta_{ei} \end{bmatrix} = \begin{bmatrix} \cos(\theta_i) & \sin(\theta_i) & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_{ri} - q_i \\ y_{ri} - y_i \\ \theta_{ri} - \theta_i \end{bmatrix}$$

and the corresponding error dynamics

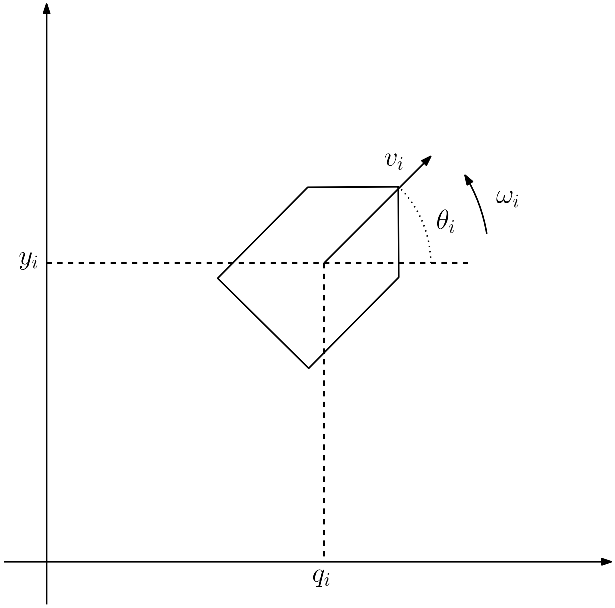

Fig. 13.2: The mobile robot

$$\begin{aligned}\dot{q}_{ei}(t) &= \omega_i y_{ei} - v_i + v_{ri}\cos(\theta_{ei}),\\ \dot{y}_{ei}(t) &= -\omega_i q_{ei} + v_{ri}\sin(\theta_{ei}),\\ \dot{\theta}_{ei}(t) &= \omega_{ri} - \omega_i\end{aligned} \tag{13.2}$$

13.2.2 Network Topology and Agreement Dynamics

As explained earlier we consider that each robot represents an agent in a network. Therefore we provide here some basic concepts related to multi-agent systems and we describe the structure of the network at hand. The communications among agents are specified by a directed graph. A link in this graph represents that an information is transmitted between the vertices situated at its ends. The direction of the link simply states which of the vertices/robots receive the information and which of them send it. The general framework is as follows. We consider a network of n agents described by the digraph (i.e. directed graph) $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where the vertex set $\mathcal{V}$ represents the set of agents and the edge set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ represents the interactions.

In the sequel, we consider that the agent set $\mathcal{V}$ is partitioned in m strongly connected clusters/communities $\mathcal{C}_1, \ldots, \mathcal{C}_m$ and no link between agents belonging to different communities exists. Each community possesses one particular agent called leader and denoted in the following by $l_i \in \mathcal{C}_i,\ \forall i \in \{1, \ldots, m\}$. The set of leaders will be referred to as $\mathcal{L} = \{l_1, \ldots, l_m\}$. The rest of the agents will be called followers

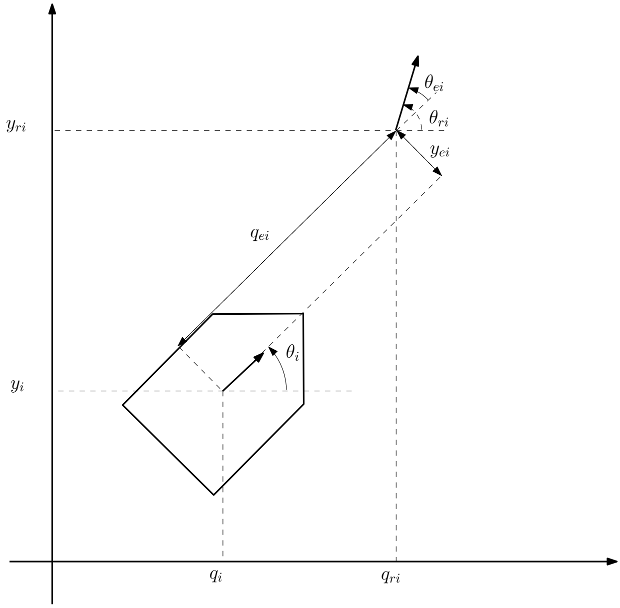

Fig. 13.3: The error dynamics

and denoted by f_j. For the sake of clarity we consider that the leader is the first element of its cluster:

$$\mathcal{C}_i = \{l_i, f_{m_{i-1}+2}, \ldots, f_{m_i}\},\ \forall i \in \{1, \ldots, m\} \tag{13.3}$$

where $m_0 = 0$, $m_m = n$ and the cardinality of $\mathcal{C}_i$ is given by $|\mathcal{C}_i| \triangleq n_i = m_i - m_{i-1}, \forall i \geq 1$. At specific time instants $t_k,\ k \geq 1$, called reset times, the leaders interact between them following a predefined interaction map $\mathcal{E}_l \subset \mathcal{L} \times \mathcal{L}$. We also assume that $\mathcal{G}_l = (\mathcal{L}, \mathcal{E}_l)$ is strongly connected meaning that there exists a directed path (i.e. sequence of directed edges) between any two different leaders. In order to keep the presentation simple, by an abuse of notation each agent will have a scalar state denoted also by l_i for the leader l_i and f_j for the follower f_j. Note that the agent state has a generic meaning in this section. We also introduce the vectors $x = (l_1, f_2, \ldots, f_{m_1}, \ldots, l_m, \ldots, f_{m_m})^\top \in \mathbb{R}^n$ and $x_l = (l_1, l_2, \ldots, l_m)^\top \in \mathbb{R}^m$ collecting all the states of the agents and all the leaders' states, respectively.
As a part of control strategy we have to design the reference trajectories through a linear reset dynamics that agrees with the communication constraints introduced before. In order to ensure that the collaborative control achieves global agreement, the references will be defined by the following overall network dynamics:

$$\begin{cases} \dot{x}(t) = -Lx(t), \quad \forall t \in \mathbb{R}_+ \setminus \mathcal{T} \\ x_l(t_k) = P_l x_l(t_k^-)\ \forall t_k \in \mathcal{T} \\ x(0) = x_0 \end{cases} \tag{13.4}$$

where $\mathcal{T} = \{t_k \in \mathbb{R}_+ \mid t_k < t_{k+1}, \ \forall k \in \mathbb{N}, t_k \text{ reset time}\}$, $L \in \mathbb{R}^{n \times n}$ is a generalized Laplacian matrix associated to the graph $\mathcal{G}$ and $P_l \in \mathbb{R}^{m \times m}$ is a Perron matrix associated to the graph $\mathcal{G}_l = (\mathcal{L}, \mathcal{E}_l)$. More precisely, the entries of L and P_l satisfy the following relations:

$$\begin{cases} L_{(i,j)} = 0, \text{ if } (i,j) \notin \mathcal{E} \\ L_{(i,j)} < 0, \text{ if } (i,j) \in \mathcal{E}, \ i \neq j \\ L_{(i,i)} = -\sum_{j \neq i} L_{i,j}, \forall i = 1, \ldots, n \end{cases}, \tag{13.5}$$

$$\begin{cases} P_{l(i,j)} = 0, \text{ if } (i,j) \notin \mathcal{E}_l \\ P_{l(i,j)} > 0, \text{ if } (i,j) \in \mathcal{E}_l, \ i \neq j \\ \sum_{j=1}^{m} P_{l(i,j)} = 1, \forall i = 1, \ldots, m \end{cases}. \tag{13.6}$$

The values $L_{(i,j)}$ and $P_{l(i,j)}$ represent the weight of the state of the agent j in the updating process of the state of agent i when using the continuous and discrete dynamics, respectively. These values describe the level of democracy inside each community and in the leaders' network. Note also that L has the following block diagonal structure $L = diag(L_1, L_2, ..., L_m)$, $L_i \in \mathbb{R}^{n_i}$ with $L_i \mathbf{1}_{n_i} = \mathbf{0}_{n_i}$ and $P_l \mathbf{1}_m = \mathbf{1}_m$. Due to the strong connectivity of $\mathcal{C}_i, \ i = 1, m$ and $\mathcal{G}_l$, 0 is a simple eigenvalue of each L_i and 1 is a simple eigenvalue of P_l. It is worth pointing out explicitly that, even though (13.4) is stated in a centralized fashion, due to the sparse nature of the L and P_l matrices, only local communication is necessary.

With the background developed so far, we can formalize the subproblems presented in the beginning of the section.

First, to make the link between consensus and formation control, note that a simple change of coordinates allows transforming the problem of realizing a formation into a consensus problem. Let us define the desired formation as a vector of positions $p = (p_1, p_2, ..., p_n)$ where p_i is the position that corresponds to the i^{th} robot. Define also:

$$z(t) = r^*(t) - p, \tag{13.7}$$

where z is the vector that aggregates the distances between the agents' desired positions (r^*) and the formation positions (p). Finally we can present subproblem 2:

Subproblem 1 (Linear consensus): *Consider a network of robots separated in clusters. Given the proposed change of coordinates z, prove that consensus will be reached in z.*

If consensus is achieved in z (not necessarily at 0), then all the robots' references will achieve the formation somewhere in the plane (not necessarily at the locations where the formation is defined). Then, to actually follow the reference trajectories, we need to solve the following trajectory tracking problem:

Subproblem 2 (Tracking for non-holonomic systems): *Consider a mobile robot with non-holonomic dynamics. Given a reference trajectory, find appropriate velocity control laws* v *and* ω *in the form* $v = v(t, q_e, y_e, \theta_e)$ *and* $\omega = \omega(t, q_e, y_e, \theta_e)$ *that follow this reference trajectory.*

13.3 Solving the Consensus and Tracking Problems

In this section we will present two preliminary results from the literature that will be instrumental to our approach. The first, in Section 13.3.1, is related to the consensus problem and comes from Bragagnolo et al (2014). It shows that given the initial conditions and topology of the network, we can find the general consensus value. We then use this result to present sufficient conditions for the stability of our trajectories by means of a Linear Matrix Inequality (LMI). In Section 13.3.2 we present the controller from Panteley et al (1998), which will be useful in tracking the trajectories defined in the consensus phase.

13.3.1 Linear Consensus for Networks with a Cluster Pattern

We start this section by characterizing the possible consensus values of system (13.4) (later, we will show that consensus is indeed achieved). Firstly, we show that each agent tracks a local agreement function that is piecewise constant. Then, we prove that the vector of local agreements lies in a subspace defined by the continuous dynamics and initial condition. Therefore, the consensus value is determined only by the initial condition of the network and by the interconnection structure.

As we have noticed $\mathbf{1}_{n_i}$ is the right eigenvector of L_i associated with the eigenvalue 0 and $\mathbf{1}_m$ is the right eigenvector of P_l associated with the eigenvalue 1. In the sequel, we denote by $\mathbf{w}_i$ the left eigenvector of L_i associated with the the eigenvalue 0 such that $\mathbf{w}_i^\top \mathbf{1}_{n_i} = 1$. Similarly, let $\mu = (\mu_1, \dots, \mu_m)^\top$ be the left eigenvector of P_l associated with the eigenvalue 1 such that $\mu^\top \mathbf{1}_m = 1$. Due to the structure (13.3) of the communities, we emphasize that each vector $\mathbf{w}_i$ can be decomposed in its first component $\mathbf{w}_{i,l}$ and the vector of all other components denoted as $\mathbf{w}_{i,f}$. Let us introduce the matrix of the left eigenvectors of the communities:

$$W = \begin{bmatrix} \mathbf{w}_1^\top & 0 & \cdots & 0 \\ 0 & \mathbf{w}_2^\top & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{w}_m^\top \end{bmatrix} \in \mathbb{R}^{m \times n}.$$

Let us first recall a well known result concerning the consensus in networks of agents with continuous time dynamics (see Olfati-Saber and Murray (2004) for instance).

Lemma 13.1. *Let $\mathcal{G}$ be a strongly connected digraph and L the corresponding Laplacian matrix. Consider a network of agents whose collective dynamics is described by $\dot{x}(t) = -Lx(t)$. Let us also consider $L\mathbf{1} = \mathbf{0}$, $u^\top L = \mathbf{0}$ and $u^\top \mathbf{1} = 1$ where u is the normalized left eigenvector of matrix L. Then, the agents asymptotically reach a consensus and the consensus value is given by $x^* = u^\top x(0)$. Moreover, the vector u defines an invariant subspace for the collective dynamics: $u^\top x(t) = u^\top x(0), \forall t \geq 0$*

Remark 13.1. When dynamics (13.4) is considered, Lemma 13.1 implies that between two reset instants t_k and t_{k+1}, the agents belonging to the same community converge to a local agreement defined by $x_i^*(k) = \mathbf{w}_i^\top x_{\mathcal{C}_i}(t_k)$ where $x_{\mathcal{C}_i}(\cdot)$ is the vector collecting the states of the agents belonging to the cluster $\mathcal{C}_i$. Nevertheless, at the reset times the value of the local agreement can change. Thus,

$$\begin{aligned}&\mathbf{w}_i^\top x(t) = \mathbf{w}_i^\top x_{\mathcal{C}_i}(t_k),\ \forall t \in (t_k, t_{k+1}) \text{ and possibly}\\ &\mathbf{w}_i^\top x_{\mathcal{C}_i}(t) \neq \mathbf{w}_i^\top x_{\mathcal{C}_i}(t_k),\ \text{ for } t \notin (t_k, t_{k+1})\end{aligned}$$

Therefore, the agents whose collective dynamics is described by the hybrid system (13.4), may reach a consensus only if the local agreements converge one to each other.

Before presenting the next result, let us introduce the following vectors:

$$\begin{aligned}x^*(t) &= (x_1^*(t), x_2^*(t), \ldots, x_m^*(t))^\top \in \mathbb{R}^m\\ u &= (\mu_1/\mathbf{w}_{1,l}, \mu_2/\mathbf{w}_{2,l}, \ldots, \mu_m/\mathbf{w}_{m,l})^\top \in \mathbb{R}^m\end{aligned} \tag{13.8}$$

where $x_i^*(t)$ represents the local agreement of the cluster $\mathcal{C}_i$ at instant t. Recall that $\mu \in \mathbb{R}^m$ and $\mathbf{w}_i \in \mathbb{R}^{n_i}$ were defined at the beginning of the section as left eigenvectors associated with the matrices describing the reset dynamics of the leaders and the continuous dynamics of each cluster, respectively. Note that $x^*(t)$ is time-varying but piecewise constant: $x^*(t) = x^*(k)\ \forall t \in (t_k, t_{k+1})$.

Proposition 13.1 (Proposition 3 in Bragagnolo et al (2014)). *Consider the system* (13.4) *with L and P_l defined by* (13.5) *and* (13.6)*, respectively. Then,*

$$u^\top x^*(t) = u^\top x^*(0),\ \forall t \in \mathbb{R}_+. \tag{13.8}$$

Corollary 13.1 (Corollary 1 in Bragagnolo et al (2014)). *Consider the system* (13.4) *with L and P_l defined by* (13.5) *and* (13.6)*, respectively. Assuming the agents of this system reach a consensus, the consensus value is*

$$x^* = \frac{u^\top W x(0)}{\sum_{i=1}^m u_i}. \tag{13.9}$$

In order to simplify the presentation and without loss of generality, in what follows, we consider that $\sum_{i=1}^m u_i = 1$. A trivial result which may be seen as a consequence of Corollary 13.1 is the following.

Corollary 13.2. *If the matrices L, P_l are symmetric (i.e. i^{th} agent takes into account the state of j^{th} agent with the same weight as j^{th} takes into account i^{th} agent) the consensus value is the average of the initial states.*

The stability analysis of the equilibrium point x^* will be given by means of some LMI conditions. Specifically, we recall and adapt some results presented in Hetel et al (2013). Since the consensus value is computed in the previous section we can first define the disagreement vector $\gamma = x - x^*\mathbf{1}_n$. We also introduce an extended stochastic matrix $P_{ex} = T^\top \begin{bmatrix} P_l & 0 \\ 0 & I_{n-m} \end{bmatrix} T$ where T is a permutation matrix allowing to recover the cluster structure of L. Note that $L\mathbf{1}_n = \mathbf{0}_n$ and $P_{ex}\mathbf{1}_n = \mathbf{1}_n$. Thus, the disagreement dynamics is exactly the same as the system one:

$$\begin{cases} \dot{\gamma}(t) = -L\gamma(t), \quad \forall t \in \mathbb{R}_+ \setminus \mathcal{T} \\ \gamma(t_k) = P_{ex}\gamma(t_k^-) \ \forall t_k \in \mathcal{T} \\ \gamma(0) = \gamma_0 \end{cases} . \tag{13.10}$$

Now we have to analyze the stability of the equilibrium point $\gamma^* = \mathbf{0}_n$ for the system (13.10). We note that Theorem 2 in Hetel et al (2013) cannot be directly applied due to the marginal stability of the matrices L and P_{ex}.

The reset sequence is defined such that $t_{k+1} - t_k = \delta + \delta'$ where $\delta \in \mathbb{R}_+$ is fixed and $\delta' \in \Delta$ with $\Delta \subset \mathbb{R}_+$ a compact set. Thus the set of reset times $\mathcal{T}$ belongs to the set $\Phi(\Delta) \triangleq \left\{ \{t_k\}_{k\in\mathbb{N}}, t_{k+1} - t_k = \delta + \delta'_k, \delta'_k \in \Delta, \forall k \in \mathbb{N} \right\}$ of all admissible reset sequences.

Definition 13.1. *We say that the equilibrium $\gamma^* = \mathbf{0}_n$ of the system* (13.10) *is Globally Uniformly Exponentially Stable (GUES) with respect to the set of reset sequences $\Phi(\Delta)$ if there exist positive scalars c, λ such that for any $\mathcal{T} \in \Phi(\Delta)$, any $\gamma_0 \in \mathbb{R}^n$, and any $t \geq 0$*

$$\|\varphi(t, \gamma_0)\| \leq ce^{-\lambda t}\|\gamma_0\| \tag{13.11}$$

The following theorem is instrumental:

Theorem 13.1 (Theorem 1 in Hetel et al (2013)). *Consider the system* (13.10) *with the set of reset times $\mathcal{T} \in \Phi(\Delta)$. The equilibrium $\gamma^* = \mathbf{0}_n$ is GUES if and only if there exists a positive function $V : \mathbb{R}^n \mapsto \mathbb{R}_+$ strictly convex,*

$$V(\gamma) = \gamma^\top S_{[\gamma]}\gamma,$$

homogeneous (of second order), such that $V(0) = 0$ and $V(\gamma(t_k)) > V(\gamma(t_{k+1}))$ for all $\gamma(t_k) \neq 0$, $k \in \mathbb{N}$ and any of the possible reset sequences $\mathcal{T} \in \Phi(\Delta)$. Here, $S_{[\cdot]} : \mathbb{R}^n \mapsto \mathbb{R}^{n\times n}$, $S_{[\gamma]} = S_{[\gamma]}^\top = S_{[a\gamma]} > 0$, $\forall x \neq 0, a \in \mathbb{R}$, $a \neq 0$.

The previous result allows reducing the stability analysis of the linear reset system (13.10) to the stability of the discrete dynamics related to the reset instants.

As shown in Bragagnolo et al (2014) a Lyapunov function V for (13.10) satisfying Theorem 13.1 can be obtained by solving a parametric LMI:

Theorem 13.2 (Theorem 6 in Bragagnolo et al (2014)). *Consider the system* (13.10) *with* $\mathcal{T}$ *in the admissible reset sequences* $\Phi(\Delta)$. *If there exist matrices* $S(\delta')$, $S(\cdot):\Delta \mapsto \mathbb{R}^{n\times n}$ *continuous with respect to* δ', $S(\delta') = S^\top(\delta') > 0$, $\delta' \in \Delta$ *such that the LMI*

$$\begin{aligned}
&\left(I_n - \mathbf{1}_n u^\top W\right)^\top S(\delta_a)\left(I_n - \mathbf{1}_n u^\top W\right) - \\
&\left(Y(\delta_a) - \mathbf{1}_n u^\top W\right)^\top S(\delta_b)\left(Y(\delta_a) - \mathbf{1}_n u^\top W\right) > 0, \\
&\textit{where } Y(\delta_a) \triangleq P_{ex} e^{-L(\delta+\delta_a)}
\end{aligned}$$

is satisfied on $span\{\mathbf{1}_n\}^\perp$ *for all* $\delta_a, \delta_b \in \Delta$, *then* x^* *is globally uniformly exponentially stable for* (13.10). *Moreover, the stability is characterized by the quasi-quadratic Lyapunov function* $V(t) = \max_{\delta'\in\Delta}(x(t)-x^*\mathbf{1}_n)^\top S(\delta')(x(t)-x^*\mathbf{1}_n)$ *satisfying* $V(t_k) > V(t_{k+1})$.

So far, we have established with Theorem 13.2 that the references achieve consensus. Next, we provide tools for solving the problem of reference tracking for the nonholonomic robots.

13.3.2 Tracking for Nonholonomic Systems

In this section we present the controller from Panteley et al (1998). Denote the reference signals for robot i by $(q_{ri}, y_{ri}, v_{ri}, \omega_{ri}, \theta_{ri})$. Recall the error dynamics defined in (13.2):

$$\begin{aligned}
\dot{q}_{ei}(t) &= \omega_i y_{ei} - v_i + v_{ri}\cos(\theta_{ei}), \\
\dot{y}_{ei}(t) &= -\omega_i q_{ei} + v_{ri}\sin(\theta_{ei}), \\
\dot{\theta}_{ei}(t) &= \omega_{ri} - \omega_i
\end{aligned}$$

We can stabilize the mobile robot's orientation, via the linear system $\dot{\theta}_{ei}(t) = \omega_{ri} - \omega_i$, by using the linear controller

$$\omega_i = \omega_{ri} + c_{1i}\theta_{ei} \tag{13.12}$$

which yields GUES for θ_{ei}, provided $c_1 > 0$. If we now set $\theta_{ei} = 0$ in (13.2) we obtain

$$\begin{aligned}
\dot{q}_{ei}(t) &= \omega_i y_{ei} - v_i + v_{ri} \\
\dot{y}_{ei}(t) &= -\omega_i q_{ei}
\end{aligned} \tag{13.13}$$

Concerning the position of the robot, if we choose the linear controller

$$v_i = v_{ri} + c_{2i} q_{ei} \tag{13.14}$$

where $c_2 > 0$, we obtain for the closed-loop system (13.2):

$$\begin{bmatrix} \dot{q}_{ei} \\ \dot{y}_{ei} \end{bmatrix} = \begin{bmatrix} -c_{2i} & \omega_{ri}(t) \\ -\omega_{ri}(t) & 0 \end{bmatrix} \begin{bmatrix} q_{ei} \\ y_{ei} \end{bmatrix}$$

which under appropriate conditions on $\omega_{ri}(t)$ is asymptotically stable.

Remark 13.2. In Panteley et al (1998) the result is made rigorous by Proposition 11. Considering the system (13.2) in closed loop with the controller (13.12,13.14), Proposition 11 states that the closed loop system is globally exponentially stable if $\omega_{ri}(t)$, $\dot{\omega}_{ri}(t)$ and $v_{ri}(t)$ are bounded.

Remark 13.3. The tracking controller presented in this section provides an exponential decrease of the error as long as the reference is continuous. This is the case for almost all the robots in the network. However, the leaders have discontinuous references (see (13.10)), so combining the tracking controller in Panteley et al (1998) with the references design defined in Bragagnolo et al (2014) requires a supplementary condition that is provided in the next section.

13.4 Overall Controller Design

We are ready to define the overall control strategy. As in Buşoniu and Morărescu (2014), and as described above in Section 13.2, the main idea is to track the reference trajectories defined by the consensus strategy. It is important to note that only local information is needed for the reference design, and once the reference is available, the tracking problem is completely decentralized since no additional information from the neighbors is needed.

Consider the directed graph defined in (13.4), with the nonlinear dynamics defined by (13.1). As we are interested in realizing a formation in the 2D plane, we will collect information only about the position (q_i, y_i) of each agent, and this will be the consensus state. As an example, for the leader of the first cluster we have $l_1 = (q_1, y_1)$ and the next agent of the same cluster has $f_2 = (q_2, y_2)$. Note that the consensus algorithms above were given for scalar states; for vector states such as here, it suffices to update in parallel all the state variables with the same formula. Define also the collective state $x = (l_1^\top, f_2^\top, \dots, f_{m_1}^\top, \dots, l_m^\top, \dots, f_{m_m}^\top)^\top \in \mathbb{R}^{nm}$, containing the consensus states of all agents. Consensus will be sought on the distances z between the robot states and the imposed formation. Thus, finally, the linear reset system describing the overall network dynamics is:

$$\begin{cases} \dot{z}(t) = -(L \otimes I_2) z(t), \quad \forall t \in \mathbb{R}_+ \setminus \mathcal{T} \\ z_l(t_k) = (P_l \otimes I_2) z_l(t_k^-) \ \forall t_k \in \mathcal{T} \\ z(0) = z_0 \end{cases} \tag{13.15}$$

From $z(t)$, the references for the robots must be obtained. First, since $z(t) = r^*(t) - p$ from (13.7), the collective reference position $r^*(t)$ can be computed as $z(t) + p$. The reference position of robot i can be extracted from the appropriate position on this overall collective reference vector. We denote the reference position for robot i by $r_i^*(t) = (q_{ri}, y_{ri})^\top$. To apply the local tracking controller, additional information is necessary including the linear velocity v_{ri}, the angle θ_{ri}, and the angular velocity ω_{ri}, and we explain next how to find these quantities.

We have $\dot{z}(t) = \dot{r}^*(t)$ since the vector p is constant, so the derivative of $r_i^*(t)$ gives us the linear velocities on the x and y axes, $\dot{r}_i^*(t) = (\dot{q}_{ri}, \dot{y}_{ri})^\top$. By taking advantage of the consensus dynamics in (13.4), this derivative can be found as:

$$\dot{r}^*(t) = \dot{z}(t) = -(L \otimes I_2) z(t) \,,$$

Then, we obtain the linear velocity by:

$$v_{ri} = \sqrt{\dot{q}_{ri}^2 + \dot{y}_{ri}^2}.$$

By taking into account the velocities on both axes we can now find the orientation of our reference trajectory (θ_{ri}),

$$\theta_{ri} = \arctan \frac{\dot{y}_{ri}}{\dot{q}_{ri}}, \tag{13.16}$$

and by differentiating this we directly get the reference angular velocity ω_{ri}

$$\omega_{ri} = \frac{\partial \theta_{ri}}{\partial t}.$$

In practice, instead of the continuous dynamics we implement a discretization of the Laplacian matrix, which is a more realistic model. We take a sampling time $\tau = \frac{\delta_{max}}{K}$, where K is an integer constant and δ_{max} is the maximum reset time in the admissible reset sequences $\Phi(\Delta)$. Denote then the discrete time instants by τ_κ, with integer κ and $\tau_{\kappa+1} - \tau_\kappa = \tau$. Then, the stochastic counterpart $P = e^{-L\tau}$ of the Laplacian matrix is computed. The network topology remains the same, and the dynamics (13.15) change to:

$$\begin{cases} \hat{z}(\tau_{\kappa+1}) = (P \otimes I_2) z(\tau_\kappa), \\ z(t_k) = (P_{ex} \otimes I_2) z(t_k^-) \quad \forall t_k \in \mathcal{T} \\ z(0) = z_0 \end{cases} \tag{13.17}$$

Here, $\hat{z}(\tau_{\kappa+1})$ is no longer the next step in the consensus but it is instead a target consensus. It is left implicit that, if the resets occur in-between sampling instants, the new references become active at the next sampling instant. By using the equation (13.7), (13.17) leads us to a target position $\hat{r}_i^*(\tau_\kappa) = (\hat{q}_{ri}, \hat{y}_{ri})^\top$ as shown in Figure 13.4.

Another practical consideration is that due to physical constraints, the robots might not be able to follow the trajectories defined by the consensus algorithm,

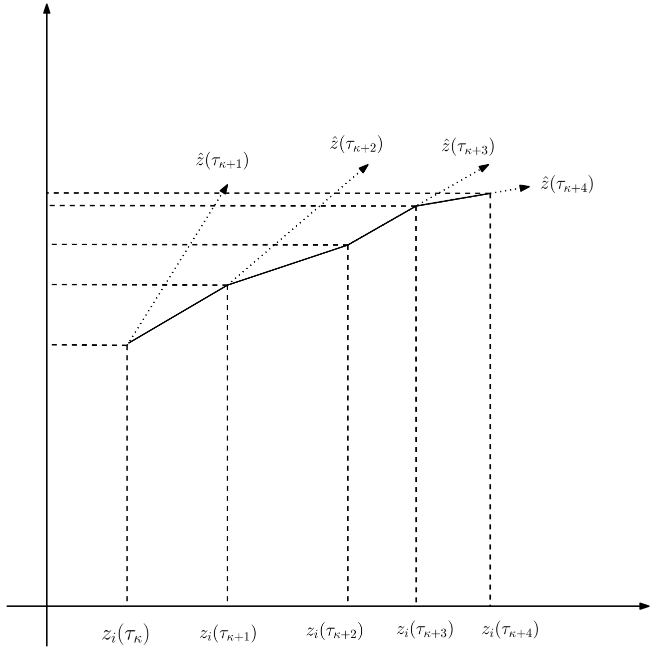

Fig. 13.4: The target value and the actual trajectory

when the speed of convergence of the Laplacian matrix leads to faster trajectories than what the robot can implement. We solve this by considering a saturation of the reference linear velocity and of the angle. With this change, the final formulas for the reference velocity, angle, and angular velocity are:

$$v_{ri}(\tau_\kappa) = \begin{cases} \sqrt{(\hat{q}_{ri}(\tau_{\kappa+1}) - q_{ri}(\tau_\kappa))^2 + (\hat{y}_{ri}(\tau_{\kappa+1}) - y_{ri}(\tau_\kappa))^2} & \text{if} < v_{max} \\ v_{max} & \text{otherwise} \end{cases}$$

$$\theta_{ri}(\tau_\kappa) = \begin{cases} \arctan \frac{(\hat{y}_{ri}(\tau_{\kappa+1}) - y_{ri}(\tau_\kappa))}{(\hat{q}_{ri}(\tau_{\kappa+1}) - q_{ri}(\tau_\kappa))} & \text{if} < \theta_{max} \\ \theta_{max} & \text{otherwise} \end{cases}$$

$$\omega_{ri}(\tau_\kappa) = \frac{\theta_{ri}(\tau_\kappa) - \theta_{ri}(\tau_{\kappa-1})}{\tau}$$

A crucial property of the overall algorithm is that, in-between the resets, the distances to the references trajectories will be reduced. This happens because the tracking controller ensures exponential stability and is allowed to run for a time $t_{k+1} - t_k$ which is always finite. Furthermore, the initial tracking errors will get smaller and smaller after each reset, due to the convergence properties of the consensus algorithm. Thus, we expect that the fleet of non-holonomic robots will reach any given formation with arbitrary precision.

The property of error reduction in-between the resets is similar to the controllability requirement in Buşoniu and Morărescu (2014), which assumed that each robot can reach any reference in a given range with at most K consecutive actions. In our case, the constant K in the time discretization of the consensus dynamics can be seen as playing a similar role to K in Buşoniu and Morărescu (2014).

13.5 Simulation Results

We first validate our approach in a small-scale example with 5 robots. Afterwards, a larger-scale example with 15 robots is shown. In all these simulations we consider the reset time of the leaders to be periodic, i.e. $t_{k+1} - t_k = \delta$.

13.5.1 Small-scale Example: Ellipse Formation

Consider a network of 5 robots partitioned in 2 clusters ($n_1 = 3, n_2 = 2$). The topology of the clusters and the leader interconnections is described by:

$$L = \begin{bmatrix} 0.04 & -0.02 & -0.02 & 0 & 0 \\ -0.01 & 0.01 & 0 & 0 & 0 \\ 0 & -0.02 & 0.02 & 0 & 0 \\ 0 & 0 & 0 & 0.03 & -0.03 \\ 0 & 0 & 0 & -0.01 & 0.01 \end{bmatrix}, \; P_l = \begin{bmatrix} 0.45 & 0.55 \\ 0.25 & 0.75 \end{bmatrix}$$

The reset sequence is given by the reset time $\delta = 5s$ with $\tau = 0.05s$. The initial positions of the robots are, on the x axis, $q = (4.0985, 4.1130, 0.2262, 3.9355, 1.2436)$ on the y axis, $y = (1.7976, 2.5639, 3.9691, 2.1607, 4.2717)$ and the initial angles are $\theta = (1.0463, -2.0224, -2.3373, 3.1358, -2.0664)$. The formation was defined as an ellipse, where the robots are equally spaced by an angle of $2\pi/n$. The positions defined in the formation are $p_q = (1.0000, 1.9511, 1.5878, 0.4122, 0.0489)$ and $p_y = (0.5000, 0.8455, 1.4045, 1.4045, 0.8455)$. The controller parameters were defined as $c_{1i} = 0.7, c_{2i} = 0.5,\ \forall i$. In Figure 13.5 the trajectories of the robots are shown, with the intended formation shown as stars and the final position of the robots as circles.

Figure 13.6 shows, for a different experiment, the positions of each robot at the reset, as well as the initial conditions and the intended formation. The only change from the experiment above was made to the reset schedule, with $\delta = 100s$ and $\tau = 0.1s$. These changes were made such that the convergence to the partial formation (local consensus at each cluster) could be ensured prior to the reset, as shown by the dashed ellipses. The red ellipse shows all the robots of cluster 1 in a partial formation at the first reset while the blue dashed ellipse shows the same for cluster

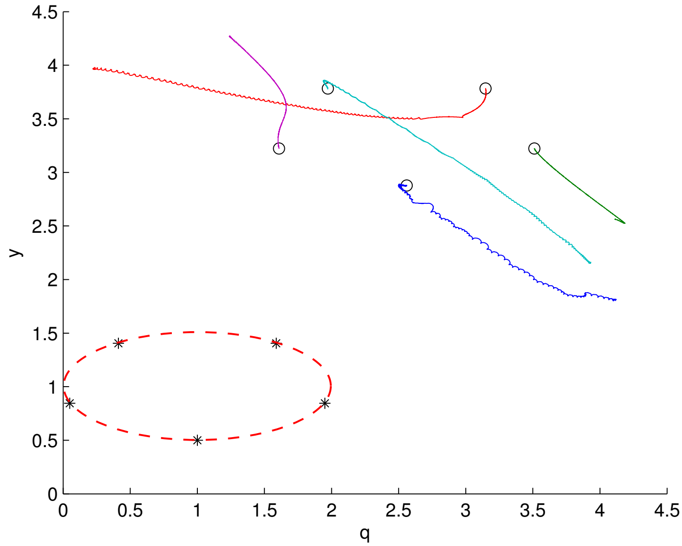

Fig. 13.5: Trajectories of the robots with their final position as circles and intended formation as stars.

2. The initial positions of the robots are represented by diamonds colored red or blue to represent cluster membership.

We conclude from these results that the algorithm is able to achieve the formation; and that partial consensus can be achieved at each reset, given that a large enough interval between resets is provided. In fact, in Figure 13.6 we can see the trajectory not as robots individually going to the formation, but as partial formations getting close to each other.

13.5.2 Large-scale Example: Three-leaf Clover Formation

In the next example we consider a larger system of 15 robots equally separated into 3 clusters ($n_1 = 5, n_2 = 5, n_3 = 5$). The inter-leader communication is driven by:

$$P_l = \begin{bmatrix} 0.1160 & 0.4650 & 0.4190 \\ 0.1360 & 0.5540 & 0.3100 \\ 0.2850 & 0.3100 & 0.4050 \end{bmatrix}$$

while the matrix L is randomized. The reset sequence is given by $\delta = 5s$ with $\tau = 0.05s$. The initial positions and orientations of the robots are also randomized. The controller parameters c_{1i} and c_{2i} are the same as those used in the five-robot

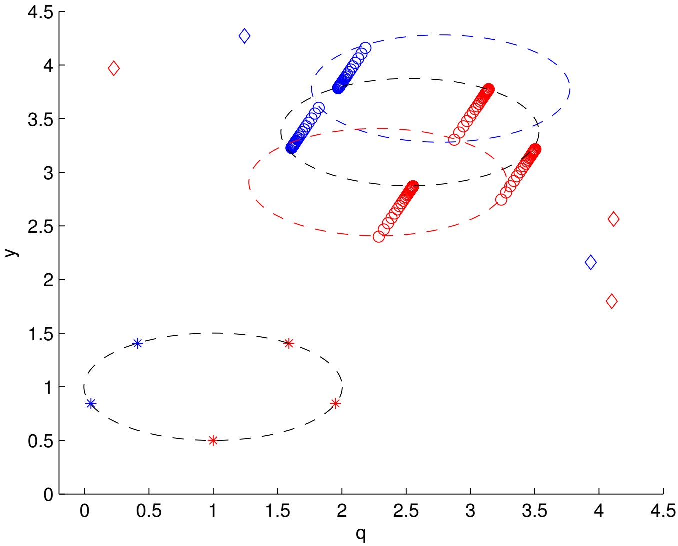

Fig. 13.6: A second experiment: The circles represent the local consensus point at each reset; the diamonds are the initial positions and stars the intended formation. Each cluster is color coded with red and blue.

example. The formation considered this time is a three-leaf clover, represented by the the following parametric equations:

$$q = q_c + a\cos(3t)\cos(t)$$
$$y = y_c + a\cos(3t)\sin(t)$$

where $(q_c, y_c) = (1,1)$ is the center of the formation and $a = 1$ the length of the leaves, with the robots equally spaced from each other by an angle of $2\pi/n$. In Figure 13.7 the trajectories show the convergence to the three-leaved clover formation.

Similarly to the first example, we run a new experiment with more widely spaced resets, $\delta = 200s$ and $\tau = 0.1s$, so as to better present the partial formations during the resets. Figure 13.8 shows the positions of the agents during the first and the last reset. The intended three-leaf clover formation is depicted at the bottom left, while the center of the figure shows the final formation as achieved in the 2D plane. Each cluster is color coded, where the first cluster is red, second is blue and third is green.

These results showcase the fact that the algorithm works despite the larger number of robots or the complexity of the formation. In Figure 13.8, two of the three clusters have already reached their partial formation at the first reset, while one cluster has not completely reached local consensus yet.

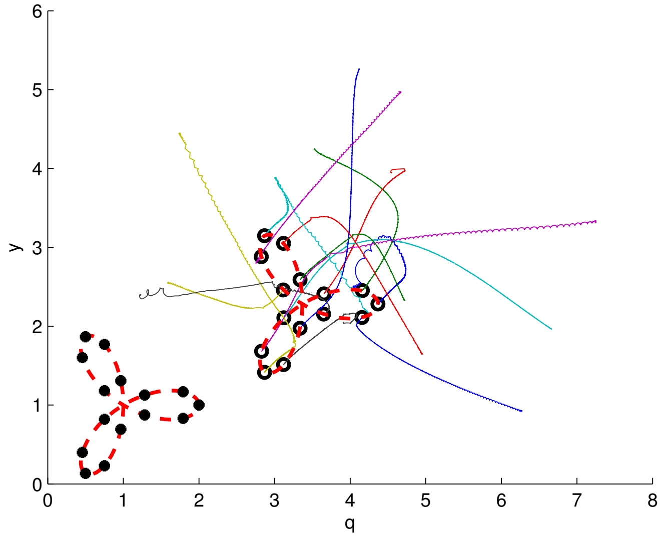

Fig. 13.7: Robots in a three-leaved clover formation.

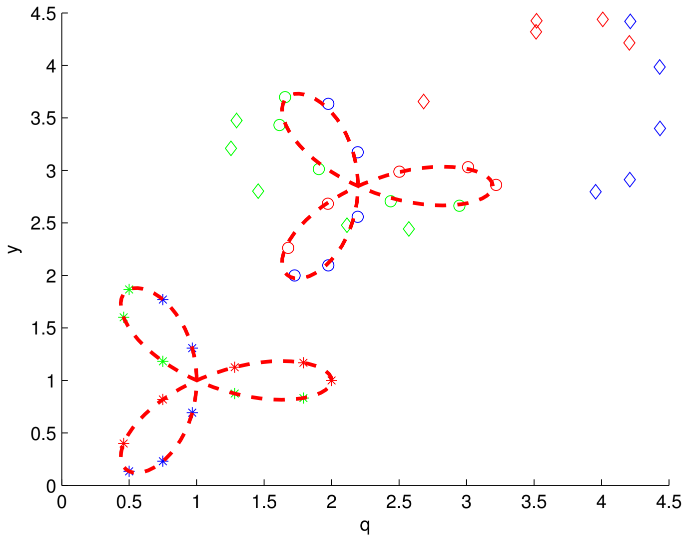

Fig. 13.8: Robots in a three-leaved clover formation, difference between first and last reset.

13.6 Conclusions and Perspectives

In this work we have considered formation control for a fleet of mobile robots separated into clusters that only communicate locally. We presented a consensus algorithm that provides reference trajectories for the robots in a decentralized manner. By taking one of the robots in each cluster as a leader, which will sporadically reset its trajectory by communicating with the leaders of the other clusters, it is possible to lead the partial formations into the desired final formation. A linear tracking controller for non-holonomic robots is then used to track the reference trajectories, thereby obtaining the overall approach. Two academic examples were presented showing both the convergence to the final formation as well as the partial formations at each reset time. Future work may consider the design of the network topology to ensure the realization of the final formation within a predefined space.

Acknowledgments

This work was supported by a Programme Hubert Curien-Brancusi cooperation grant (CNCS-UEFISCDI contract no. 781/2014 and Campus France grant no. 32610SE) and by the PICS project No 6614 "Artificial-Intelligence-Based Optimization for the Control of Networked and Hybrid Systems". Additionally, the work of L. Buşoniu was supported by the Romanian National Authority for Scientific Research, CNCS-UEFISCDI (No. PNII-RU-TE-2012-3-0040). The work of I.-C. Morărescu was partially funded by the National Research Agency (ANR) project "Computation Aware Control Systems" (No. ANR-13-BS03-004-02).

References

Anta A, Tabuada P (2010) To sample or not to sample: self-triggered control for nonlinear systems. IEEE Transactions on Automatic Control 55(9):2030–2042

Beard R, Stepanyan V (2003) Information consensus in distributed multiple vehicle coordinated control. In: IEEE Conference on Decision and Control, vol 2, pp 2029–2034

Bertuccelli LF, Choi HL, Cho P, How JP (2009) Real-time multi-UAV task assignment in dynamic and uncertain environments. In: AIAA Guidance, Navigation, and Control Conference

Bragagnolo MC, Morărescu IC, Daafouz J, Riedinger P (2014) LMI sufficient conditions for the consensus of linear agents with nearly-periodic resets. In: American Control Conference, pp 2575–2580

Brinon Arranz L, Seuret A, Canudas de Wit C (2014) Cooperative control design for time-varying formations of multi-agent systems. IEEE Transactions on Automatic Control 59(8):2283–2288

Brogliato B, Lozano R, Maschke B, Egeland O (2007) Dissipative Systems Analysis and Control. Theory and Applications, 2nd edn. Springer CCES, London

Buşoniu L, Morărescu IC (2014) Consensus for black-box nonlinear agents using optimistic optimization. Automatica 50(4):1201–1208

Bullo F, Cortés J, Martinez S (2009) Distributed Control of Robotic Networks. A Mathematical Approach to Motion Coordination Algorithms. Princeton University Press

Ding X, Rahmani A, Egerstedt M (2009) Optimal multi-UAV convoy protection,. In: ROBOCOMM ’09. Second International Conference on Robot Communication and Coordination

Fiacchini M, Morărescu IC (2014) Convex conditions on decentralized control for graph topology preservation. IEEE Transactions on Automatic Control 59(6):1640–1645

Halgamuge MN, Guru SM, Jennings A (2003) Energy efficient cluster formation in wireless sensor networks. In: 10th International Conference on Telecommunications, 2003.

Heemels W, Johansson K, Tabuada P (2012) An introduction to event-triggered and self-triggered control. In: IEEE Conference on Decision and Control

Hetel L, Daafouz J, Tarbouriech S, Prieur C (2013) Stabilization of linear impulsive systems through a nearly-periodic reset. Nonlinear Analysis: Hybrid Systems 7:4–15

Jadbabaie A, Lin J, Morse AS (2003) Coordination of groups of mobile autonomous agents using nearest neighbor rules. IEEE Transactions on Automatic Control 48(6):988–1001

Jiang ZP, Nijmeijer H (1997) Tracking control of mobile robots: a case study in backstepping. Automatica 33(7):1393–1399

Kanayama Y, Kimura Y, Miyazaki F, Noguchi T (1990) A stable tracking control method for an autonomous mobile robot. In: Proceedings of 1990 IEEE International Conference on Robotics and Automation

Kolmanovsky H, McClamroch N (1995) Developments in nonholonomic control systems. IEEE Control Systems Magazine 15(6):20–36

Lin X, Cassandras C (2014) Trajectory optimization for multi-agent persistent monitoring in two-dimensional spaces. In: IEEE Conference on Decision and Control

Mahacek P, Kitts C, Mas I (2011) Dynamic guarding of marine assets through cluster control of automated surface vessel fleets. IEEE/ASME Transactions on Mechatronics 17(1):65–75

Martin S, Girard A (2013) Continuous-time consensus under persistent connectivity and slow divergence of reciprocal interaction weights. SIAM Journal on Control and Optimization 51(3):2568–2584

McClamroch N, Wang D (1988) Feedback stabilization and tracking of constrained robots. IEEE Transactions on Automatic Control 33:419–426

Michiels W, Morarescu IC, Niculescu SI (2009) Consensus problems with distributed delays, with application to traffic flow models. SIAM Journal on Control and Optimization 48(1):77–101

Moreau JJ (1988) Unilateral contact and dry friction in finite freedom dynamics, Nonsmooth Mechanics and Applications, CISM Courses and Lectures., vol 302. New York: Springer-Verlag

Moreau L (2005) Stability of multiagent systems with time-dependent communication links. IEEE Transactions on Automatic Control 50(2):169–182

Morărescu IC, Brogliato B (2010a) Passivity-based switching control of flexible-joint complementarity mechanical systems. Automatica 46(1):160–166

Morărescu IC, Brogliato B (2010b) Passivity-based tracking control of multiconstraint complementarity lagrangian systems. IEEE Transactions on Automatic Control 55(6):1300–1310

Olfati-Saber R, Murray R (2004) Consensus problems in networks of agents with switching topology and time-delays. IEEE Transactions on Automatic Control 49:1520–1533

Panteley E, Lefeber E, Loria A, Nijmeijer H (1998) Exponential tracking control of a mobile car using a cascaded approach. In: Proceedings of the IFAC workshop on motion control

Postoyan R, Tabuada P, Nesic D, Anta A (2011) Event-triggered and self-triggered stabilization of distributed networked control systems. In: Joint IEEE Conference on Decision and Control and European Control Conference, pp 2565–2570

Ratliff LJ, Dong R, Ohlsson H, Cárdenas AA, Sastry SS (2014) Privacy and customer segmentation in the smart grid. In: IEEE Conference on Decision and Control

Ren W, Beard R (2004) Formation feedback control for multiple spacecraft via virtual structures. IEEE Proceedings of Control Theory and Applications 151:357–368

Samad T, Bay J, Godbole D (2007) Network-centric systems for military operations in urban terrain: The role of UAVs. Proceedings of the IEEE 95(1):92–107

Samson C, Ait-Abderrahim K (1991) Feedback control of a nonholonomic wheeled cart in cartesian space. In: in Proc. IEEE International Conference on Robotics and Automation, pp 1136–1141

Sariel S, Balch T, Stack J (2006) Distributed multi-AUV coordination in naval mine countermeasure missions. Tech. rep., Georgia Institute of Technology

Scharf DP, Hadaegh FY, Ploen SR (2003) A survey of spacecraft formation flying guidance and control (part i): Guidance. In: American Control Conference, vol 2, pp 1733–1739

Scharf DP, Hadaegh FY, Ploen SR (2004) A survey of spacecraft formation flying guidance and control (part ii): Control. In: American Control Conference, IEEE, vol 4, pp 2976–2985

Su H, Chen G, Wang X, Lin Z (2011) Adaptive second-order consensus of networked mobile agents with nonlinear dynamics. Automatica 47(2):368–375

Sun K, Peng P, Ning P, Wang C (2006) Secure distributed cluster formation in wireless sensor networks. In: 22nd Annual Computer Security Applications Conference.

Tanner H, Jadbabaie A, Pappas G (2007) Flocking in fixed and switching networks. IEEE Transactions on Automatic Control 52(5):863–867