



HAL
open science

Flocking of Multiple Unmanned Aerial Vehicles by LQR Control

Osamah Saif, Isabelle Fantoni, Arturo Zavala-Río

► **To cite this version:**

Osamah Saif, Isabelle Fantoni, Arturo Zavala-Río. Flocking of Multiple Unmanned Aerial Vehicles by LQR Control. International Conference on Unmanned Aircraft Systems (ICUAS 2014), May 2014, Orlando, FL, United States. pp.222-228. hal-01026318

HAL Id: hal-01026318

<https://hal.science/hal-01026318>

Submitted on 21 Jul 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Flocking of Multiple Unmanned Aerial Vehicles by LQR Control

Osamah Saif¹ and Isabelle Fantoni¹ and Arturo Zavala-Río²

Abstract—In this paper, we address the control problem of multiple Unmanned Aerial Vehicles (UAVs) flocking by using a behavior-based strategy. We conceive a behavior intending to address the control design towards a successful achievement of the flocking task without fragmentation. Moreover, through its implementation in UAVs, no rendezvous point is needed to perform flocking. We design a control law, which is independent of the number of UAVs in the flock. We use the LQR control method to develop our strategy. Our proposed strategy deals with the flocking problem from a measurement-mapping perspective. Simulation results show two scenarios of aggregation and navigation of multiple UAVs.

I. INTRODUCTION

Flocking is a collective, harmonic and collision-free motion of animals. Biologists have studied flocks of birds, swarms of insects, herds of quadruped, and schools of fish, in order to understand the secret of this collective motion. Moreover, they have searched to discover motivations that lead animals to aggregate in groups [1] [2].

The amazing phenomenon of flocking has attracted control and robotics scientists, so they have tried to imitate it on robots platoons. One of the most advanced fields that developed structures similar to flocking is flight formation control. Three of them are cited here: *leader-follower*, *virtual leader* and *behavior-based control*. In the leader-follower structure, individuals in the formation follow one UAV (Unmanned Aerial Vehicle), which is designated as a leader. A formation flight mission trajectory is loaded in the leader, and the followers track their leader. This structure is simple and widely implemented in multi-UAVs formation [3], [4]. However, since the entire formation depends on one UAV, the whole formation will be affected by a faulty leader.

The virtual leader structure consists of replacing a formation leader, in the leader-follower structure, by a virtual one. All the UAVs in the formation receive the mission trajectory. This trajectory is the virtual leader itself. One of the main drawbacks of this structure is that the possibility of collision between UAVs increases, since there is no feedback to the formation. Moreover, the predefined trajectory of the mission decreases the autonomy of the formation [4].

In the behavior-based control, each UAV follows some rules to achieve the formation. In fact, this structure is inspired from the collective motion of animals. Among the first technical works on this structure, we found the distributed

behavioral model by *Reynolds* [5]. Although Reynolds was specialized in computer graphics, his work inspired researchers in control theory and robotics that applied the rules of Reynolds in a theoretical and experimental framework [6], [7]. Reynolds inspired his rules from biologists study of animals collective motion. He considered that each individual in a flock should follow rules in order to perform the flocking behavior. These rules are: 1) Collision Avoidance; 2) Velocity Matching, and 3) Flock Centering.

Olfati-Saber in [6] presented a concrete theoretical framework for control design and stability analysis of multi-agent flocking of dynamic systems. The author proved that his first proposed algorithm embodies the rules of Reynolds. He used a collective potential function to design the flock centering behavior. Moreover, a velocity consensus term was added to the potential term to achieve the velocity matching behavior. Other algorithms were proposed for navigation and obstacle avoidance. The stability of the proposed algorithms was analyzed by a monotonically decreasing structural function and LaSalle's invariance principle. One of the limitations of the proposed algorithms is their dependence on the connectivity degree between UAVs in a flock. Moreover, algorithm 1 (without navigation) dealt only with a local flocking problem, *i.e.*, he supposed that individuals are already in a flock. A global problem is to consider, beside the local problem, the trend to form a flock from scattered individuals or small flocks.

In [7], *G. Antonelli, et al* followed the rules of Reynolds and designed behavior-based control laws for multi-robot systems flocking. Null-Space-based behavioral control laws were used to follow the rules of Reynolds with the same flocking local problem and connectivity degree dependence.

In this paper, we use the behavior-based control structure to achieve a multi-UAVs flocking. We conceive a behavior intending to address the control design towards a successful achievement of the flocking task without fragmentation. The proposed behavior treats the flocking problem from a global perspective, that is, we include a tendency of separated UAVs to form a flock. We propose a new control strategy to achieve the rules of Reynolds and the new proposed behavior. This strategy is based on the LQR control. Moreover, this strategy allows us to design an LQR controller, for each UAV, which is independent of the number of UAVs within a flock. In fact, we consider the flocking as a problem of trajectory (or reference) generation, rather than an issue of control design. The desired trajectory for each UAV is generated by using the measured states of the UAVs in its field of view. Furthermore, we expand our control strategy to perform a navigation behavior. In this paper, we suppose that UAVs have a simple

¹O. Saif and I. Fantoni are with the University of Technology of Compiègne (UTC) – CNRS, UMR 7253 Heudiasyc, Royallieu, CS 60319, 60200 Compiègne, France. {osamah.saif/isabelle.fantoni}@hds.utc.fr

²A. Zavala-Río is with Instituto Potosino de Investigación Científica y Tecnológica, Camino a la Presa San José 2055, Lomas 4a. Sección 78216, San Luis Potosí, Mexico. azavala@ipicyt.edu.mx

linear dynamics (double integrator), in a similar manner as the work in [6]. Nonlinear models will be treated in an upcoming sequel of this paper.

The outline of this paper is: section II addresses, firstly, preliminaries on the representation of multi-UAVs systems by graph theory, on the formulation of the problem in the Euclidean framework, on the modeling of multi-UAVs system, and on the LQR control. Then, sections III and IV present the main contributions and the results of simulations. Finally, section V concludes by making remarks and presenting future works.

II. PRELIMINARIES

In this section, we introduce the basic elements of our work. Firstly, we start by presenting the graph theory as a tool for modeling of multi-UAVs systems. Then, a Euclidean representation of our system is introduced. After that, we present the simplified dynamic model of a multi-UAVs system. Finally, we introduce the LQR control in the perspective of our work.

A. Multi-UAVs modeling by graph theory

Graph theory is a powerful tool to model multi-agents systems. In the same way, we use Graph theory to represent the multi-UAVs systems. Graph theory is used to describe the topology of the system in a mathematical formulation. A multi-UAVs system is represented by a graph $G = (\mathcal{V}, E)$, where \mathcal{V} is a set of nodes $\mathcal{V} = \{1, 2, \dots, M\}$, and E is a set of edges $E \subseteq \{(i, j) : i, j \in \mathcal{V}, i \neq j\}$. Every node represents a UAV and edges depict the sensing between UAVs.

An adjacency matrix \mathcal{A} is an $M \times M$ matrix with elements $a_{i,j} = 1$ if $(i, j) \in E$ and $a_{i,j} = 0$ otherwise. Connected UAVs with a UAV i in the graph could be modeled by a set $\gamma_i = \{j \in \mathcal{V} : a_{i,j} \neq 0\}$. We assume that every UAV has an omni-directional detection capability, *i.e.* it can detect in all directions. This capability of each UAV means that there is a mutual detection between connected UAVs. The adjacency matrix is then symmetric $\mathcal{A}^T = \mathcal{A}$. Therefore, our graph is undirected. For more information about graph theory, the reader can refer to [8]. Fig. 1 shows a multi-UAVs system of four UAVs represented as an undirected graph.

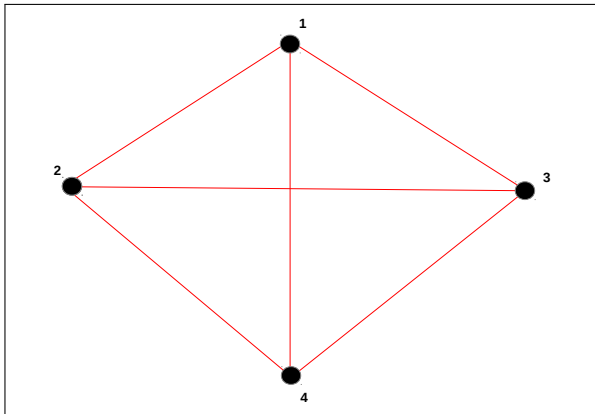


Fig. 1: Four UAVs in an undirected graph

B. Euclidean framework

Before working on the dynamics of UAVs, we need to represent our multi-UAVs system in the Euclidean space. Therefore, every node i in the graph is represented as a position $\mathbf{p}_i \in \mathbb{R}^f$ where f is the dimension of the space (example: $f = 2, 3$).

The neighbors of a UAV i is defined in the Euclidean space by:

$$\eta_i = \{\mathbf{p}_j \in \mathbb{R}^f, j \in \mathcal{V} : \|\mathbf{p}_j - \mathbf{p}_i\| < c\} \quad (1)$$

where $\|\cdot\|$ is the Euclidean norm, and c is the neighborhood range.

C. Simplified dynamics of multi-UAVs

In this paper, we suppose that the UAV has a simple dynamics. The proposed dynamic model is a double integrator. Furthermore, we assume that the body of a UAV is a particle in the Euclidean framework. Equation (2) is a state-space representation model of a UAV i in a flock.

$$\begin{cases} \dot{\mathbf{x}}_i &= A_i \mathbf{x}_i + B_i \mathbf{u}_i \\ \mathbf{y}_i &= C_i \mathbf{x}_i \end{cases} \quad (2)$$

where $\mathbf{x}_i = [\mathbf{p}_i \ \dot{\mathbf{p}}_i]^T$, $\mathbf{x}_i \in \mathbb{R}^l$ is the state vector, $\mathbf{u}_i \in \mathbb{R}^f$ is the control input and $\mathbf{y}_i \in \mathbb{R}^l$ is the system output, with $(l = 2f)$. In this paper, we consider $\mathbf{y}_i = \mathbf{x}_i$. The state-space matrices are:

$$A_i = \begin{pmatrix} \mathbf{0}_f & \mathbf{I}_f \\ \mathbf{0}_f & \mathbf{0}_f \end{pmatrix} \quad B_i = \begin{pmatrix} \mathbf{0}_f \\ \mathbf{I}_f \end{pmatrix} \\ C_i = \mathbf{I}_l$$

where, \mathbf{I}_f and \mathbf{I}_l are $f \times f$ and $l \times l$ identity matrices and $\mathbf{0}_f$ is an $f \times f$ zero matrix. This system is, therefore, controllable and observable.

Now, we can formulate the state-space model of multiple UAVs. Let us consider, as in [6], $\mathbf{p} = \text{col}(\mathbf{p}_1, \dots, \mathbf{p}_M) \in \mathbb{R}^{f \times M}$ the position configuration of all nodes in the multi-UAVs graph, where M is the number of UAVs. Moreover, all the states, the outputs, and the control inputs of the multi-UAVs system are written as $\mathbf{x} = \text{col}(\mathbf{x}_1, \dots, \mathbf{x}_M) \in \mathbb{R}^{l \times M}$, $\mathbf{y} = \text{col}(\mathbf{y}_1, \dots, \mathbf{y}_M) \in \mathbb{R}^{l \times M}$ and $\mathbf{u} = \text{col}(\mathbf{u}_1, \dots, \mathbf{u}_M) \in \mathbb{R}^{f \times M}$ respectively. Thus, we can write the multi-UAVs state-space system as follows:

$$\begin{cases} \dot{\mathbf{x}} &= \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u} \\ \mathbf{y} &= \mathbf{C} \mathbf{x} \end{cases} \quad (3)$$

where, $\mathbf{A} = \mathbf{I}_M \otimes A_i$, $\mathbf{B} = \mathbf{I}_M \otimes B_i$, and $\mathbf{C} = \mathbf{I}_M \otimes C_i$, with \otimes being the Kronecker product.

We define $\mathbf{x}_{\eta_i}^j = [\mathbf{p}_j \ \dot{\mathbf{p}}_j]^T \in \mathbb{R}^l$, with $j \in \gamma_i$, as the state of a neighboring UAV j measured by a UAV i . Then, the measured states of all the neighbors could be written as the vector $\mathbf{x}_{\eta_i} = \text{col}(\mathbf{x}_{\eta_i}^1, \dots, \mathbf{x}_{\eta_i}^q) \in \mathbb{R}^{l \times q}$ where $q = |\eta_i|$ is the number of neighbors.

D. LQR control

The LQR (Linear Quadratic Regulator) objective for a dynamic system such as (2) is to find the control input that minimizes the following cost function:

$$J_{LQR} = \int_0^{\infty} \mathbf{y}_i^T(t) \mathbf{Q} \mathbf{y}_i(t) + \mathbf{u}_i^T(t) \mathbf{R} \mathbf{u}_i(t) dt \quad (4)$$

where \mathbf{y}_i is the controlled output, \mathbf{Q} and \mathbf{R} are symmetric positive-definite matrices of dimensions $l \times l$ and $f \times f$ respectively. These two matrices are weighting matrices used to design the control law in order to respect the performance specification of the controlled system. This problem will be treated in a sequel paper dealing with a real-time application of our proposed strategy.

The control input \mathbf{u}_i , which minimizes J_{LQR} and which stabilizes the controlled output to zero, is then given by:

$$\mathbf{u}_i = -\mathbf{K} \mathbf{x}_i \quad (5)$$

where \mathbf{K} is the LQR gain matrix:

$$\mathbf{K} = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} \quad (6)$$

\mathbf{P} is a symmetric positive-definite matrix and the solution of the following Algebraic Riccati Equation [9].

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} + \bar{\mathbf{Q}} - \mathbf{P} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} = 0 \quad (7)$$

with $\bar{\mathbf{Q}} = \mathbf{C}^T \mathbf{Q} \mathbf{C}$.

In the trajectory tracking problem, the objective is to stabilize the error $e = \mathbf{x}_i - \mathbf{x}_d$ to zero, asymptotically. Then, the control input is written as:

$$\mathbf{u}_i = -\mathbf{K}(\mathbf{x}_i - \mathbf{x}_d) + \mathbf{u}_d \quad (8)$$

where, \mathbf{x}_d and \mathbf{u}_d are the state of the desired trajectory and the feedforward control, respectively. This trajectory should be feasible, *i.e.*, the trajectory must be an equilibrium solution of the closed-loop system [10]. In this paper, we consider $\mathbf{u}_d = 0$.

III. MAIN RESULTS

In this section, we present the main contributions of this paper. Firstly, we introduce a new behavior in addition to the rules of Reynolds. Then, we present an LQR control strategy for multi-UAVs flocking. Finally, the control strategy is extended to adapt the navigation behavior.

A. Aggregation behavior

Several studies on flocking and schooling behavior discuss a *behavioral tendency* in the individuals [1] [2]. In [1], a crucial question was: *What makes fish gather in a school?* The answer to this question reveals a tendency in the individuals to join and to stay within a school for different reasons, like hunting or protection.

In the technological side of dealing with flocking or schooling phenomena, scientists focus on a local problem. They tried to answer the question: *How do individuals avoid collision and align themselves?* They supposed that individuals are already in a flock. A global problem is to

consider, in addition to the local problem, the trend to form a flock from scattered individuals or small flocks.

The consideration of the flocking local problem is clear in the rules of Reynolds [5]. The rules of Reynolds focus on collision avoidance, velocity matching, and flock centering which are a local flocking problems. In [6], algorithm 1 of *Olfati-Saber* embodied the rules of Reynolds. This algorithm failed to perform the flocking, as emphasized by the author. Moreover, in the second algorithm of *Olfati-Saber*, a rendezvous point was defined to attract the individuals, in order to form a flock.

From a robotics and control theory point of view, flock formation should be autonomous. Therefore, no rendezvous point should be predefined to form a flock. In fact, Olfati-Saber's algorithms can achieve an autonomous formation control, without a rendezvous point, by increasing the neighboring region of agents. However, defining an attracting rendezvous point and restricting the neighboring region limit the computational cost due to the increase of connections between agents. This limitation of neighboring region leads to a fragmentation phenomenon, which appeared in the simulations results of *Olfati-Saber* without a rendezvous point [6]. The same considerations are also found in [7].

The limitation of neighboring regions is useful to fix distances between agents. However, naturally, the distances between flock or school individuals are not fixed [1]. In this paper, we prefer to follow the natural phenomenon of flocking and consider variable security distances between agents.

In this work, we introduce a new behavior, which views the flocking problem from a global perspective. We suppose that individuals in a flock are not yet grouped. This behavior is the *Aggregation behavior*.

Definition 1: (Aggregation behavior) It is the trend of individuals or sub-flocks to join each other, in their range of sight, in order to form a flock.

To model such behavior in a flock of UAVs, we propose a new set for each UAV. This set includes all the individuals in a UAV range of sight. It is defined as $\Gamma_i = \{v_j \in V : \|\mathbf{p}_j - \mathbf{p}_i\| < b\}$, where b is the UAV range of sight and $b > c$ (c is the neighboring region).

Unlike the works of *Olfati-Saber* in [6] and *G. Antonelli et al* in [7], there is no need for a rendezvous point to gather UAVs in a flock. Moreover, we realize that the fragmentation problem, issued in the two aforementioned works, is overcome by implementing this behavior in the individuals of the flock (see section IV).

B. LQR control for multiple UAVs

The problem of multiple agents (or vehicles) is that the computational cost and the control design complexity increase with the raise of the number of agents and the connectivity between them [11]. In this section, we introduce a new strategy of multiple UAVs control that shows the independence (or slight-dependence) of control design on the number of UAVs and on the connectivity. The linear nature

of the LQR control is one of the main characteristics that allowed us to develop this strategy. In this paper, we assume that each UAV can measure the states of all the agents in its range of sight.

The most crucial part of our work is the trajectory generation. In fact, we claim that the flocking is a trajectory generation problem rather than a control design issue. The desired trajectory for each UAV is generated by using the measured states of the UAVs in its field of view.

Drawing inspiration from Reynolds [5], one of the objectives of multi-UAVs flocking control is to ensure a collision-free region between the flock individuals. The control objective could be written as follows:

$$\|\mathbf{p}_j(t) - \mathbf{p}_i(t)\| \rightarrow d, \quad t \rightarrow \infty, \quad \forall j \in \eta_i, \forall i \in \mathcal{V} \quad (9)$$

This objective could be written in a vectorial form as:

$$\mathbf{p}_j(t) - \mathbf{p}_i(t) \rightarrow d \mathbf{n}_{ij}(t), \quad t \rightarrow \infty, \quad \forall j \in \eta_i, \forall i \in \mathcal{V} \quad (10)$$

where d is a security distance or the radius of the collision-free region, with $d < c$. \mathbf{n}_{ij} is the unit vector indicating the direction from UAV i to UAV j . This vector could be represented by two different ways, as in following equations

$$\mathbf{n}_{ij} = \frac{\mathbf{p}_j - \mathbf{p}_i}{\|\mathbf{p}_j - \mathbf{p}_i\|} \quad (11)$$

$$\mathbf{n}_{ij} = [\cos \theta_{ij} \quad \sin \theta_{ij}]^T \quad (12)$$

where θ_{ij} is the orientation angle. The representation in equation (12) is given in Fig. 2, with $f = 2$.

Moreover, another objective of multi-UAVs flocking is the velocity matching or consensus of all the UAVs in the flock. This objective could be represented mathematically as follows:

$$\dot{\mathbf{p}}_j(t) - \dot{\mathbf{p}}_i(t) \rightarrow 0, \quad t \rightarrow \infty, \quad \forall j \in \eta_i, \forall i \in \mathcal{V} \quad (13)$$

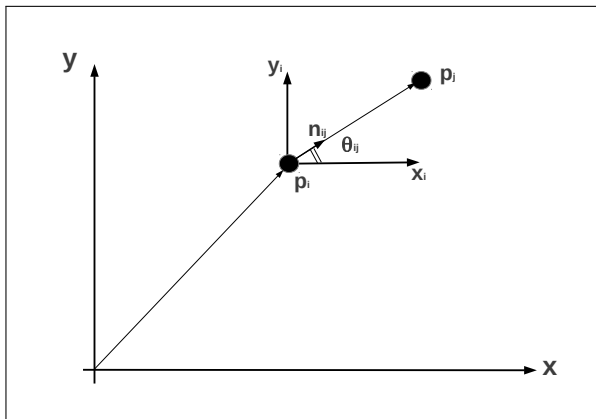


Fig. 2: Unit vector representation.

Let us consider the simple case of two UAVs i and j that detect each other and aggregate to form a flock. We deal with this problem as a trajectory tracking problem. The

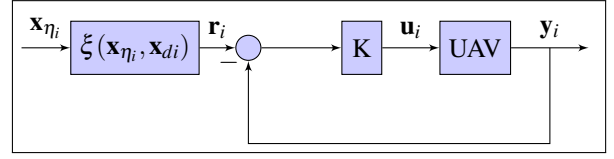


Fig. 3: The desired trajectory for UAV i is generated by using the measured states of the UAVs in its field of view. Only one controller is needed.

error system in UAV i could be, therefore, written by using equations (10) and (13), as the following:

$$\mathbf{e}_{ij} = \begin{pmatrix} \mathbf{p}_i - \mathbf{p}_j + d \mathbf{n}_{ij} \\ \dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j \end{pmatrix} = \mathbf{x}_i - \mathbf{r}_{ij} \quad (14)$$

where \mathbf{r}_{ij} is considered as the state of the desired trajectory. In other words, it is the desired relative state of UAV i . In fact, \mathbf{r}_{ij} is the state of the UAV j , shifted by a security distance d . \mathbf{r}_{ij} is written as follows:

$$\mathbf{r}_{ij} = \mathbf{x}_j - \mathbf{x}_{dij} \quad (15)$$

where \mathbf{x}_{dij} is a shifting state defined as:

$$\mathbf{x}_{dij} = \begin{pmatrix} d \mathbf{n}_{ij} \\ \mathbf{0} \end{pmatrix} \in \mathbb{R}^l \quad (16)$$

The control law, in the UAV i , that ensures the convergence of UAVs and the control constraint in (10) is written as follows:

$$\mathbf{u}_{ij} = -K(\mathbf{x}_i - \mathbf{r}_{ij}) \quad (17)$$

We calculate the control law of UAV j by the same way.

In the problem of multiple UAVs, we define the vector of the shifting states of a UAV i as $\mathbf{x}_{di} = \text{col}(\mathbf{x}_{di1}, \dots, \mathbf{x}_{dij}) \in \mathbb{R}^{l \times q} \quad \forall j \in \eta_i$. Moreover, we can obviously see that each UAV in a flock has more than one neighbor. Therefore, the dimensions of the vector of the neighboring states \mathbf{x}_{η_i} and the state \mathbf{x}_i of a UAV i are not compatible. This problem could be solved by mapping \mathbf{x}_{η_i} to be compatible with the dimensions of \mathbf{x}_i .

Let $\xi: \mathbb{R}^{l \times q} \rightarrow \mathbb{R}^l$ be a differentiable function that is used to generate the trajectory of a UAV i in the flock in order to achieve a desired behavior. Then \mathbf{r}_i could be written as:

$$\mathbf{r}_i = \xi(\mathbf{x}_{\eta_i}, \mathbf{x}_{di}) \quad (18)$$

The function in (18) means that it is not necessary to design multi-controllers for each UAV. This method decreases the computational cost. The block diagram in Fig. 3 shows the application of this strategy on a UAV i .

We define the average function of a vector $z = \text{col}(z_1, \dots, z_n)$ as follows:

$$\text{Ave}_n(z) = \frac{1}{n} \sum_{j=1}^n z_j \quad (19)$$

Theorem 1: (Average control strategy) Consider q UAVs that tend to form a flock. The following control law of UAV $i \in \mathcal{V}$

$$\mathbf{u}_i = -K\mathbf{x}_i + K\xi(\mathbf{x}_{\eta_i}, \mathbf{x}_{d_i}) \quad (20)$$

ensures the aggregation behavior and asymptotically stabilizes the flock formation. The function $\xi(\mathbf{x}_{\eta_i}, \mathbf{x}_{d_i})$ is written as the following:

$$\xi(\mathbf{x}_{\eta_i}, \mathbf{x}_{d_i}) = \text{Ave}_n(\mathbf{x}_{\eta_i} - \mathbf{x}_{d_i}) \quad (21)$$

Proof: The LQR control law in (17) stabilizes the UAV i and ensures the aggregation of two UAVs. For q UAVs $\in \eta_i$, we need q control laws. Then, we have:

$$\begin{aligned} \mathbf{u}_{i1} &= -K\mathbf{x}_i + K\mathbf{r}_{i1} \\ &\vdots \\ \mathbf{u}_{iq} &= -K\mathbf{x}_i + K\mathbf{r}_{iq} \end{aligned} \quad (22)$$

A solution to decrease the number of controllers is to calculate the average of the q control laws and apply the output of the average to the system. We obtain:

$$\mathbf{u}_i = \frac{1}{q} \sum_{j=1}^q \mathbf{u}_{ij} \quad (23)$$

By applying (23) we get:

$$\mathbf{u}_i = -K\mathbf{x}_i + \frac{K}{q} \sum_{j=1}^q \mathbf{r}_{ij} \quad (24)$$

Replacing \mathbf{r}_{ij} by its expression in (15), then we get the resultant control law given by equation (20). \blacksquare

Remark 1: The design of this control law in (20) is independent of the number of UAVs.

Remark 2: The security distance d is user-defined and it is dependent on the number of UAVs. Gaps between flock individuals will converge to distances around d . The security distance should be set higher when the number of UAVs increases. The distance d should be determined carefully in order to avoid collision, since the state of each UAV converges to an average vector.

C. Navigation behavior

In this section, we introduce a navigation behavior that allows a flock of multiple UAVs to navigate from initial positions toward a destination. The destination is predefined by the user and stocked in the memory of UAVs. The navigation of a multi-UAVs flock is in a free space, *i.e.* there are no obstacles. In this section, no obstacles are faced in the path of the navigation.

The control strategy used to perform the navigation task is similar to the average strategy. It consists of defining a navigation vector that contains the position and the velocity of the destination. It could be written as $\mathbf{x}_{nav} = [\mathbf{p}_{nav} \ \dot{\mathbf{p}}_{nav}]^T \in \mathbb{R}^l$. In this paper, we focus on the navigation toward a fixed destination rather than tracking a trajectory or a mobile target. Therefore, the velocity in the navigation vector is $\mathbf{0} \in \mathbb{R}^f$. Tracking trajectories will be addressed in a future work.

Now, we define the following function to achieve the aggregation and the navigation behaviors:

$$\xi(\mathbf{x}_{\eta_i}, \mathbf{x}_{d_i}, \mathbf{x}_{nav}) = \text{Ave}_n(\text{col}(\mathbf{x}_{\eta_i} - \mathbf{x}_{d_i}, \mathbf{x}_{nav})) \quad (25)$$

Therefore, the control law that allows the multi-UAVs system to make the aggregation and the navigation behaviors is given as the following:

$$\mathbf{u}_i = -K\mathbf{x}_i + K\xi(\mathbf{x}_{\eta_i}, \mathbf{x}_{d_i}, \mathbf{x}_{nav}) \quad (26)$$

IV. SIMULATION RESULTS

In this section, we illustrate simulations of the multi-UAVs system. We simulate both behaviors, presented in the previous section, by using MATLAB. The simulation is done in a two-dimensional space. The initial positions of UAVs are generated randomly by using the normal distribution function. Moreover, the initial velocities of UAVs are set at zeros. The step size of the simulation is 0.01s. The weighting matrices are: $Q = \mathbf{I}_4$ and $R = 10^{-3} \times \mathbf{I}_2$. The matrix K is calculated by using the *lqr* MATLAB function [9]. Furthermore, UAVs are represented as red circles, where the real position of a UAV is the center of the circle. For clarity purposes, the size of UAVs is chosen greater than simulation scales, which could yield to ambiguous separation distances. The videos of the simulation results are available at the link in [12].

A. Aggregation behavior

In this part, we simulate the aggregation of 5 UAVs. The variance and the mean of the initial positions are set at 21 and 0, respectively. The range of the two-dimensional simulation plan is the x y position square $[-20, 20]$. The flocking parameters are defined as the following: $d = 5$, $c = 30$. Therefore, we suppose that the range of sight of each UAV covers the entire simulation plan. Moreover, we suppose that each UAV can measure the states of all the other UAVs in its range of sight.

Fig. 4 shows six snapshots of running simulation over time. This simulation is performed by using the *average control strategy* (20). The UAVs start scattered in the simulation plan. By analyzing the simulation, each UAV performs a repulsion action when there are some UAVs in its collision-free region. This action ensures a security distance between UAVs. Moreover, every UAV activates the aggregation behavior when a UAV or more are in its range of sight. Therefore, UAVs start to converge to each other to form a flock. The convergence is performed without collision. After a sufficient time, a flock of UAVs is formed and stabilized. In this simulation, we do not define a rendezvous point, and there is no fragmentation in the flock. Moreover, we can see clearly security distances between individuals. These security distances ensure collision-free flock.

B. Navigation

In this part, we illustrate the navigation of 8 UAVs in a free space. The variance of initial positions is 21 and the mean is -30 . The simulation plan is expanded to be the x y position

square $[-60, 60]$. The flocking parameters are defined as the following: $d = 12$, $c = 30$. The desired destination is the position $(40, 40)$, so the navigation vector is $[40 \ 40 \ 0 \ 0]^T$.

Fig. 5 shows six snapshots of the navigation behavior realized by using the control law in (26), which is applied in every UAV in the flock. The flock performs the aggregation behavior while navigating toward the desired destination. Safety distances are kept throughout the navigation path. Finally, the flock is uniformly formed and stabilized at the destination position.

V. CONCLUSIONS

In this paper, we addressed the control problem of multiple Unmanned Aerial Vehicles (UAVs) flocking by using a behavior-based strategy. We conceived a behavior intending to address the control design towards a successful achievement of the flocking task without fragmentation. Moreover, through its implementation in UAVs, no rendezvous point was needed to perform flocking. We designed an LQR control law, which is independent of the number of UAVs in the flock. Our proposed strategy dealt with the flocking problem from a measurement-mapping perspective. Simulation results showed the efficiency of our strategy in the aggregation and in the navigation of multi-UAVs flocking in a free space.

Our future work will focus on expanding the proposed strategies on non linear models of (Unmanned Aerial Vehicles) UAVs. Moreover, we will focus on implementing these strategies in a real-time application.

ACKNOWLEDGMENT

This work was carried out and funded in the framework of the Labex MS2T. It was supported by the French Government, through the program "Investments for the future" managed by the National Agency for Research (Reference ANR-11-IDEX-0004-02)

REFERENCES

- [1] B. Partridge, "The structure and function of fish schools," *Scientific American*, vol. 246, no. 06, pp. 114+, 1982.
- [2] I. D. Couzin, "Collective cognition in animal groups," *Trends in Cognitive Sciences*, vol. 13, no. 1, pp. 36 – 43, 2009.
- [3] J.-A. Guerrero and R. Lozano, *UAV flight formation control*. John Wiley-ISTE, 2012.
- [4] M. Chiamonti, F. Giuliotti, and G. Mengali, "Formation control laws for autonomous flight vehicles," in *14th Mediterranean Conference on Control and Automation. MED '06.*, pp. 1 –5, june 2006.
- [5] C. W. Reynolds, "Flocks, herds, and schools: A distributed behavioral model," in *Computer Graphics*, pp. 25–34, 1987.
- [6] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, pp. 401 – 420, march 2006.
- [7] G. Antonelli, F. Arrichiello, and S. Chiaverini, "Flocking for multi-robot systems via the null-space-based behavioral control," *Swarm Intelligence*, vol. 4, no. 1, pp. 37–56, 2010.
- [8] R. Diestel, *Graph Theory*, vol. 173 of *Graduate Texts in Mathematics*. Springer-Verlag, Heidelberg, third ed., 2005.
- [9] J. P. Hespanha, *Linear Systems Theory*. Princeton, New Jersey: Princeton Press, Sept. 2009. ISBN13: 978-0-691-14021-6.
- [10] R. M. Murray, *Optimization-Based Control*. California Institute of Technology, 2010.[Online]. Available: http://www.cds.caltech.edu/~murray/amwiki/index.php/Supplement:_Optimization-Based_Control
- [11] R. Olfati-Saber and R. M. Murray, "Distributed cooperative control of multiple vehicle formations using structural potential functions," in *IFAC World Congress*, 2002.
- [12] <https://www.hds.utc.fr/~ifantoni/dokuwiki/doku.php?id=en:videos>

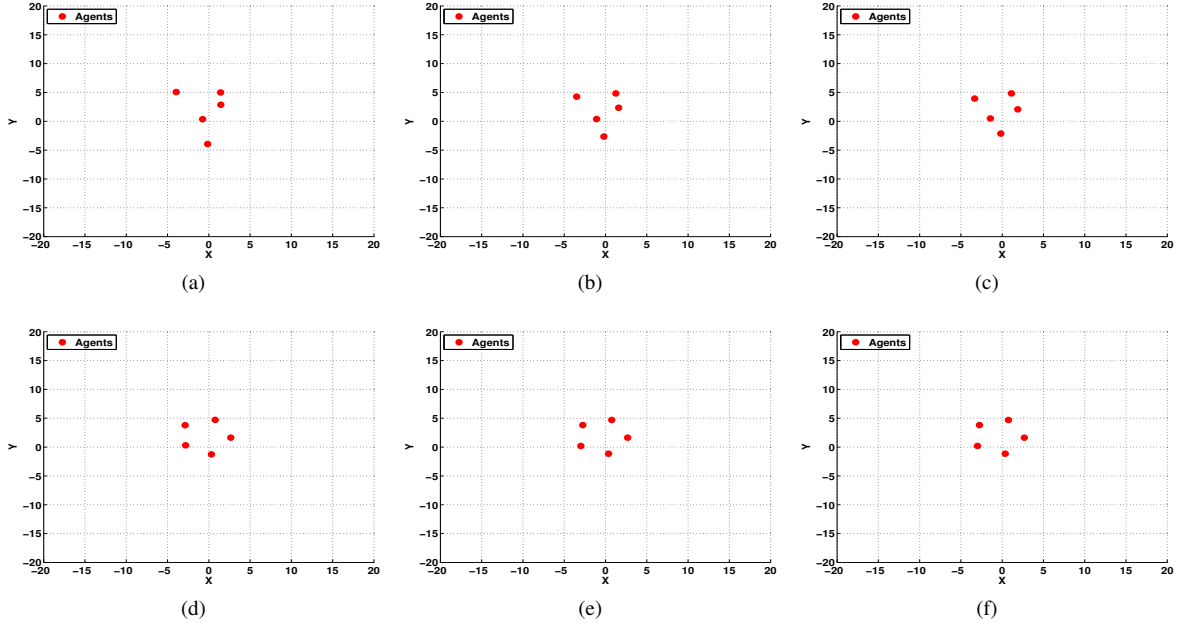


Fig. 4: multi-UAVs aggregation snapshots (Average strategy). No fragmentation in the flock, even if there is no rendezvous point.

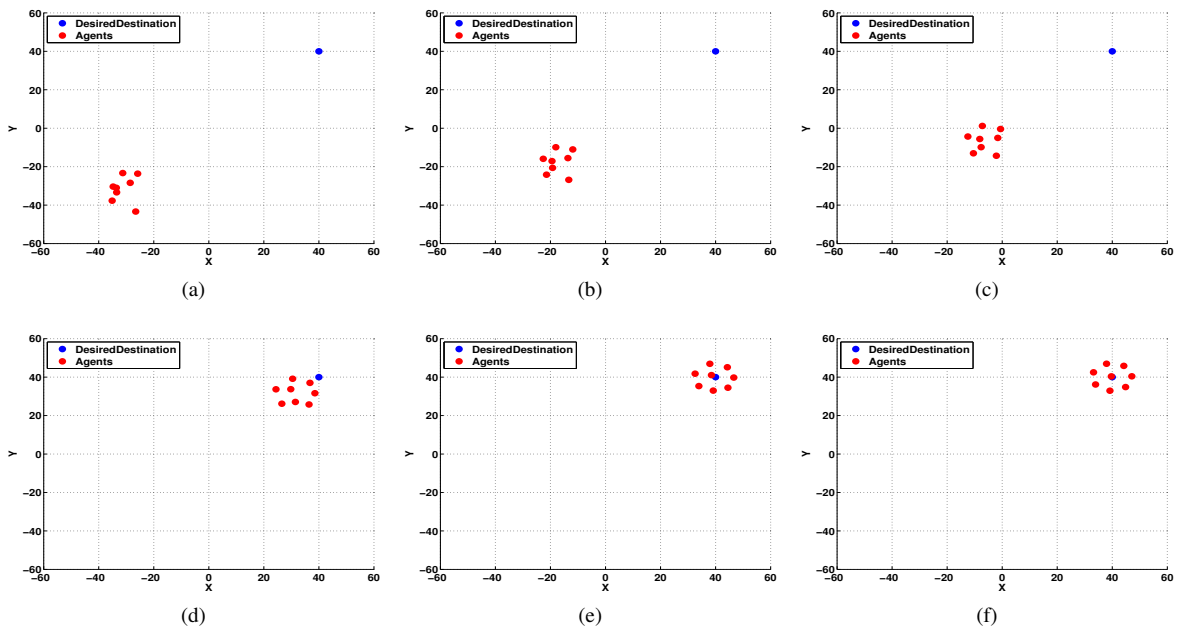


Fig. 5: multi-UAVs navigation snapshots. The algorithm ensures the aggregation and the navigation.