A Review of Consensus-based Multi-agent UAV Applications

A Review of Consensus-based Multi-agent UAV Applications

Conference Paper · June 2021

DOI: 10.1109/ICUAS51884.2021.9476858

3 authors:

Fausto Francesco Lizzio
Politecnico di Torino

Elisa Capello
Politecnico di Torino

Giorgio Guglieri
Politecnico di Torino

A Review of Consensus-based Multi-agent UAV Applications

Fausto Francesco Lizzio[1], Elisa Capello[2] and Giorgio Guglieri[2]

***Abstract*—In this paper, a review of distributed control for multi-agent systems is proposed, focusing on consensus-based applications. Both rotary-wing and fixed-wing Unmanned Aerial Vehicles (UAVs) are considered. On one side, methodologies and implementations based on collision and obstacle avoidance through consensus are analyzed for multirotor UAVs. On the other hand, a target tracking through consensus is considered for fixed-wing UAVs. This novel approach to classify the literature could help researchers to assess the outcomes achieved in these two directions in view of potential practical implementations of consensus-based methodologies.**

***Index Terms*—consensus, distributed control, UAV, swarm, multi-agent**

I. Introduction

Distributed control has received a considerable amount of attention among researchers over the last decades, due to the substantial advantages coming from the deployment of multiple agents in comparison with a single operating unit. In particular, Unmanned Aerial Vehicles (UAVs) multi-agent applications represent one of the most promising areas of interest of distributed control, as the typical weakness of a single-UAV mission, including low endurance, limited communication range, or inaccurate sensing capability, can be overcome by employing a swarm of drones.

In this paper, a review of distributed control in multi-agent systems is presented, considering consensus-based methodologies. Consensus aims at reaching an "agreement" among the agents of a system on a given variable of interest, exploiting only local information exchange among neighbors [1]. As briefly said, the main objective of this review is to analyze some of the most relevant and recent advances in the literature regarding consensus-based multi-agent UAV systems, classifying them in function of the considered platform configurations. Then, both fixed-wing and multirotor UAVs are considered, focusing on different applications.

Due to the extensive investigation related to the subject, several reviews of consensus-based UAV applications can be found in the literature. In particular, [2] provides a thorough theoretical framework for general multi-agent consensus, analyzing the conditions under which consensus can be reached in different configurations. In [3] and [4], a survey of the most common practical issues of applying consensus to a multi-UAV system is presented. Ding et.al. [5] provides an overview on distributed consensus protocols taking into account communication/measurements noise, while [6] focuses on event-triggered consensus. An approach similar to the consensus strategy can be found in [7] and [8], where the concept of inter-neighbors communication is used to build a rigid graph topology. In [9], conditions for reaching optimal consensus are provided.

As briefly said before, the main objective of this paper is to analyze consensus-based strategies for different platform configurations. Two main classes are here considered: fixed and rotary wing [10], [11]. Fixed-wing UAVs are characterized by considerable endurance and high minimum airspeed and can fly over large distances. On the contrary, rotary-wing UAVs, in particular multirotor UAVs, usually provide a lower endurance, but they allow hovering flight and vertical take-off and landing. The peculiar properties of the fixed-wing platforms make them suitable to operate in large and sparse areas, performing missions as patrolling, surveillance or data gathering over vast regions [12]. On the other hand, multirotors are generally deployed in confined and dense areas, such as indoor and urban environment, to perform operations as data collection in hard-to-reach sites, or delivery services [13].

In this scenario, switching from a single drone to a swarm of UAVs poses new challenges that need to be correctly addressed in relation to the adopted distributed control methodology. Indeed, operating in a densely crowded environment, a swarm of multirotor UAVs requires an accurate obstacle and collision avoidance strategy. Instead, multiple fixed-wing drones need an efficient framework for distributed sensing to enhance the precision of target detection and tracking.

This paper aims at categorizing the literature focusing on one side on the implementation of collision and obstacle avoidance through consensus in rotary-wing UAVs, and on the other hand on the implementation of distributed target tracking through consensus in fixed-wing UAVs. This approach to classify the literature could help researchers to assess the outcomes in view of future implementations of consensus methodologies.

The rest of the paper is organized as follows: In Section 2, some preliminaries on graph theory and consensus control are provided. In Section 3, we review the common issues and benefits of applying consensus to a swarm of UAVs. Section 4 and 5 describe the latest trends of consensus in relation to rotary and fixed wing multi-agent systems, respectively.

[1]Fausto F. Lizzio is with Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Torino, Italy, fausto.lizzio@polito.it

[2]Elisa Capello and Giorgio Guglieri are with Department of Mechanical and Aerospace Engineering, Politecnico di Torino, CNR-IEIIT, Torino, Italy, elisa.capello,giorgio.guglieri@polito.it

Finally, concluding remarks are provided in Section 6.

II. PRELIMINARIES

In this section, we will provide a theoretical framework for analysing consensus based on the most relevant surveys found in the literature [1], [14], [15].

A. Graph Theory

In the scenario of inter-agents communication in a multi-vehicle system, it is natural to model the exchange of information among n drones through the concepts of graph theory. A graph $\mathcal{G}$ is defined by a nonempty set of n nodes $\mathcal{V}$ connected by a set of edges $\mathcal{E}$, with $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. The communication flowing among multiple drones of a swarm can be either directed or undirected. Given two agents i and j belonging to the set $\mathcal{V}$, in a directed graph information can flow from agent i to agent j, but not necessarily vice versa. The undirected graph considers only bidirectional communication, in which both agents send and receive information to each other simultaneously [1]. In the case of directed communication topology, the edges in $\mathcal{E}$ have to be ordered, meaning that $(i, j) \in \mathcal{E}$ states that j, the child node, can receive information from i, the parent node, and not vice versa. In undirected graphs, it follows that the edges in $\mathcal{E}$ do not contain any order. If $(i, j) \in \mathcal{E}$, in both directed and undirected cases, the two nodes are said to be connected or neighbors. Clearly, self-edges of the kind (i, i) are not considered and do not belong to $\mathcal{E}$.

A directed graph is said to be strongly connected if there is an ordered sequence of edges in the set $\mathcal{E}$ from every node i to every other node j. An undirected graph is said to be connected if there is a sequence of edges in the set $\mathcal{E}$ between any two nodes in $\mathcal{G}$ [16].

In this context it is possible to define the adjacency matrix $\mathcal{A} \in \mathbb{R}^{n\times n}$, in such a way that $a_{ij} > 0$ if the nodes i and j are neighbors, while $a_{ij} = 0$ otherwise. The value of a_{ij} is a positive weight, that could be proportional to the inverse of the distance between two nodes, or else could be set equal to 1 when $(i, j) \in \mathcal{E}$. The matrix $\mathcal{A}$ is symmetrical for an undirected graph. Starting from $\mathcal{A}$, it is possible to define the Laplacian matrix $\mathcal{L}$, such that:

$$l_{ii} = \sum_{j=1, j\neq i}^{n} a_{ij}, \quad l_{ij} = -a_{ij} \quad \text{for } i, j = 1, ..., n \tag{1}$$

Notice that $\mathcal{L}$ has zero row sum [17]. The multiplication of each row of $\mathcal{L}$ for the vector of ones $\mathbf{1}_n$ is always equal to 0. This means that the Laplacian matrix always has at least one null eigenvalue, associated with the eigenvector $\mathbf{1}_n$.

B. Consensus algorithm for continuous time systems

If the communication between neighbors allows continuous information sharing, or if the communication bandwidth is large enough, the system could be modelled as a continuous time one, ruled by a set of differential equations [1]. Given a generic variable x_i with $i = 1, ..., n$ (n is the number of drones), the aim of consensus is to obtain the convergence of x_i to a common value, only exploiting local information exchange. Let $\mathcal{N}_i$ be the set of neighbors of a node i. The traditional form of consensus algorithm is

$$\dot{x}_i(t) = -\sum_{j\in\mathcal{N}_i} a_{ij}[x_i(t) - x_j(t)]. \tag{2}$$

So, the value of the variable of interest x_i is driven towards the values of the variables of interest of its neighbors as $\|x_i(t)-x_j(t)\| \to 0$ as $t \to \infty$ [18], [19]. It is straightforward to notice how applying locally (2) is equivalent to apply globally

$$\dot{x}(t) = -\mathcal{L}\ x(t), \tag{3}$$

where $x(t) = [x_1(t), ..., x_n(t)]^T$. This means that the distributed multi-agent system behaves as a linear dynamic system where $x(t)$ is the state vector and $-\mathcal{L}$ is the state matrix. The stability properties of such a system depend on the spectrum of the state matrix. As already said, matrix $-\mathcal{L}$ always has at least one null eigenvalue, meaning that the system is not asymptotically stable. However, the system could be internally stable if 0 is the only null eigenvalue of the spectrum, and the other eigenvalues have negative real parts.

In [1] it is shown that the opposite $-\mathcal{L}$ of the Laplacian associated to a directed graph $\mathcal{G}$ has a simple 0 eigenvalue and all other eigenvalues with negative real parts if and only if $\mathcal{G}$ has a directed spanning tree. Likewise, the opposite $-\mathcal{L}$ of the Laplacian associated to an undirected graph $\mathcal{G}$ has a simple 0 eigenvalue and all other eigenvalues are negative if and only if $\mathcal{G}$ is connected. This means that the system is internally stable, and its state variables $x_i(t)$ remain bounded for any initial condition $x_i(0)$.

From algebraic control theory, we know that the equilibrium state of such a linear system is only affected by its kernel, i.e., the eigenvectors associated to the null eigenvalue [20].

In particular, $x(t) \to (\mathbf{1}_n\nu^T)x(0)$ as $t \to \infty$, where ν is the unit left eigenvector of $\mathcal{L}$ associated to the eigenvalue 0. Since $(\mathbf{1}_n\nu^T)$ is a matrix with identical rows, it is clear how each x_i tends to a common value given by $\sum_{i=1}^{n} \nu_i x_i(0)$, that is, the system reaches consensus [1].

C. Consensus algorithm for discrete time systems

A similar approach can be adopted when information data are shared in discrete packets, i.e., the system can be modelled as a discrete time one, ruled by difference equations [1]. In this case, the update of the information variable $x[k]$ is

$$x_i[k+1] = \sum_{j\in\mathcal{N}_i} d_{ij}x_j[k], \tag{4}$$

where d_{ij} is the element of a row stochastic matrix $\mathcal{D}$ such that $d_{ij} = 0$ when no information flows from i to j or vice versa, and $d_{ij} > 0$ otherwise. Moreover, $d_{ii} > 0$, meaning that the updated value $x_i[k+1]$ is a weighted average of the current state of the agent i itself and of its neighbors $\mathcal{N}_i$. Applying (4) locally is equivalent to

$$x[k+1] = \mathcal{D}\ x[k], \tag{5}$$

that is a linear discrete system whose state matrix is $\mathcal{D}$. Being row stochastic, $\mathcal{D}$ always has at least one eigenvalue equal to 1 associated to the eigenvector $\mathbf{1}_n$ [18], [19]. In [1] it is shown that matrix $\mathcal{D}$ associated to a directed graph $\mathcal{G}$ has a simple 1 eigenvalue and all other eigenvalues strictly inside the unit circle if and only if $\mathcal{G}$ has a directed spanning tree. Likewise, matrix $\mathcal{D}$ associated to an undirected graph $\mathcal{G}$ has a simple 1 eigenvalue and all other eigenvalues strictly inside the unit circle if and only if $\mathcal{G}$ is connected. The system is internally stable in such cases.

As in the continuous time scenario, $x[k] \to (\mathbf{1}_n \mu^T)x[0]$ as $k \to \infty$, where μ is the unit left eigenvector of $\mathcal{D}$ associated to the eigenvalue 1 [20]. From this consideration, it is clear how each x_i tends to a common value given by $\sum_{i=1}^{n} \mu_i x_i[0]$, that is, the system reaches consensus [1].

D. Consensus for a second-order dynamic system

The previous subsections focused on a generic variable of interest modelled as a first-order integrator dynamics. As in [1], many systems can be modeled as double-integrators in a linearized way. In the framework of a continuous time system,

$$\dot{\xi}_i = \zeta_i, \quad \dot{\zeta}_i = \mathbf{u}_i, \quad \text{for } i = 1, ..., n, \tag{6}$$

where $\xi_i \in \mathbb{R}^m$ and $\zeta_i \in \mathbb{R}^m$ are the position and velocity of the i^{th} agent. The most comprehensive consensus protocol for a second-order dynamic system is the following

$$\begin{aligned} \mathbf{u}_i = & \underbrace{\dot{\zeta}_i^{ref} - \beta[\kappa(\xi_i - \xi_i^{ref}) + \gamma(\zeta_i - \zeta_i^{ref})]}_{\textit{reference model tracking}} + \\ & \underbrace{- \sum_{j \in \mathcal{N}_i} a_{ij}[(\xi_i - \xi_j) + \gamma(\zeta_i - \zeta_j)]}_{\textit{consensus building}}, \end{aligned} \tag{7}$$

where β, γ and κ are non-negative real parameters, a_{ij} is the element of the adjacency matrix $\mathcal{A}$, and $(\xi_i^{ref}, \zeta_i^{ref})$ represent reference states for (ξ_i, ζ_i), such that $\dot{\xi}_i^{ref} = \zeta_i^{ref}$. This reference state can be locally associated at each agent, or provided by an external source according to the adopted strategy. In the latter case, $(\xi_i^{ref}, \zeta_i^{ref}) = (\xi^{ref}, \zeta^{ref})$ for all $i = 1, ..., n$.

The consensus building part in protocol (7) is required to achieve consensus between members of the swarm, through the coupling of states (ξ_i, ζ_i). This allows to reach $\xi_i \to \xi_j$, and $\zeta_i \to \zeta_j$ as $t \to \infty$ for all $i, j = 1, ..., n$. With the presence of the consensus building term only, the states of the agents would reach a common value given by a weighted average of their initial states, as seen for Eqs. (2) and (4).

The reference model tracking part in protocol (7) is able to make the state of the agents converge to a reference state. As pointed out in [1], it is useful to distinguish two operational situations. In the first one, consensus has to be reached with a reference for information state derivatives only, meaning that $\kappa = 0$. This allows to reach $\zeta_i \to \zeta_i^{ref}$ as $t \to \infty$ for all $i = 1, \dots, n$. In the second case, consensus has to be reached with a reference for both information state and their derivatives, so that $\kappa = 1$. This allows to reach $\xi_i \to \xi_i^{ref}$ and $\zeta_i \to \zeta_i^{ref}$ as $t \to \infty$ for all $i = 1, ..., n$.

Altogether, $\xi_i \to \xi_j \to \xi^{ref}$, and $\zeta_i \to \zeta_j \to \zeta^{ref}$ as $t \to \infty$. In [1], [21], it is shown that under the typical conditions regarding graph topology (presence of a directed spanning tree in the directed case, being connected in the undirected one), plus a lower bound on the value of γ for the directed case, through protocol (7) the system achieves consensus.

III. UAV SWARM APPLICATIONS

A. Formation control

Consensus has been actively studied in relation to UAV swarm applications. One of the most investigated issues in this field is formation control [21]–[25]. Generating a UAV formation through consensus is particularly favorable given the limited communication resources available onboard. Following the classification proposed in [7], formation through consensus is a displacement-based control methodology, in which a shared global coordinate system between every agent is not needed. Instead, each UAV is required to have its own local coordinate system, that has to be aligned to the global one. In this framework, the agents can sense the relative positions (displacements) of their neighbors and achieve a desired formation. Note that this configuration is the midpoint between a swarm in which all the agents share the same global coordinate system and a situation in which each drone has its own local reference system, not aligned to a global one. In this case, only distance-based control is possible [7], [26], [27].

Displacement-based formation control is usually categorized in three main strategies that can be realized through the consensus protocol (7): leader-follower (LF), behaviour-based (BB) and virtual structure (VS). According to the aim of the control methodology, it is possible to assign to each of the variable of interest ξ, ζ and (ξ^{ref}, ζ^{ref}) specific roles. In table I, the results found in [21] are summarized.

TABLE I
CONSENSUS STRATEGIES

	ξ_i	ζ_i	(ξ^{ref}, ζ^{ref})	κ
LF	$r_i - \delta_i$	v_i	(r^L, v^L)	0
BB	$r_i - r_i^{ref}$	$v_i - v_i^{ref}$	(r_i^{ref}, v_i^{ref})	1
VS	$r_i - r_i^F$	$v_i - v_i^F$	$(r_i^O + r_i^F, v_i^O + v_i^F)$	0

In the leader-follower strategy, the information flows through a directed graph from a leader UAV to all the followers. In this case, the followers share with their neighbors the information regarding their position r_i minus a constant δ_i and their velocity v_i, to reach $r_i - \delta_i \to r_j - \delta_j$ and $v_i \to v_j$ as $t \to \infty$. Note that $r_i - r_j \to \delta_i - \delta_j$, and the values of δ_i and δ_j can be chosen such that $\delta_i - \delta_j = \Delta_{ij}$, i.e. the desired separation between agents. Moreover, the leader broadcasts to every agent in the swarm its information state derivatives $(\dot{r}^L, \dot{v}^L)$ with the aim of reaching $v_i \to v^L$ as $t \to \infty$ for $i = 1, \dots, n$, where L refers to the leader.

In the behaviour-based approach, several behaviours are weighted in the control input of each UAV, so that every agent is able to simultaneously maintain a desired inter-agent distance (formation keeping behaviour) while tracking a desired final position $r^{ref} = [r_i^{ref}, \ldots, r_n^{ref}]$ (goal-seeking behaviour). In this case, a desired reference final position r_i^{ref} is locally known to each drone. The agents share with their neighbors their deviation vector $r_i - r_i^{ref}$ and its derivative $v_i - v_i^{ref}$, to reach $r_i - r_i^{ref} \to r_j - r_j^{ref}$ and $v_i - v_i^{ref} \to v_j - v_j^{ref}$ as $t \to \infty$. Note that, if consensus is reached upon the deviation vectors, the agents keep the desired formation shape. Every drone is able to reach its destination r_i^{ref} due to the reference model tracking terms of (7). Even if there is no information exchange among the agents, each drone can track its own trajectory. However, better transient performance are obtained with information coupling [1].

In the virtual structure strategy, each agent is required to keep a desired (possibly time-varying) distance with respect to a virtual formation center, that is chosen as the shared information variable among the drones. Thus, consensus has to be reached on the state of the virtual formation center (r^O, v^O). Each agent shares with their neighbors their current understanding of the formation center state (r_i^O, v_i^O) expressed as $(r_i - r_i^F, v_i - v_i^F)$, where r_i^F and v_i^F are the position and velocity of the agent i with respect to the formation center. In this way, $r_i^O \to r_j^O$ and $v_i^O \to v_j^O$ as $t \to \infty$, and the formation is preserved. Moreover, through the reference model tracking terms, the protocol is able to make each agent follow its current understanding of the formation center velocity v_i^O using $(r_i^O + r_i^F, v_i^O + v_i^F)$ as the reference for information state derivatives. Indeed, $v_i \to v_i^O + v_i^F$, that is, $v_i - v_i^F \to v_i^O$ as $t \to \infty$ for $i = 1, \ldots, n$.

B. Time-varying topology

The design of a consensus protocol has to take into account the fact that communication between any pair of agents in the swarm may be unreliable or could fail [1]. This issue is particularly relevant when the system is operating in a densely crowded environment, in which adverse conditions (obstacles, environmental disturbances) may degrade the quality of information sharing. In such cases, the adjacency matrix $\mathcal{A}(t)$ and the laplacian $\mathcal{L}(t)$ become time-variant and the analysis of consensus stability has to be performed accordingly. It is typical the assumption that the communication topology $\mathcal{G}$ is piece-wise constant during finite time intervals called dwell times that usually have a positive lower bound τ_D [1]. This means that $a_{ij}(t) \in \mathcal{A}(t)$ and $l_{ij}(t) \in \mathcal{L}(t)$ can be considered constant or continuous (e.g. if $a_{ij}(t)$ is inversely proportional to the inter-agents distance) at least for τ_D seconds.

In [1], conditions for consensus are given for a first-order linear dynamic system. In particular, a directed (undirected) time-varying topology reaches consensus if the union of the directed (undirected) graphs across each contiguous, nonempty, uniformly bounded time interval in which $\mathcal{A}(t)$ and $\mathcal{L}(t)$ are constant or continuous has a directed spanning tree (is connected). In [28], it is shown that, for a system of multiple single-integrator dynamics agents, if the limit of the laplacian $\mathcal{L}(t)$ as $t \to \infty$ is stable, then the system reaches consensus. This means that only the connectivity properties of the limit topology influence the stability of the system.

Concerning second-order linear dynamic systems, the conditions for consensus are generally more stringent. Indeed, [1] proves that for an undirected switching topology, the graph $\mathcal{G}$ has to be connected at each time instant in order to reach consensus. In the directed case, not only the graph $\mathcal{G}$ has to contain a spanning time at every time instant, but a lower bound on the dwell time τ_D has to be respected. However, in a more recent study [29], it is shown that for a second-order nonlinear dynamic system, consensus over a directed switching topology can be reached even if the graph does not contain a spanning tree every time instant. Letting T_C be the total time in which $\mathcal{G}$ has a directed spanning tree and T_U the total time in which $\mathcal{G}$ does not contain it, [29] provides a lower bound on the ratio $\frac{T_C}{T_U}$ such that consensus can still be reached. Similar results were found in [30] regarding a general linear system described by a state-space representation. In [29], [30], the switching between a known finite set of different topologies is performed both in a deterministic and stochastic fashion.

Another approach adopted in the literature when dealing with unreliable communication sharing is to consider the topology itself as stochastic [31], meaning that the communication links between any pair of nodes have an associated probability p_{ij}, such that:

$$a_{ij} = \begin{cases} 0 & \text{with probability } p_{ij} \\ 1 & \text{with probability } 1 - p_{ij} \end{cases}$$

In [31], conditions for consensus are provided for a general linear multi-agent system. In particular, if the probability of connectivity of the network is not zero and some linear matrix inequalities in the design of the protocol are met, the system reaches consensus. The main contribution of [31] is that this approach does not require any knowledge on the set of feasible topologies. In [32], these results are applied to high-order dynamic systems with uncertain nonlinearities.

Note that all the previous results apply to continuous time systems.

C. Time Delays

In practical implementations of consensus, another prominent issue to take into account is the presence of delays arising from the limited capabilities of the hardware or the environment. In the literature, these delays are usually classified as communication delays, due to the finite speed of information transmission between agents, and input delays, due to signal processing after receipt and execution time of the actuators of each agent [1]. Studies on the subject usually aim at finding an upper bound on the maximum delay admissible so that the system is still able to reach consensus [3]. As an example, the algorithm regarding continuous first-order linear dynamic system becomes

$$\dot{x}_i(t) = -\sum_{j \in \mathcal{N}_i} a_{ij}[x_i(t - \tau_i) - x_j(t - \tau_{ij})], \qquad (8)$$

to include time delays. In particular, τ_i is the input delay, and τ_{ij} is the communication delay. All the other consensus protocols previously introduced (discrete-time, second-order dynamics) can be modified in a similar way.

Regarding the single-integrator dynamics, a prominent result for continuous time systems is given in [33]. In the case of undirected connected topology and uniform delay across the entire swarm ($\tau_i = \tau_{ij} = \tau$), the robustness of the system to a relatively high time delay is inversely proportional to its convergence performance. Since a higher number of links inside an undirected graph implies faster convergence, there is a balance between the performance of the system and its robustness to time delays. Moreover, in [34], [35] it is shown that only the time delay affecting x_i in equation (8) is responsible for the stability of the system and must be lower than a specific upper bound. The presence of a delay affecting x_j only influences the convergence speed of the system, meaning that the consensus is reached more slowly but is eventually achieved [1].

In [36], an upper bound on the maximum allowable time delay is provided for the case of switching directed topology. However, all the topologies must be strongly connected for reaching consensus. Finally, [37] provides conditions for consensus with fixed directed topology and time-varying delays. The main contribution in [37] is that the topology only has to contain a spanning tree, and the delay can be unbounded under the assumption that the growth rate of the delay is limited.

The analysis of consensus protocols including time delays regarding second-order systems has been only recently addressed. In [38], an upper bound on the maximum time delay is provided for fixed directed topologies containing a spanning tree, considering the delay as fixed and uniform. In [39], a delay-decomposition approach is used to expand the analysis to time-varying delays, while in [40] switching topologies are also taken into account. Similar results can be found in [41], [42] for undirected graphs and in [43], [44] for generic linear systems.

IV. ROTARY-WING CONSENSUS APPLICATIONS

Multirotor UAVs are particularly suited to operate in confined and crowded spaces, due to their high maneuverability and their ability to hover in a fixed position [12]. Several applications [45] as delivery services, bridge inspection, and monitoring of road traffic are usually deployed in urban environments, where an adequate safety level of the operation must be maintained. A crucial requirement in such conditions is the capability of avoiding collisions between members of the swarm and crashes with external obstacles [46].

Following the thorough classification proposed in [47], collision/obstacle avoidance methodologies typically adopted in UAVs applications can be divided into four categories: geometric, force-fields, optimization-based, and sense-and-avoid. In the literature, consensus has been used typically for implementing collision/obstacle avoidance along with formation control strategies. The basic form of consensus algorithm does not take into account the possibility of collision between agents. Indeed, while generating a formation, two drones could collide while trying to reach their desired position [48]. A great effort has been made among researchers to formulate collision-free formation strategies through consensus.

A. Optimization-based Approaches

Optimization-based approaches encode the collision and obstacle avoidance problem by means of a cost function to be minimized over a finite time interval. Generally, trajectories associated with a high probability of collision yield a large value of the cost function.

In [49], the authors proposed an improved consensus algorithm able to convert the three Degrees Of Freedom (DOFs) of each drone, namely the yaw angle ϕ, the velocity in the xOy plane v_{xy}, and the vertical position z into the DOFs describing the relative positions among UAVs, i.e., the x, y and z positions. The obstacle avoidance is performed for both static and dynamic objects, classified either as sphere- or cylinder-shaped. The Particle Swarm Optimization (PSO) algorithm is employed locally in each agent to deal with static obstacles. It evaluates the fitness of the drone's states (ϕ, v_{xy}, z) with respect to a cost function penalizing large deviation from the current trajectory while ensuring a collision-free path. To tackle dynamic obstacles, the PSO is applied in the framework of the Distributed Model Predictive Control (DMPC), in order to predict the future relative position between the drone and the obstacle in the prediction horizon. If needed, the formation can be dissolved while avoiding dynamic obstacles. The topology is considered directed and a goal-seeking behaviour is adopted as the consensus protocol. Minimum adjustment strategies are employed to take into account the actuator constraints.

In [50], the consensus protocol itself is embedded inside the DMPC framework to compute the states of the drone in the following N_p time steps, where N_p is the length of the prediction horizon. The collision avoidance between UAVs is deployed as a penalty term added to the cost function of the DMPC. The relative distance between the predicted position of agent i and the reference position of its neighbors is defined as d_{ij}, while a safety distance is defined as D_s.The penalty term is inversely proportional to the difference between d_{ij} and D_s so that the cost function value drastically increases when two drones come dangerously close to each other. The method is applied to a directed topology with a leader-follower strategy.

In [51], a leader-less formation task is analysed under a directed topology. DMPC is used to generate locally the trajectory of each agent until the desired position computed through consensus is reached by every drone. Collision avoidance is achieved by adding an output constraint on the model, in terms of the relative distance between the predicted positions of agent i and the current position of neighbour agents $j \in \mathcal{N}_i$. In particular, this predictive distance must not become lower than a safety threshold.

B. Force Field Approaches

Force-field methods borrow the concept of attractive and repulsive force from electromagnetic theory. In particular, they consider the agent as a charged particle immersed in an Artificial Potential Field (APF), so that it is attracted by a target point and repelled by surrounding obstacles.

In [22], the authors introduced a smooth potential with a finite cut-off to manage both collision and obstacle avoidance. The finite cut-off is needed to activate both the behaviours only when a certain distance threshold is violated, thus making the strategy scalable. Collision avoidance is realized through a pairwise attractive/repulsive potential that drives each couple of agents (i, j) to a desired distance D_{ref} that corresponds to the minimum of the potential function. The gradient of the potential is simply added to the consensus protocol, and the result is an α-lattice formation. When an obstacle is detected, a repulsive potential is activated and added to the consensus protocol, to drive the agents to a safety distance D_s corresponding to the minimum of the potential function.

In [52], a similar approach was used and applied to directed graphs. In particular, a cyclic strongly connected topology, a directed spanning tree, and a topology containing a directed spanning tree were compared in terms of convergence performance. The simulations showed how a faster convergence could be reached with the strongly connected topology, while the other configurations yielded oscillations in control input and drones' position during the collision avoidance process.

Such oscillatory behaviour occurred also in [53], where an experimental verification on actual quadcopters is carried out to test the performance of a mixed MPC-APF approach. In particular, a repulsive potential function is formulated to ensure collision avoidance between the UAVs and added to the MPC cost function. In order to reduce the computational burden of the real-time operation, the potential function is linearly approximated. The experimental results showed how the control input during the collision avoidance process displayed an oscillatory behaviour, probably due to the fact that the agents repeatedly attracted and repelled each other because of the potential term of the cost function.

To attenuate these oscillations, the authors in [54] proposed an improved APF approach. In particular, the potential field computed in the previous iteration and the current potential field are summed according to a certain ratio so that the change in direction due to the collision avoidance process will not become too fast and cause jitter.

In [49], the authors embedded in the consensus protocol an additive term taking care of collision avoidance. In particular, the gradient of a potential function is activated when a certain safety threshold is violated. In such a case, the agent takes evasive actions in the vertical direction to avoid collisions without changing the motion of the UAV in xOy plane.

C. Geometric Approaches

Geometric approaches rely on the computation of geometric attributes to assure that the agent will not collide with any other member of the swarm or with an obstacle. Information as relative distance, bearing angle, or velocity are exploited.

In [55], consensus for an undirected connected topology is combined with a collision cone strategy for obstacle avoidance. The collision cone represents a set of velocity vectors that will cause the agent and obstacle to be on the collision course. The control signal must drive the drone in such a way that its velocity vector is aligned with the boundary of the collision cone and does not enter inside it.

D. Sense-and-avoid Approaches

Sense-and-avoid approaches are based on individual obstacle detection and avoidance by each drone in a swarm without knowledge of the plans of other drones. It is characterized by short response time and low required computational power, and it is particularly suitable for dynamic environments. To the best of the authors' knowledge, no consensus-based collision avoidance strategy was investigated adopting this approach [47], [56]. This may be due to the fact that consensus control needs significant information sharing at least among neighbors agents in the swarm, thus making the application of sense-and-avoid strategies inappropriate.

E. Discussion

In the previous subsections, several collision and obstacle avoidance methods have been described. The mentioned strategies present their own benefits and limitations that are now discussed and compared through the following metrics: the operational environment, the main operational requirement, and the progress in experimental validation of the methods.

From an operational point of view, optimization-based approaches are more suitable for static environments, in which the presence of obstacles is known in advance or detected by the whole formation. Force-field and geometric methods can tackle better dynamic environments, even though the application of the former could lead to local minima when the action of multiple artificial potential fields cancel each other. Also, the computation of the collision cone for the geometric approaches could be quite demanding in a crowded dynamic environment.

Optimization-based methods are usually employed when the swarm has a pre-defined reference trajectory to follow. Depending on the specific algorithm, each drone either knows the reference trajectories of all the other members of the swarm or only has its own pre-loaded path. Instead, force-field and geometric approaches usually rely more on inter-agent communication or relative sensing during the deployment of the mission.

Finally, it is worth noticing that very few of the mentioned methods were validated through the use of a real swarm. Optimization-based approaches are usually simulated through some dynamic model inserted into the frame of the optimization method employed, for instance, the MPC. Force-field approaches have been tested more in real platforms, even though the distributed sensing capability needed for the deployment of the algorithms is often replaced during flight tests by some centralized visual tracking method. Being more computationally demanding for a swarm, also geometric approaches lack actual flight tests and are usually simulated through a double-integrator dynamic model.

V. FIXED-WING CONSENSUS APPLICATIONS

Fixed-wing UAVs are frequently employed in operations as patrolling, surveillance or data collection in outdoor environments, where a crucial feature needed for the fulfilment of the mission is target tracking [12], [57]. Given a physical phenomenon, or a moving object to be tracked, deploying a network of UAVs with sensing capabilities can drastically reduce the observation noise of the process with respect to the performance of a single drone [58], [59]. This is why data fusion for sensor networks has received great attention over the last decade. The main results achieved in this direction focused on proposing distributed scalable versions of the Kalman filter exploiting the consensus theory [60]. The basic framework of the filter consists of a linear model of the process to be sensed, and a sensing model of each drone i, given by

$$x(k+1) = A(k)x(k) + B(k)w(k) \tag{9}$$

$$z_i(k) = H_i(k)x(k) + v_i(k), \tag{10}$$

where $x(k) \in \mathbb{R}^m$ and $w(k) \in \mathbb{R}^m$ are the state and input noise of the process, while $z_i(k) \in \mathbb{R}^p$ and $v_i(k) \in \mathbb{R}^p$ are the measured output of sensor i and the measurement noise affecting it. The initial state $x[0]$ is considered a random variable with normal distribution, while the input and measurement noises are considered white Gaussian noises satisfying $E[w(k)w(l)^T] = Q(k)\delta_{kl}$ and $E[v_i(k)v_i(l)^T] = R_i(k)\delta_{kl}$, where $\delta_{kl} = 1$ if $k = l$ and $\delta_{kl} = 0$ if $k \neq l$ [61].

Letting $Z(k)$ be the collective sensor data of the entire network at time k, it is possible to define

$$\begin{gathered} \hat{x}(k) = E(x(k)|Z(k)),\ \bar{x}(k) = E(x(k)|Z(k-1)) \\ M(k) = \Sigma(k|k),\ P(k) = \Sigma(k|k-1) \end{gathered}, \tag{11}$$

where $\hat{x}(k)$ and $\bar{x}(k)$ are the central estimate and prediction of state $x(k)$, while $M(k)$ and $P(k)$ are the estimation error covariance matrices of $\hat{x}(k)$ and $\bar{x}(k)$ respectively and their inverses are known as the information matrices.

The aim of distributed Kalman filter is to provide at each agent i a local estimate $\hat{x}_i(k)$ of the target state $x(k)$ equal to the central estimate $\hat{x}(k)$ exploiting only local information exchange among neighbors [61]. Several consensus-based methodologies have been developed. An in-depth survey in [60] classified the Kalman-like distributed filters in three main categories, depending on the quantity the system reaches consensus on. In particular, consensus can be reached on the sensor measurements, estimates, or information matrices.

A. Consensus on measurements

In [59], the authors showed that a distributed network of n micro-Kalman filters with p-dimensional measurement vectors observing an m-dimensional process are able to provide the same estimate as the one that a centralized Kalman filter with np-dimensional measurement vector would provide. Defining the following

$$S_i(k) = H_i^T(k)R_i^{-1}(k)H_i(k) \tag{12}$$

$$y_i(k) = H_i^T(k)R_i^{-1}(k)z_i(k) \tag{13}$$

as the innovation pair, [59] showed that if consensus is reached among the agents on these two time-varying quantities, the drones are able to provide an estimate identical to the one obtained through a central Kalman filter. Moreover, with respect to a centralized approach, the complexity of the problem in terms of the size of the matrices to be manipulated is reduced by a factor of n if a network of n sensors is employed. In [61], the approach was extended to consider sensors with different observation matrices H_i, so that the network acts as a collective observer of the process even if the target is not completely observable by each agent. However, the performance of this approach in terms of estimation error and cohesiveness of the estimates are quite poor [61].

B. Consensus on estimates

In [61], the authors introduced another approach to distributed target tracking. Similarly to the algorithm presented in the previous subsection, it relies on communication between neighbor agents of the innovation pair (12) and (13), but also of the current prediction $\bar{x}_i(k)$. Consensus is performed on the prediction $\bar{x}_i(k)$ during the estimation step of the filter, so that the estimate of each filter is driven towards the average estimate of the network. This is done to improve the accuracy of the estimation at each node. The authors showed the stability of the collective dynamics of the estimation errors without input noise, and the boundedness of the error covariance matrix. With respect to the consensus on measurements, this approach performs better in terms of estimation errors and provides more cohesive estimates. In [62], it is shown that computing the optimal gain of the Kalman filter based on consensus on estimates is not scalable in n, as it requires all-to-all communication. Thus, conditions for faster sub-optimal consensus are provided. In [63], the authors coupled the target tracking task to motion control. The network collaboratively performs an estimation of the state of a moving target and chases it through a protocol similar to (7). Clearly, the motion of the swarm affects the accuracy of the collected data since the information quality of a sensor increases as the agent gets closer to the target. These results were extended in [64], where input noise and network connectivity were taken into account. In [65], consensus is performed both during the estimation and the prediction step of the filter, using the previous predicted values of neighbors. This is done to make the system more robust against packet-loss considered as the stochastic failure of agent i to perform an observation at time k.

C. Consensus on information

A more recently developed approach is introduced in [66], where consensus has to be reached on the information matrix $(M_i(k))^{-1}$ and the information vector $(M_i(k))^{-1}\hat{x}_i(k)$. The authors showed that the estimation error is asymptotically bounded. The main contribution of [66] is that, unlike the previous approaches, the results hold also assuming the presence of unknown correlations between the measurements coming from different sensors. In [67], the results were adapted to the extended Kalman filter to include nonlinearities in the sensing model. Moreover, the communication noise between UAVs is considered in terms of signal-to-noise ratio and treated as an observation noise. In [68], random connection failures were taken into account by associating a probability to every communication link in the graph. A thorough stability analysis for different time-varying settings was conducted in [69], where also weaker conditions on the observability of the process were given. Finally, in [70], nonlinearities affecting both the sensing model and the process to be observed were considered, and sufficient conditions for the boundedness of the error covariance matrix were provided. Recently, some mixed approaches have been emerging in the literature. In [71], the authors proposed to make the agents reach consensus on the information matrix $(M_i(k))^{-1}$ and on the estimate $\hat{x}_i(k)$ after the estimation step of the filter. With respect to consensus on estimates, this method does not require the sharing of the innovation pair. Compared to consensus on information, the authors proved that in a steady-state situation, only the local estimates $\hat{x}_i(k)$ have to be shared, reducing the communication traffic to a minimum.

D. Discussion

The previous subsections provided an overview of three consensus-based Kalman filter paradigms for distributed target tracking. To summarize and discuss the presented methodologies, the following metrics are employed: the observability of the process to track, the correlation of measurements coming from different sensors and the cohesiveness of the provided estimates.

None of the three methods require the target to be completely observable by every agent. However, both consensus on measurements and estimates need that each set of neighbour agents is locally able to observe all the states of the target. Instead, consensus on information relies only on collective observability.

Both consensus on measurements and estimates assume that the measurements collected by each sensor are mutually independent. Instead, consensus on information considers the correlation among different sensors as unknown but present.

Finally, it is worth noticing how numerical simulations taking into account process noise showed that consensus on measurements methods lead to less cohesive estimates with respect to the other two strategies. This may indicate the need to share more than just the sensed data when a consistent process noise is present.

VI. CONCLUSIONS

In this work, an overview of consensus-based applications for multi-UAV systems is performed, classifying methodologies in function of the adopted platform. First, the basic notions of graph theory and consensus control are introduced for both first- and second-order linear dynamic systems. Then, formation control is discussed and the latest progress regarding switching topologies and time delays is also analysed. Next, rotary-wing applications are considered, focusing on collision and obstacle avoidance strategies. Optimization-based and force-field approaches appear to be the most promising area of interest. Future research areas could involve dynamic and nonsphere-shaped obstacles, different communication topologies to speed up the convergence rate or the relief of the oscillatory behaviour that appeared during flight tests. Finally, fixed-wing applications are investigated, considering target tracking. Consensus on information and mixed approaches have been recently developed and present the most appropriate feature for practical implementations. Future investigation could involve the coupling between these target tracking algorithms and UAVs motion dynamics.

REFERENCES

[1] Wei Ren, Randal W. Beard, "Distributed Consensus in Multi-vehicle Cooperative Control", Springer-Verlag London Limited, 2008.

[2] Olfati-Saber, J. A. Fax and R. M. Murray, "Consensus and Cooperation in Networked Multi-Agent Systems," in Proceedings of the IEEE, vol. 95, no. 1, pp. 215-233, Jan. 2007.

[3] Muhammad Majid Gulzar, Syed Tahir Hussain Rizvi, Muhammad Yaqoob Javed, Umer Munir, Haleema Asif, " Multi-Agent Cooperative Control Consensus: A Comparative Review", in Electronics vol. 7, pp. 2-22, 2018.

[4] J. Qin, Q. Ma, Y. Shi and L. Wang, "Recent Advances in Consensus of Multi-Agent Systems: A Brief Survey," in IEEE Transactions on Industrial Electronics, vol. 64, no. 6, pp. 4972-4983, June 2017.

[5] S. Djaidja and Q. Wu, "An overview of distributed consensus of multi-agent systems with measurement/communication noises," 2015 34th Chinese Control Conference (CCC), Hangzhou, pp. 7285-7290, 2015.

[6] L. Ding, Q. Han, X. Ge and X. Zhang, "An Overview of Recent Advances in Event-Triggered Consensus of Multiagent Systems," in IEEE Transactions on Cybernetics, vol. 48, no. 4, pp. 1110-1123, April 2018.

[7] K. Oh, M. Park, H. Ahn, "A survey of multi-agent formation control" in Automatica, vol. 53, pp. 424-440, March 2015.

[8] B. D. O. Anderson, C. Yu, B. Fidan and J. M. Hendrickx, "Rigid graph control architectures for autonomous formations," in IEEE Control Systems Magazine, vol. 28, no. 6, pp. 48-63, Dec. 2008.

[9] H. Sun, Y. Liu, F. Li and X. Niu, "A survey on optimal consensus of multi-agent systems," 2017 Chinese Automation Congress (CAC), Jinan, pp. 4978-4983, 2017.

[10] Carlos Perez-Montenegro, Matteo Scanavino, Nicoletta Bloise, Elisa Capello, Giorgio Guglieri, Alessandro Rizzo, "A Mission Coordinator Approach for a Fleet of UAVs in Urban Scenarios", in Transportation Research Procedia, vol. 35, pp. 110-119, 2018.

[11] Elisa Capello, Matteo Dentisa, Giorgio Guglieri, Laura Novaro Mascarello, Luca Spanó Cuomo, "An Innovative Cloud-based Supervision System for the Integration of RPAS in Urban Environments", in Transportation Research Procedia, vol. 28, pp. 191-200 , 2017.

[12] M. Hassanalian, A. Abdelkefi, "Classifications, applications, and design challenges of drones: A review", in Progress in Aerospace Sciences, vol. 91, pp. 99-131, May 2017.

[13] Donkyu Baek, Yukai Chen, Alberto Bocca, Alberto Macii, Enrico Macii, Massimo Poncino, "Battery-Aware Energy Model of Drone Delivery Tasks", in SLPED '18: Proceedings of the International Symposium on Low Power Electronics and Design, vol. 49, pp. 1-6, July 2018.

[14] Wei Ren, "Consensus strategies for cooperative control of vehicle formations", in IET Control Theory and Applications 1, vol.2 , pp. 505-512, April 2007.

[15] Anton V Proskurnikov, Ming Cao, "Consensus in Multi-Agent Systems", in Wiley Encyclopedia of Electrical and Electronics EngineeringPublisher: John Wiley and Sons, Inc.Editors: J. Webster, November 2016.

[16] C. Godsil, G. Royle, "Algebraic Graph Theory", in Graduate Texts in Mathematics. Berlin, Germany: Springer-Verlag, vol. 207, 2001.

[17] R. Merris, ”Laplacian matrices of a graph: A survey”, in Linear Algebra its Applications, vol. 197, pp. 143–176, 1994.

[18] Chai Wah Wu, ”Agreement and consensus problems in groups of autonomous agents with linear dynamics,” in 2005 IEEE International Symposium on Circuits and Systems, Kobe, pp. 292-295 Vol. 1, 2005.

[19] Wei Ren, Randal W. Beard, Timothy W. McLain, ”Coordination Variables and Consensus Building in Multiple Vehicle Systems”, in V. Kumar, N. E. Leonard, and A. S. Morse, editors, Cooperative Control, volume 309 of Lecture Notes in Control and Information Sciences, pages 171–188, 2005.

[20] Inomzhon Mirzaev, Jeremy Gunawardena, ”Laplacian Dynamics on General Graphs”, in Bulletin of Mathematical Biology, vol. 75, no. 11, September 2013.

[21] Wei Ren ”Consensus strategies for cooperative control of vehicle formations”, in IET Control Theory and Applications vol. 1, no. 2, pp. 505 - 512 , April 2007.

[22] R. Olfati-Saber, ”Flocking for multi-agent dynamic systems: algorithms and theory,” in IEEE Transactions on Automatic Control, vol. 51, no. 3, pp. 401-420, March 2006.

[23] J. Zhang, W. Wang, Z. Zhang, K. Luo and J. Liu, ”Cooperative Control of UAV Cluster Formation Based on Distributed Consensus,” 2019 IEEE 15th International Conference on Control and Automation (ICCA), Edinburgh, United Kingdom, 2019, pp. 788-793, 2019.

[24] J. Yu, C. Yan and M. Huang, ”Research of consistency problem for quadrotor UAV system with leader-follower,” 2019 Chinese Automation Congress (CAC), Hangzhou, China, 2019, pp. 616-621, 2019.

[25] S. Mukherjee and K. Namuduri, ”Formation Control of UAVs for Connectivity Maintenance and Collision Avoidance,” 2019 IEEE National Aerospace and Electronics Conference (NAECON), Dayton, OH, USA, pp. 126-130, 2019.

[26] D. Yu and C. L. Philip Chen, ”Relation-Invariable Persistent Formation Control with Collision Avoidance and Connectivity Preservation,” 2018 Chinese Automation Congress (CAC), Xi’an, China, pp. 2803-2808, 2018.

[27] Rui Ren, Yu-Yan Zhang, Xiao-Yuan Luo, Shao-Bao Li, ”Automatic generation of optimally rigid formations using decentralized methods” in International Journal of Automation and Computing, vol. 7, pp. 557–564, 2010.

[28] W. Yuan and G. Han, ”Consensus of Multi-Agent Systems With Piecewise Continuous Time-Varying Topology,” in IEEE Access, vol. 7, pp. 92048-92058, 2019.

[29] H. Wang, X. Liao and T. Huang, ”Consensus in multi-agent networks with switching topology and nonlinear dynamics,” 2017 Ninth International Conference on Advanced Computational Intelligence (ICACI), Doha, pp. 23-29, 2017.

[30] Dmitry Vengertsev, Hongkeun Kim, Jin Heon Seo, Hyungbo Shim, ”Consensus of output-coupled high-order linear multi-agent systems under deterministic and Markovian switching networks” in International Journal of Systems Science, vol. 46, no. 10, pp. 1-10, 2013.

[31] Hamed Rezaee, Farzaneh Abdollahi, ”Consensus Problem in General Linear Multiagent Systems Under Stochastic Topologies”, in IFAC-Papers OnLine, vol. 49, no. 13, pp. 13-18, 2016.

[32] H. Rezaee and F. Abdollahi, ”Consensus Problem Over High-Order Multiagent Systems With Uncertain Nonlinearities Under Deterministic and Stochastic Topologies,” in IEEE Transactions on Cybernetics, vol. 47, no. 8, pp. 2079-2088, Aug. 2017.

[33] R. Olfati-Saber and R. M. Murray, ”Consensus problems in networks of agents with switching topology and time-delays,” in IEEE Transactions on Automatic Control, vol. 49, no. 9, pp. 1520-1533, Sept. 2004.

[34] J. Yu and D. Teng, ”Weighted Group Consensus Analysis of Multi-agent Networks with Different Time Delays,” 2019 Chinese Control And Decision Conference (CCDC), Nanchang, China, 2019.

[35] Ji Liang-Hao, Liao Xiao-Feng, ”Consensus problems of first-order dynamic multi-agent systems with multiple time delays”, in Chinese Physics B 22(4):040203, April 2013.

[36] J. Zhang and Z. Wang, ”Average consensus of multi-agents with directed switching network and time-varying communication delays,” 2017 Chinese Automation Congress (CAC), pp. 5067-5072, Jinan, 2017.

[37] S. Zong and Y. -P. Tian, ”Consensus of discrete-time multi-agent systems with unbounded time-varying delays,” 2020 39th Chinese Control Conference (CCC), pp. 4924-4929, Shenyang, China, 2020.

[38] Wenwu Yu, Guanrong Chena, Ming Cao, ”Some necessary and sufficient conditions for second-order consensus in multi-agent dynamical systems”, in Automatica, vol. 46, no. 6, Pages 1089-1095, June 2010.

[39] X. Zhang, B. Cui and Z. Jiang, ”A delay-decomposition approach to consensus of multi-agent network with time-varying delay,” Proceeding of the 11th World Congress on Intelligent Control and Automation, pp. 4830-4834, Shenyang, China, 2014.

[40] C. Chen, Z. Wang, L. Jin, Y. Yin and F. Liu, ”Consensus of second-order multi-agent systems with time delays,” 2018 3rd International Conference on Control and Robotics Engineering (ICCRE), pp. 109-113, Nagoya, 2018.

[41] P. Lin and Y. Jia, ”Consensus of a Class of Second-Order Multi-Agent Systems With Time-Delay and Jointly-Connected Topologies,” in IEEE Transactions on Automatic Control, vol. 55, no. 3, pp. 778-784, March 2010.

[42] H. Wu, B. An and Y. Song, ”On the General Consensus Protocol for Second-order Multi-agent Systems with Diverse Time Delays,” 2018 37th Chinese Control Conference (CCC), pp. 6394-6397, Wuhan, 2018.

[43] X. Luo, X. Li, S. Li and X. Guan, ”Consensus for linear multi-agent systems with constant and time-varying communication delays via delay-decomposition approach,” Proceeding of the 11th World Congress on Intelligent Control and Automation, , pp. 353-358, Shenyang, China, 2014.

[44] J. Wanga, Z. Zhou, C. Wanga, Z. Ding, ”Cascade structure predictive observer design for consensus control with applications to UAVs formation flying”, in Automatica, vol. 121, November 2020.

[45] H. Shakhatreh et al., ”Unmanned Aerial Vehicles (UAVs): A Survey on Civil Applications and Key Research Challenges,” in IEEE Access, vol. 7, pp. 48572-48634, 2019.

[46] Huahua Liu, Hugh H. T. Liu, Cheng Chi, Yawei Zhai, Xingqun Zhan, ”Navigation information augmented artificial potential field algorithm for collision avoidance in UAV formation flight” in Aerospace Systems, vol. 3, pp. 229–241, 2020.

[47] J. N. Yasin, S. A. S. Mohamed, M. Haghbayan, J. Heikkonen, H. Tenhunen and J. Plosila, ”Unmanned Aerial Vehicles (UAVs): Collision Avoidance Systems and Approaches,” in IEEE Access, vol. 8, pp. 105139-105155, 2020.

[48] Hu JinWen, WANG Man, Zhao ChunHui, Pan Quan, Du Chang, ”Formation control and collision avoidance for multi-UAV systems based on Voronoi partition”, in Science China Technological Sciences, vol. 63, pp. 65–72, 2020.

[49] Yu Wuab, Jinzhan Goua, Xinting Huc, Yanting Huang, ”A new consensus theory-based method for formation control and obstacle avoidance of UAVs”, in Aerospace Science and Technology, vol. 107, 106332, December 2020.

[50] Y. Kuriki and T. Namerikawa, ”Formation control with collision avoidance for a multi-UAV system using decentralized MPC and consensus-based control,” 2015 European Control Conference (ECC), Linz, Austria, pp. 3079-3084, 2015.

[51] S. Dubay and Y. Pan, ”Distributed MPC based Collision Avoidance Approach for Consensus of Multiple Quadcopters,” 2018 IEEE 14th International Conference on Control and Automation (ICCA), Anchorage, AK, pp. 155-160, 2018.

[52] Q. Gong, C. Wang, Z. Qi and Z. Ding, ”Gradient-based collision avoidance algorithm for second-order multi-agent formation control,” 2017 36th Chinese Control Conference (CCC), Dalian, pp. 8183-8188, 2017.

[53] K. Yamamoto, K. Sekiguchi and K. Nonaka, ”Experimental verification of formation control by model predictive control considering collision avoidance in three dimensional space with quadcopters,” 2017 11th Asian Control Conference (ASCC), Gold Coast, QLD, pp. 1602-1607, 2017.

[54] Yang Huang, Jun Tang, Songyang Lao, ”UAV Group Formation Collision Avoidance Method Based on Second-Order Consensus Algorithm and Improved Artificial Potential Field”, in Symmetry 11(9):1162, 2019.

[55] S. Akkaya, O. Akbati and A. F. Ergenc, ”Consensus Control of Mobile Agents with Obstacle Avoidance using Collision Cone Approach,” 2018 6th International Conference on Control Engineering and Information Technology (CEIT), Istanbul, Turkey, pp. 1-5, 2018.

[56] Xiang Yu, Youmin Zhang, ”Sense and avoid technologies with applications to unmanned aircraft systems: Review and prospects”, in Progress in Aerospace Sciences, vol. 74, pp. 152-166, April 2015.

[57] M. Tagir, M. Rustem, ”Multi-UAV Cooperative Target Tracking via Consensus-based Guidance Vector Fields and Fuzzy MRAC”, TechRxiv, Preprint, https://doi.org/10.36227/techrxiv.13537313.v1, 2021.

[58] R. Olfati-Saber and J. S. Shamma, "Consensus Filters for Sensor Networks and Distributed Sensor Fusion," Proceedings of the 44th IEEE Conference on Decision and Control, Seville, Spain, pp. 6698-6703, 2005.
[59] R. Olfati-Saber, "Distributed Kalman Filter with Embedded Consensus Filters," Proceedings of the 44th IEEE Conference on Decision and Control, Seville, Spain, pp. 8179-8184, 2005.
[60] W. Li, Z. Wang, G. Wei, L. Ma, J. Hu, D. Ding, "A Survey on Multisensor Fusion and Consensus Filtering for Sensor Networks", in Discrete Dynamics in Nature and Society 2015, (Article ID 683701):12, October 2015.
[61] R. Olfati-Saber, "Distributed Kalman filtering for sensor networks," 2007 46th IEEE Conference on Decision and Control, New Orleans, LA, USA, pp. 5492-5498, 2007.
[62] R. Olfati-Saber, "Kalman-Consensus Filter : Optimality, stability, and performance," Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference, Shanghai, China, pp. 7036-7042, 2009.
[63] R. Olfati-Saber and P. Jalalkamali, "Coupled Distributed Estimation and Control for Mobile Sensor Networks," in IEEE Transactions on Automatic Control, vol. 57, no. 10, pp. 2609-2614, Oct. 2012.
[64] Rui Zhou, Yu Feng, Bin Di, Jiang Zhao, Yan Hu, "Multi-UAV cooperative target tracking with bounded noise for connectivity preservation", in Frontiers of Information Technology and Electronic Engineering, vol. 21, pp. 1494–1503, 2020.
[65] S. Fan, H. Yan, H. Zhang, M. Wang and X. Zhan, "Prediction consensus-based distributed Kalman filtering with packet loss," 2017 36th Chinese Control Conference (CCC), Dalian, pp. 7950-7955, 2017.
[66] Giorgio Battistelli, Luigi Chisci, Stefano Morrocchi, Francesco Papi, "An Information-Theoretic Approach to Distributed State Estimation", in IFAC Proceedings Volumes, vol. 44, Issue 1, Pages 12477-12482, January 2011.
[67] Xiaowei Fu, Kunpeng Liu, Xiaoguang Gao, "Multi-UAVs Communication-Aware Cooperative Target Tracking", in Applied Sciences 8(6):870, May 2018.
[68] Q. Liu, Z. Wang, X. He and D. H. Zhou, "On Kalman-Consensus Filtering With Random Link Failures Over Sensor Networks," in IEEE Transactions on Automatic Control, vol. 63, no. 8, pp. 2701-2708, Aug. 2018
[69] G. Wei, W. Li, D. Ding and Y. Liu, "Stability Analysis of Covariance Intersection-Based Kalman Consensus Filtering for Time-Varying Systems," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 50, no. 11, pp. 4611-4622, Nov. 2020.
[70] Z. Zhang, X. Dong, Q. Tan, Y. Liang, Q. Li and Z. Ren, "Distributed Extended Kalman-consensus Filtering Algorithm Based on Sensor Network for Nonlinear System," 2018 37th Chinese Control Conference (CCC), Wuhan, China, pp. 6977-6982, 2018.
[71] S. P. Talebi and S. Werner, "Distributed Kalman Filtering and Control Through Embedded Average Consensus Information Fusion," in IEEE Transactions on Automatic Control, vol. 64, no. 10, pp. 4396-4403, Oct. 2019.