



# A Survey on Swarming With Micro Air Vehicles: Fundamental Challenges and Constraints

Mario Coppola<sup>1,2\*</sup>, Kimberly N. McGuire<sup>1</sup>, Christophe De Wagter<sup>1</sup> and Guido C. H. E. de Croon<sup>1\*</sup>

<sup>1</sup> Micro Air Vehicle Laboratory (MAVLab), Department of Control and Simulation, Faculty of Aerospace Engineering, Delft University of Technology, Delft, Netherlands, <sup>2</sup> Department of Space Systems Engineering, Faculty of Aerospace Engineering, Delft University of Technology, Delft, Netherlands

## OPEN ACCESS

### Edited by:

Eliseo Ferrante,  
Vrije Universiteit Amsterdam,  
Netherlands

### Reviewed by:

Titus Cieslewski,  
University of Zurich, Switzerland  
Tomas Krajnik,  
Czech Technical University in Prague,  
Czechia

### \*Correspondence:

Mario Coppola  
m.coppola@tudelft.nl  
Guido C. H. E. de Croon  
g.c.h.e.decroon@tudelft.nl

### Specialty section:

This article was submitted to  
Multi-Robot Systems,  
a section of the journal  
Frontiers in Robotics and AI

**Received:** 18 November 2019

**Accepted:** 04 February 2020

**Published:** 25 February 2020

### Citation:

Coppola M, McGuire KN, De Wagter C and de Croon GCH (2020) A Survey on Swarming With Micro Air Vehicles: Fundamental Challenges and Constraints. *Front. Robot. AI* 7:18. doi: 10.3389/frobt.2020.00018

This work presents a review and discussion of the challenges that must be solved in order to successfully develop swarms of Micro Air Vehicles (MAVs) for real world operations. From the discussion, we extract constraints and links that relate the local level MAV capabilities to the global operations of the swarm. These should be taken into account when designing swarm behaviors in order to maximize the utility of the group. At the lowest level, each MAV should operate safely. Robustness is often hailed as a pillar of swarm robotics, and a minimum level of local reliability is needed for it to propagate to the global level. An MAV must be capable of autonomous navigation within an environment with sufficient trustworthiness before the system can be scaled up. Once the operations of the single MAV are sufficiently secured for a task, the subsequent challenge is to allow the MAVs to sense one another within a neighborhood of interest. Relative localization of neighbors is a fundamental part of self-organizing robotic systems, enabling behaviors ranging from basic relative collision avoidance to higher level coordination. This ability, at times taken for granted, also must be sufficiently reliable. Moreover, herein lies a constraint: the design choice of the relative localization sensor has a direct link to the behaviors that the swarm can (and should) perform. Vision-based systems, for instance, force MAVs to fly within the field of view of their camera. Range or communication-based solutions, alternatively, provide omni-directional relative localization, yet can be victim to unobservable conditions under certain flight behaviors, such as parallel flight, and require constant relative excitation. At the swarm level, the final outcome is thus intrinsically influenced by the on-board abilities and sensors of the individual. The real-world behavior and operations of an MAV swarm intrinsically follow in a bottom-up fashion as a result of the local level limitations in cognition, relative knowledge, communication, power, and safety. Taking these local limitations into account when designing a global swarm behavior is key in order to take full advantage of the system, enabling local limitations to become true strengths of the swarm.

**Keywords:** swarm, challenges, review, robustness, autonomous, micro air vehicles, drones, MAV

## 1. INTRODUCTION

Micro Air Vehicles (MAVs), or “small drones,” are becoming commonplace in the modern world. The term refers to small, light-weight, flying robots. Several MAV designs exist, including multirotors (Kumar and Michael, 2012), flapping wing (Michelson and Reece, 1998; Wood et al., 2013; de Croon et al., 2016), fixed wing (Green and Oh, 2006), morphing designs (Falanga et al., 2019b), or “hybrid” vehicles (Itasse et al., 2011). Of these, quadrotors have enjoyed the spotlight due to their high maneuverability, their ability to take-off vertically (as opposed to most fixed wing MAVs, for instance), and their relative simplicity in design (Gupte et al., 2012; Kumar and Michael, 2012). MAVs can be used for surveillance and mapping (Mohr and Fitzpatrick, 2008; Scaramuzza et al., 2014; Saska et al., 2016b), infrastructure inspection (Sa and Corke, 2014), load transport and delivery (Palunko et al., 2012), or construction (Lindsey et al., 2012; Augugliaro et al., 2014). Such applications are particularly useful in areas that are not easily accessible by humans, like forests or disaster sites (Alexis et al., 2009; Achtelik et al., 2012). Smaller and lighter designs push the boundaries of their applications further. Aside from the asset of increased portability, smaller MAVs can also navigate through tighter spaces, such as narrow indoor environments with higher agility (Mohr and Fitzpatrick, 2008). They also cause less damage to their surroundings (including people) in the event of a collision, making them intrinsically safer tools (Kushleyev et al., 2013).

Unfortunately, smaller size comes at the expense of more limited capabilities. The interplay between limited flight time, limited sensing, and limited power hinder an MAV from performing grander tasks on its own. This has created a strong interest in developing MAV *swarms* (Yang et al., 2018). The paradigm of swarm robotics aims to transcend the limitations of a single robot by enabling cooperation in larger teams. This is inspired by the animal kingdom, where animals and insects have been observed to unite forces toward a common goal that is otherwise too complex or challenging for the lone individual (Garnier et al., 2007). Using several robots at once can bring several advantages and possibilities, such as: redundancy, faster task completion due to parallelization, or the execution of collaborative tasks (Martinoli and Easton, 2003; Trianni and Campo, 2015; Nedjah and Junior, 2019). The control of robotic swarms is envisioned to be fully distributed. The individual robots perceive and process their environment locally and then act accordingly without global awareness or direct awareness of the final goal of the swarm. Nevertheless, by means of collaboration, the robots can achieve an objective that they would not have been able to achieve by themselves. As they say: there is strength in numbers.

It is easy to imagine swarms of MAVs jointly carrying a load that is too heavy for a single one to lift, or persistently exploring an area without interruption. As is often the case, however, putting such visions into practice is another story altogether. Developing self-organizing swarms of MAVs in the real world is a multi-disciplinary challenge coarsely divided in two main aspects. One aspect is that of the individual MAV design, where the local abilities of a single MAV are defined. The second aspect is the swarm design, whereby we need to develop controllers with

which the global goal can be efficiently achieved, autonomously, by the swarm. To make matters more complicated, the two are not decoupled. As we shall explore in this paper, there exist fundamental links between the local limitations of an MAV and the behaviors that a swarm of MAVs could, or should, execute as a result. Vice versa, in order to realize certain swarm behaviors, there are local requirements that the individual MAVs must meet. This bond between the local and the global cannot be ignored if MAV swarms are to be brought to the real world. In this paper, we aim to reconcile these two aspects and present a discussion of the fundamental challenges and constraints linking local MAV properties and global swarm behaviors.

## 2. CO-DEPENDENCE OF SWARM DESIGN AND INDIVIDUAL MAV DESIGN

Let us begin from the primary challenge of swarm robotics: to design local controllers that successfully lead to global swarm behaviors (Şahin et al., 2008). Concerning MAVs, these global behaviors include, but are not limited to: collaborative transport, collaborative construction, distributed sensing, collaborative object manipulation, and parallelized exploration and mapping of environments. Albeit the individual MAV may be limited in its ability to successfully perform these tasks (for instance, as areas get larger or loads get heavier), they can be tackled by collaborating in a swarm. Generally, swarms of robots are expected to feature the following inherent advantages (Şahin et al., 2008; Brambilla et al., 2013):

- **Robustness:** The swarm is robust to the loss or failure of individual robots.
- **Flexibility:** The swarm can reconfigure to tackle different tasks.
- **Scalability:** The swarm can grow and shrink in size depending on the needs of the global task.

When designing a swarm of MAVs, we must then ask ourselves: how can we design a swarm that is robust, flexible, and scalable? It is true that these properties pertain to the swarm rather than the individual, but if the swarm is composed of individual units, then it follows that they must also be present (although perhaps not always apparent) at the local level. We cannot use individual robots that are not robust and merely expect the swarm as a whole to be immune or tolerant to individual failures (Bjerknes and Winfield, 2013). If there is a high probability of errors at the local level, such as erroneous observations, poorly executed commands, or failure of a unit, then this may have a repercussion on the swarm’s performance; an effect that Bjerknes and Winfield (2013) have shown can worsen with the number of robots in a swarm. There is a point after which the individual robots are too unreliable and the swarm can fail to achieve its goal, or it can be shown to be outperformed by smaller teams with more reliable units (Stancliff et al., 2006) or even by a single reliable system (Engelen et al., 2014). The further complication with MAVs is that local failures do not remain local, but are likely to cause collisions and damages to other nearby MAVs and/or objects. For some tasks, such as collective transport, the impact may be even more severe as the MAVs are mechanically attached to the

load (Tagliabue et al., 2019). It thus follows that, to develop a robust swarm for real world deployment, we must also ensure robustness at the local level.

Of equal importance is to make sure that the robots have the satisfactory tools and sensors to carry out their individual components of a global task. The more capable the sensors are, the more likely it is that the swarm can be flexible and adjust to different tasks or unexpected changes. When performing pure swarm intelligence research, we can afford to abstract away from lower level issues (Brutschy et al., 2015). For instance, in a study on making a decision about selecting a new location for a swarm's nest, one can abstract away from actually evaluating the quality of a nest location, and instead focus the analysis on a particular aspect of the system, such as the decision making process. However, when dealing with real-world applications, this is not an option. If we want to develop nest selection capabilities for a swarm in the real world, each robot should be capable of: flying and operating safely, recognizing the existence of a site, evaluating the quality of a site with a certain reliability, exchanging this information with its neighbors, and more. All these lower level requirements need to be appropriately realized for the global level outcome to emerge, or otherwise need to be accepted as limitations of the system. The way in which they are implemented shape the final behavior of the swarm.

Last but not least, unless properly accounted for, there are scalability problems that may also occur as the swarm grows in size. Examples of issues are: a congested airspace whereby the MAVs are unable to adhere to safety distances, a cluttered visual environment as a result of several MAVs (thus obstructing the task), or poor connectivity as a result of low-range communication capabilities. To achieve scalability, the MAV design must be such that these properties are appropriately accommodated, from the appropriate hardware design all the way to the higher level controllers which make up the swarm behavior.

## 2.1. The Challenge of Local Sensing and Control

When flying several MAVs at once, the control architecture can be of two types: (1) centralized, or (2) decentralized. In the centralized case, all MAVs in a swarm are controlled by a single computer. This “omniscient” entity knows the relevant states of all MAVs and can (pre-)plan their actions accordingly. The planning can be done a-priori and/or online. In the decentralized case, the MAVs make their decisions locally.

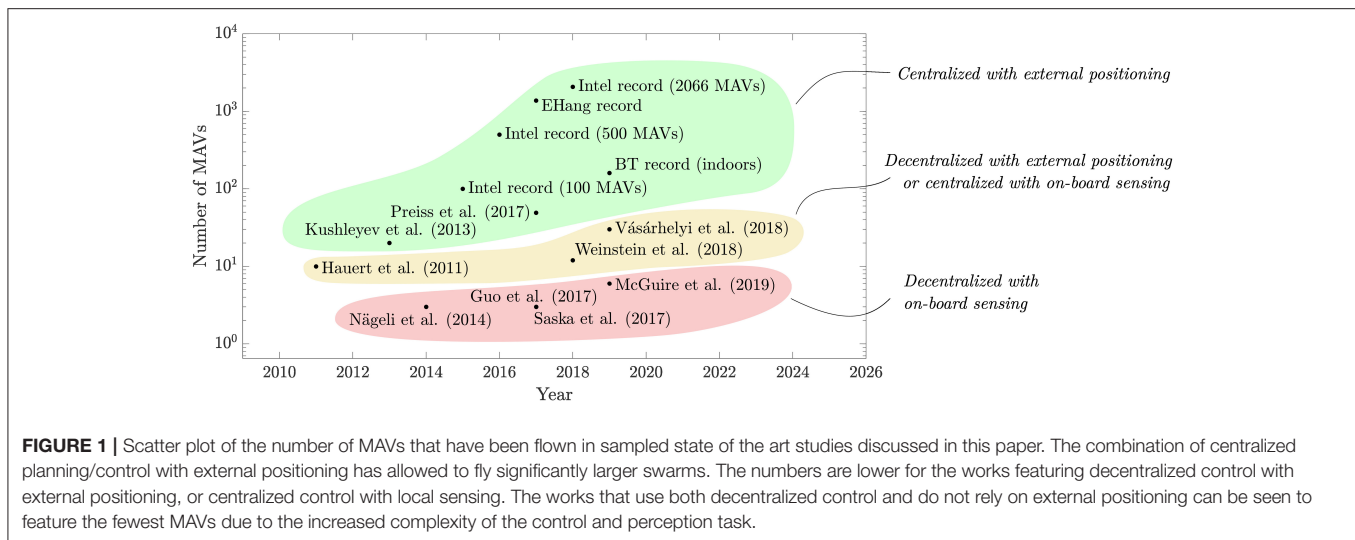
A second dichotomy can also be defined for how the MAVs sense their environment: (1) using external position sensing, or (2) locally. External positioning is typically achieved with a Global Navigation Satellite System (GNSS) or with a Motion Capture System (MCS), depending on whether the MAVs are flying outdoors or indoors, respectively. Alternatively, the latter only relies on the sensors that are on-board of the MAV.

Currently, the combination of centralized architecture and external positioning have achieved the highest stage of maturity, allowing for flights with several MAVs. Kushleyev et al. (2013)

showed a swarm of 20 micro quadrotors that could re-organize in several formations. Lindsey et al. (2012), Augugliaro et al. (2014), and Mirjan et al. (2016) developed impressive collaborative construction schemes using a team of MAVs. Preiss et al. (2017) showcased “Crazyswarm,” an indoor display of 49 small quadrotors flying together. The strategy of centralized planning and external positioning has also attracted large industry investments, leading to shows with record-breaking number of MAVs flying simultaneously. In 2015, Intel and Ars Electronica Futurelab first flew 100 MAVs, making a Guinness World Record (Swatman, 2016a). In 2016, Intel beat its own record by flying 500 MAVs simultaneously (Swatman, 2016b). In 2018, EHang claimed the record with 1,374 MAVs flying above the city of Xi'an, China (Cadell, 2018). In 2019, Intel reclaimed the title by flying 2,066 MAVs (Guinness World Records, 2018) outdoors. Meanwhile, the record for the most MAVs flying indoors (from a single computer) was recently broken by BT with 160 MAVs (Guinness World Records, 2019).

Without external positioning systems or centralized planning/control, the problem of flying several MAVs at once becomes more challenging. This is because: (1) the MAVs have to rely only on on-board perception, or (2) they have to make local decisions without the benefit of global planning, or (3) both. It is then not surprising that, as shown in **Figure 1**, the swarms that have been flown without external positioning and/or centralized control are significantly smaller. When the control is decentralized, but the MAVs benefit from an external positioning system, or vice versa, the largest swarms are in the dozens (Hauert et al., 2011; Vásárhelyi et al., 2018; Weinstein et al., 2018). For swarms featuring both local perception and distributed control, the highest numbers are currently in the single digits (Nägeli et al., 2014; Guo et al., 2017; Saska et al., 2017; McGuire et al., 2019). Despite the fact that these numbers have been increasing in the last few years, they are still lower, as the operations are shifted away from external system and toward on-board perception and control. If the past is any indication for the future, we expect that: (1) the numbers of drones will keep increasing for all cases, and (2) businesses will take over the records as the technologies for on-board decision making and perception become more mature.

Although we can fly a high number of MAVs when using centralized planning and external positioning, swarming is not just a numbers game. Flying with many MAVs does not automatically imply that we are achieving the benefits of swarm robotics (Hamann, 2018). A centralized system relies on a main computer to take all decisions. This means that a prompt online re-planning is needed in order to achieve robustness and flexibility. This re-planning grows in complexity with the size of the swarm, making the system unscalable. Moreover, the central computer represents a single point of failure. Instead, a swarm adopts a distributed strategy whereby each robot takes a decision independently. The fact that each MAV needs to take its own decisions, and additionally, if the MAVs do not rely on external infrastructure, introduces a new layer of difficulty. However, this is also what brings new advantages: redundancy, scalability, and



adaptability to changes (Şahin, 2005; Bonabeau and Théralaz, 2008)<sup>1</sup>.

When we analyze swarms of MAVs with local on-board sensing and control, we can observe two trends: (1) As the size of the swarm increases, the *relative* knowledge that each MAV will have of its global environment, which includes the remainder of the swarm, decreases (Bouffanais, 2016); (2) as the individual MAV's size and/or mass decreases, its capability to sense its own local environment decreases (Kumar and Michael, 2012). This creates an interesting challenge. On the one hand, we aim to design smaller, lighter, cheaper, and more efficient MAVs. On the other hand, as we make these MAVs smaller, the gap between the microscopic and macroscopic widens further. Designing the swarm becomes a more challenging task because each MAV has less information about its environment and is also less capable to act on it. This can be generalized to other robotic platforms as well, but MAVs feature the increased difficulty of having a tightly bound relationship between their on-board capabilities, their dynamics, their processing power, and their sensing (Chung et al., 2018). This is sometimes referred to as the SWaP (Size, Weight, and Power) trade-off (Mahony et al., 2012; Liu et al., 2018). The relationship is often non-linear. For instance, if we add a sensor that results in 5% more power usage, it does not only spend more energy per second, but it also affects the total energy that can be extracted from the battery as it will be operating in a different regime (de Croon et al., 2016). For many MAVs, grams and milliwatts matter. This makes the design of autonomous decentralized swarms of MAVs a more unique challenge.

## 2.2. Overview of Design Challenges Throughout the Design Chain

Throughout this paper, we shall review the state of the art in MAV technology from the swarm robotics perspective. To facilitate

<sup>1</sup>Of course, flying several MAVs with a centralized controller has its own challenges, which we do not mean to undermine. We only mean that it appears that these methods are at a more mature stage with respect to self-organized approaches, which is the focus of this article.

our discussion, we will break down the challenges for the design and control of an MAV swarm in the following four levels, from “local” to “global.”

1. MAV design. This defines the processing power, flight time, dynamics, and capabilities of the single MAV. Most importantly from a swarm engineering perspective, it defines the sensory information available on each unit, from which it can establish its view of the world. This is discussed in section 3.
2. Local ego-state estimation and control. At the lowest level, an MAV must be capable of controlling its motion with sufficient accuracy. This lower level layer handles basic flight operations of the MAV. This includes attitude control, height control, and velocity estimation and control. Moreover, the MAV should be capable of safely navigating in its environment. Minimally, it should detect and avoid potential obstacles. The challenges and state of the art for these methods are discussed in section 4.
3. Intra-swarm relative sensing and avoidance. There are two key enabling technologies for swarming. The first is the knowledge on (the location of) nearby neighbors. This is particularly important for MAV swarms as it not only enables several higher level swarming behaviors, but it also ensures that MAVs do not collide with one another in mid-air. The second enabling technology is communication between MAVs, such that they can share information and thus expand their knowledge of the environment via their neighborhood. These are discussed in section 5.
4. Swarm behavior. This is the higher level control policy that the robots follow to generate the global swarm behavior. Examples of higher level controllers in swarms range from attraction and repulsion forces for flocking (Gazi and Passino, 2002; Vászárhelyi et al., 2018) to neural networks for aggregation, dispersion, or homing (Duarte et al., 2016). We discuss how MAV swarm behaviors can be designed in section 6.

Other similar taxonomies have been defined. Floreano and Wood (2015) describe three levels of robotic cognition:

sensory-motor autonomy, reactive autonomy, and cognitive autonomy. Meanwhile, de Croon et al. (2016) divide the control process for autonomous flight into four levels: attitude control, height control, collision avoidance, and navigation. Although the taxonomies above are conceptually similar (generally going from low level sensing and control to a higher level of cognition), the re-definition that we provide here is designed to better organize our discussion within the context of swarm robotics. Moreover, we also include the design of the MAV within the chain. As we will explain in this manuscript, this has a fundamental impact on the higher level layers.

The four stages that have been defined have an increasing level of abstraction. The lower levels enable the robustness, flexibility, and scalability properties expected at the higher level, while the higher levels dictate, accommodate, and make the most out of the capabilities set at the lower level. From a more systems perspective, the MAV design poses constraints on what the higher level controllers can expect to achieve, while the higher level controllers create requirements that the MAV must be able to fulfill. A simplified view of the flow of requirements and constraints is shown in **Figure 2**.

Throughout the remainder of this paper, as we discuss the state of the art at each level, we will highlight the major constraints that flow upwards and the requirements that flow downwards. Naturally, each sub-topic that we will treat features a plethora of solutions, challenges, and methods, each deserving of a review paper of its own. It is beyond the scope (and probably far beyond any acceptable word limit, too) to present an exhaustive review about each topic. Instead, we keep our focus to highlighting the main methodologies and how they can be used to design swarms of MAVs. Where possible, we will refer the reader to more in-depth reviews on a specific topic.

### 3. MAV DESIGN

The differentiating challenge faced by a flying robot, namely (and somewhat trivially) the fact that it has to carry its own mass around, creates a strong design driver toward minimalism. Despite battery mass consisting of up to 20–30% of the total system mass, the flight time of quadrotor MAVs still remains limited to the order of magnitude of minutes (Kumar and Michael, 2012; Mulgaonkar et al., 2014; Oleynikova et al., 2015). To increase the carrying capabilities of an MAV, enabling it to carry more/better sensors, processors, or actuators, while keeping flight time constant, means that the size of the battery should also increase. In turn, this leads to a new increase in mass, and so on. This type of spiral, often referred to as the “snowball effect,” is a well-known issue for the design of any flying vehicle, from MAVs to trans-Atlantic airliners (Obert, 2009; Lammering et al., 2012; Voskuil et al., 2018). It then becomes paramount for an MAV design to be as minimalist as possible relative to its task, such that it may fulfill the mission requirements with a minimum mass (or, at the very least, there is a trade-off to be considered). This design driver has been taken to the extreme and has led to the development of miniature MAV systems, popular examples of which include the Ladybird drone and the Crazyfly (Lehnert

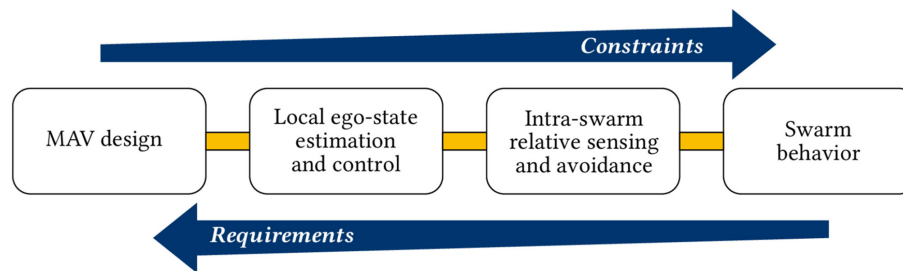
and Corke, 2013; Remes et al., 2014; Giernacki et al., 2017). These MAVs have a mass of <50 g, making them attractive due to their low cost and the fact that they are safer to operate around people. This makes them appealing for swarming, especially in indoor environments (Preiss et al., 2017).

A substantial body of literature already exists on single MAV design, the specifics of which largely vary depending on the type of MAV in question. We refer the reader to the works of Mulgaonkar et al. (2014) and Floreano and Wood (2015) and the sources therein for more details. From the swarming perspective, it is important to understand that, independently of the type of MAV in question, the following constraints are intertwined during the design phase: (1) flight time, (2) on-board sensing, (3) on-board processing power, and (4) dynamics. This means that the choice of MAV directly constrains the application as well as the swarming behavior that can be achieved (or, vice versa, a desired swarming behavior requires a specific type of MAV). For example, fixed wing MAVs benefit from longer autonomy. This makes them ideal candidates for long term operations, and also give the operators more time to launch an entire fleet and replace members with low batteries (Chung et al., 2016). However, fixed wing MAVs also have limited agility in comparison to quadrotors or flapping wing MAVs. The latter, for instance, can have a very high agility (Karásek et al., 2018), but also comes with more limited endurance and payload constraints (Olejnik et al., 2019). The MAV design impacts the number and type of sensors that can be taken on-board. It can also impact how these sensors are positioned and their eventual disturbances and noise. In turn, this affects the local sensing and control properties of the MAV and can also impact its ability to sense neighbors and operate in a team more effectively. We will return to this where relevant in the next chapters, whereby we discuss how an MAV can estimate and control its motion, sense its neighbors, and navigate in an environment together with the rest of the swarm.

A special note is made to designs that are *intended* for collaboration. Oung and D’Andrea (2011) introduced the Distributed Flight Array, a design whereby multiple single rotors can attach and detach from each other to form larger multi-rotors. More recently, Saldaña et al. (2018) introduced the ModQuad: a quadrotor with a magnetic frame designed for self-assembly with its neighbors. This design provides a solution for collaborative transport by creating a more powerful rigid structure with several drones. Gabrich et al. (2018) have shown how the ModQuad design can be used to form an aerial gripper. Because of the frame design, one of the difficulties of the ModQuad was in the disassembly back to individual quadrotors. This was tackled with a new frame design which enabled the quadrotors to disassemble by moving away from each other with a sufficiently high roll/pitch angle (Saldaña et al., 2019).

### 4. LOCAL EGO-STATE ESTIMATION AND CONTROL

The primary objective for a single MAV operating in a swarm is to remain in flight and perform higher level tasks with a



**FIGURE 2** | Generalized depiction of the flow of requirements and constraints for the design of MAV swarms. The lower level design choices create constraints on the higher level properties of the swarm. Higher level design choices create requirements for the lower levels, down to the physical design of the MAV. Note that, for the specific case, the flow of requirements and constraints is likely to be more intricate than this picture makes it out to be. However, the general idea remains.

given accuracy. This requires a robust estimation of the on-board state as well as robust lower level control, preferably while minimizing the size, power, and processing required. The design choices made here dictate the accuracy (i.e., noise, bias, and disturbances) with which each MAV will know its own state, as well as which variables the state is actually comprised of. In turn, this affects the type of maneuvers and actions that an MAV can execute. For instance, aggressive flight maneuvers likely require relatively accurate real-time state estimation (Bry et al., 2015). Of equal importance are the considerations for the processing power that remains for higher level tasks. While it can be attractive to implement increasingly advanced algorithms to achieve a more reliable ego-state estimate, these can be too computationally expensive to run on-board even by modern standards (Ghadiok et al., 2012; Schauwecker and Zell, 2014). This limits the MAV, as processing power is diverted from tasks at a higher level of cognition. If not properly handled, it can lead to sub-optimal final performances by the MAVs and by the swarm<sup>2</sup>.

## 4.1. Low-Level State Estimation and Control

This section outlines the main sensors and methods that can be used by MAVs to measure their on-board states, laying the foundations for our swarm-focused discussion in later sections. We organize the discussion by focusing on the following parameters: attitude (section 4.1.1), velocity and odometry (section 4.1.2), and height and altitude (section 4.1.3). Moreover, we restrict our overview to *on-board* sensing, as this is in line with the swarming philosophy and the relevant applications.

### 4.1.1. Attitude

It is essential for an MAV to estimate and control its own attitude in order to control its flight (Beard, 2007; Bouabdallah and Siegwart, 2007). Accelerations and angular rotation rates are typically measured through the on-board Inertial Measurement Unit (IMU) sensor (Bouabdallah et al., 2004; Gupte et al., 2012). The IMU measurements can be fused together to both

estimate and control the attitude of an MAV (Shen et al., 2011; Schauwecker et al., 2012; Macdonald et al., 2014; Mulgaonkar et al., 2015). Additionally to the IMU, MAVs equipped with cameras can also use it to infer the attitude with respect to certain reference features or planar surfaces, as in Schauwecker and Zell (2014). Thurrowgood et al. (2009), Dusha et al. (2011), de Croon et al. (2012), and Carrio et al. (2018) estimate the roll and pitch angles of an MAV based on the horizon line (outdoors). The measurements from the IMU and vision can then be filtered together to improve the estimate as well as filter out the accumulating bias from the IMU (Martinelli, 2011). Once known, attitude control can be achieved with a variety of controllers. For a recent survey that treats the topic of attitude control in more detail, we refer the reader to the review by Nascimento and Saska (2019). Of particular interest to swarming are controllers that can provide robustness to disturbances or mishaps. One interesting example is the scheme devised by Faessler et al. (2015), which can automatically re-initialize the leveled flight of an MAV in mid-air.

Measuring and controlling the heading (for instance, with respect to North) is not strictly needed for basic flight. However, it can be an enabler for collective motion by providing a common reference that can be measured locally by all MAVs (Flocchini et al., 2008). Heading with respect to North can be measured with a magnetometer, which is a common component for MAVs (Beard, 2007). A main limitation of this sensor is that it is highly sensitive to disturbances in the environment (Afzal et al., 2011). The disturbances can be corrected for with the use of other attitude sensors. For example, Pascoal et al. (2000) fused gyroscope measurements with the magnetometer in order to filter out disturbances from the magnetometer while also reducing the noise from the gyroscope. Another sensor that has been explored is the celestial compass, which extracts the orientation based on the Sun (Jung et al., 2013; Dupeyroux et al., 2019). Although this sensor is not subject to electro-magnetic disturbances, it is limited to outdoor scenarios and performs best under a clear sky, which may also not always be the case.

### 4.1.2. Velocity and Odometry

A tuned sensor fusion filter with an accurate prediction model can estimate velocity just based on the IMU readings (Leishman et al., 2014). However, the use of additional and dedicated velocity sensors is commonly used to achieve a more robust system

<sup>2</sup>When we relate this to nature, then low-level control and state-estimation seldom requires large “computational” efforts by the individual animal. Rather, they eventually become second nature (Rasmussen, 1983). The real focus is directed to higher level tasks.

without bias. Fixed wing MAVs can be equipped with a pitot tube in order to measure airspeed (Chung et al., 2016). For other designs, such as quadrotors, a popular solution is to measure the optic flow, i.e., the motion of features in the environment, from which an MAV can extract its own velocity (Santamaria-Navarro et al., 2015). To observe velocity, the flow needs to be scaled with the help of a distance measurement, such as height (albeit this assumes that the ground is flat, which may be untrue in cluttered/outdoor environments). Optic flow can be measured with a camera or with dedicated sensors, such as PX4FLOW (Honegger et al., 2013) or the PixArt sensor<sup>3</sup>. Using optical mouse sensors, Briod et al. (2013) were able to make a 46 g quadrotor fly based on only inertial and optical-flow sensors, even without the need to scale the flow by a distance measurement. This was achieved by only using the direction of the optic flow and disregarding its magnitude. In nature, optic flow has also been shown to be directly correlated with how insects control their velocity in an environment (Portelli et al., 2011; Lecoeur et al., 2019). Similar ideas have also been ported to the drone world, whereby the optic flow detection is directly correlated to a control input, without even necessarily extracting states from it (Zufferey et al., 2010). This can be an attractive property in order to create a natural correlation between a sensor and its control properties. State estimates improve when optic flow is fused with other sensors, such as IMU readings or pressure sensors (Kendoul et al., 2009a,b; Santamaria-Navarro et al., 2015), or with the control input of the drone (Ho et al., 2017). As opposed to optic flow sensors, a camera has the advantage that it can observe both optic flow as well as other features in the environment, thus enabling an MAV to get more out of a single sensor. Although this is more computationally expensive, it also provides versatility.

The use of vision also enables the tracking of features in the environment, which a robot can use to estimate its odometry. Using Visual Odometry (VO), a robot integrates vision-based measurements during flight in order to estimate its motion. The inertial variant of VO, known as Visual Inertial Odometry (VIO), further fuses visual tracking together with IMU measurements. This makes it possible for an MAV to move accurately relative to an initial position (Scaramuzza and Zhang, 2019). VIO has been exploited for swarm-like behaviors, such as in the work by Weinstein et al. (2018), whereby twelve MAVs form patterns by flying pre-planned trajectories and use VIO to track their motion. A step beyond VO and its variants is to use Simultaneous Localization And Mapping (SLAM). The advantage of SLAM is that it can mitigate the integration drift of VO-based methods. When solving the full SLAM problem, a robot estimates its odometry in the environment and then corrects it by recognizing previously visited places and optimizing the result accordingly, so as to make a consistent map (Cadena et al., 2016; Cieslewski and Scaramuzza, 2017). Yousif et al. (2015) and Cadena et al. (2016) provide more in-depth reviews of VO and SLAM algorithms. Within the swarming context, a map can also be shared so as to make use of places and features that have been seen by other members of the swarm. One common drawback of VO

and SLAM methods is that they are computationally intensive and thus reserved for larger MAVs (Ghadiok et al., 2012; Schauwecker and Zell, 2014). However, recent developments have also seen the introduction of more light-weight solutions, such as Navion (Suleiman et al., 2018).

Odometry and SLAM are not limited to the use of vision. A viable alternative sensor is the LIDAR (Light Detection and Ranging) scanner, more commonly referred to as “laser scanner.” LIDAR-based SLAM feature the same philosophy as the vision counterparts, but instead of a camera it uses LIDAR to measure depth information and build a map (Bachrach et al., 2011; Opromolla et al., 2016; Doer et al., 2017; Tripicchio et al., 2018). A LIDAR is generally less dependent on lighting conditions and needs less computations, but it is also heavier, more expensive, and consumes more on-board power (Opromolla et al., 2016). Vision and LIDAR can also be used together to further enhance the final estimates (López et al., 2016; Shi et al., 2016).

#### 4.1.3. Height and Altitude

In an abstract sense, the ground represents an obstacle that the MAV must avoid, much like walls, objects, or other MAVs. It does not need to be explicitly known in order to control an MAV, as shown in the work of Beyeler et al. (2009). Unlike other obstacles, however, gravity continuously pulls the MAV toward the ground, meaning that measuring and controlling height and altitude often requires special attention.

Note that we differentiate here between height and altitude. Height is the distance to the ground surface, which can vary when there is a high building, a canyon, or a table. The height of an MAV can be measured with an ultrasonic range finder (or “sonar”). Sonar can provide more accurate data at the cost of power, mass, size, and a limited range. Its accuracy, however, made it a part of several designs (Krajník et al., 2011; Ghadiok et al., 2012; Abeywardena et al., 2013). Infra-red or laser range finders have also been used as an alternative (Grzonka et al., 2009; Gupte et al., 2012). The advantage of an infra-red sensor is that it can be very power efficient, albeit it is only reliable up to a limited range of a few meters, and on favorable light conditions (Laković et al., 2019)<sup>4</sup>. Altitude is the distance to a fixed reference point, such as sea level or a take-off position. A pressure sensor is a common sensor to obtain this measurement (Beard, 2007), but it can be subject to large noise and disturbances in the short term, which can be reduced via low pass filters (Sabatini and Genovese, 2013; Shilov, 2014). If flying outdoors, a Global Navigation Satellite System (GNSS) can also be used to obtain altitude.

The choice of height/altitude sensor has an impact on the swarm behaviors that can be programmed. GNSS and pressure sensors provide a measurement of the altitude of the MAV with respect to a certain position. This is an attractive property, although, as previously discussed, GNSS is limited to outdoor environments, while pressure sensors can be noisy. Moreover, all pressure sensors of all MAVs in the swarm should be equally calibrated. Unlike pressure sensors, ultrasonic sensors or laser range finders do not require this calibration step, since the

<sup>3</sup> “PMW3901MB Product Datasheet” by PixArt Imaging Inc., June 2017.

<sup>4</sup> See [www.st.com/en/imaging-and-photonics-solutions/vl53l0x.html](http://www.st.com/en/imaging-and-photonics-solutions/vl53l0x.html)

measurement is made from the MAV to the nearest surface. However, one must then assume that the MAVs all fly on a flat plane with no objects (or other MAVs below them), which may turn out to not be a valid assumption. SLAM and VIO methods, previously discussed in section 4.1.2, can also estimate altitude/height as part of the odometry/mapping procedure provided that a downwards facing camera is available.

Just as for the use of a common heading like North, the measurements of height and/or altitude can provide a common reference plane for a swarm of MAVs. If the vertical distance between the MAVs is sufficient, it can provide a relatively simple solution for intra-swarm collision avoidance (albeit with constraints—we return to this in section 5.2). It can also enable self-organized behaviors, such as in the work of Chung et al. (2016), where the MAVs are made to follow the one with the highest altitude within their sub-swarm. In this way, the leader is automatically elected in a self-organized manner by the swarm. For example, should a current leader MAV need to land as a result of a malfunction, a new leader can be automatically re-elected so that the rest of the swarm can keep operating.

## 4.2. Achieving Safe Navigation

It is important that each MAV remains safe and that it does not collide with its surroundings, or that damages remain limited in case this happens. This safety requirement can be satisfied in two ways. The first, which is more “passive” and brings us back to MAV design, is to develop MAVs that are mechanically collision resilient. This allows the MAV to hit obstacles without risking significant damage to itself or its environment. With this rationale, Briod et al. (2012), Mulgaonkar et al. (2015, 2018), and Kornatowski et al. (2017) placed protective cages around an MAV. However, the additional mass of a cage can negatively impact flight time and the cage can also introduce drag and controllability issues (Floreano et al., 2017). Instead, Mintchev et al. (2017) developed a flexible design for miniature quadrotors in order to be more collision resilient upon impact with walls. The use of airships has also been proposed as a more collision resilient solution (Melhuish and Welsby, 2002; Troub et al., 2017). The limitations of airships, however, are in their lower agility and restricted payload capacity. More recently, Chen et al. (2019) demonstrated insect scale designs that use soft artificial muscles for flapping flight. The soft actuators, combined with the small scale of the MAV, are such that the MAVs can be physically robust to collisions with obstacles and with each other. Collision resistant designs can even be exploited to improve on-board state estimation, such as in the recent work by Lew et al. (2019), whereby collisions are used as pseudo velocity measurement under the assumption that the velocity perpendicular to an obstacle, at the time of impact, is null. The alternative, or complementary, solution to passive collision resistance is “active” obstacle sensing and avoidance, whereby an MAV uses its on-board sensors to identify and avoid obstacles in the environment.

Collision-free flight can be achieved via two main navigation philosophies: (1) map-based navigation, and (2) reactive navigation. With the former, a map of the environment can be used to create a collision-free trajectory (Shen et al., 2011; Weiss

et al., 2011; Ghadiok et al., 2012). The map can be generated during flight (using SLAM) and/or, for known environments, it can be provided a-priori. The advantage of a map-based approach is that obstacle avoidance can be directly integrated with higher level swarming behaviors (Saska et al., 2016b). Instead, a reactive control strategy uses a different philosophy whereby the MAV only reacts to obstacles in real-time as they are measured, regardless of its absolute position within the environment. In this case, if an MAV detects an obstacle, it reacts with an avoidance maneuver without taking its higher level goal into account. The trajectories pursued with a reactive controller may be less optimal, but the advantage of a reactive control strategy is that it naturally accounts for dynamic obstacles and it is not limited to a static map. The two can also operate in a hierarchical manner, such that the reactive controller takes over if there is a need to avoid an obstacle, and the MAV is otherwise controlled at a higher level by a path planning behavior. Regardless of the navigation philosophy in use, if the MAV needs to sense and avoid obstacles during flight, it will require sensors that can provide it with the right information in a timely manner.

Of all sensors, vision provides a vast amount of information from which an MAV can interpret its direct environment. By using a stereo-camera, the disparity between two images gives depth information (Heng et al., 2011; Matthies et al., 2014; Oleynikova et al., 2015; McGuire et al., 2017). Alternatively, a single camera can also be used. For example, the work of de Croon et al. (2012) exploited the decrease in the variance of features when approaching obstacles. Ross et al. (2013) used a learning routine to map monocular camera images to a pilot command in order to teach obstacle avoidance by imitating a human pilot. Kong et al. (2014) proposed edge detection to detect the boundary of potential obstacles in an image. Saha et al. (2014) and Aguilar et al. (2017) used feature detection techniques in order to extract potential obstacles from images. Alvarez et al. (2016) used consecutive images to extract a depth map (a technique known as “motion parallax”), albeit the accuracy of this method is dependent on the ego-motion estimation of the quadrotor. Learning approaches have also been investigated in order to overcome the limitations of monocular vision. By exploiting the collision resistant design of a Parrot AR Drone, Gandhi et al. (2017) collected data from 11,500 crashes and used a self-supervised learning approach to teach the drone how to avoid obstacles from only a monocular camera. Self-supervised learning of distance from monocular images can also be accomplished without the need to crash, but with the aid of an additional sensor. Lamers et al. (2016) did this by exploiting an infrared range sensor, and van Hecke et al. (2018) applied this to see distances with one single camera by learning a behavior that used a stereo-camera. This is useful if the stereo-camera were to malfunction and suddenly become monocular. Alternative camera technologies have also been developed, providing new possibilities. RGB-D sensors are cameras that also provide a per-pixel depth map, a mainstream example of which is the Microsoft Kinect camera (Newcombe et al., 2011). This particular sensor augments one RGB camera with an IR camera and an IR projector, which together are capable of measuring depth (Smisek et al., 2013). RGB-D sensors have been used on MAVs



to navigate in an environment and avoid obstacles (Shen et al., 2014; Stegagno et al., 2014; Odelga et al., 2016; Huang et al., 2017). One of the disadvantages of these RGB-D sensors over a stereo-camera set-up (whereby depth is inferred from the disparity) is that RGB-D sensors can be more sensitive to natural light, and may thus perform less well in outdoor environments (Stegagno et al., 2014). Finally, in recent years, the introduction of Dynamic Vision Sensor (DVS) cameras has also enabled new possibilities for reactive obstacle sensing. A DVS camera only measures changes in the brightness, and can thus provide a higher data throughput. This enables a robot to quickly react to sudden changes in the environment, such as the appearance of a fast moving obstacle (Mueggler et al., 2015; Falanga et al., 2019a).

The capabilities of a vision algorithm will depend on the resolution of the on-board cameras, the number of the on-board cameras, as well as the processing power on-board. On very lightweight MAVs, such as flapping wings, even carrying a small stereo-camera can be challenging (Olejnik et al., 2019). A further known disadvantage of vision is the limited Field of View (FOV) of cameras. Omni-directional sensing can only be achieved with multiple sets of cameras (Floreano et al., 2013; Moore et al., 2014) at the cost of additional mass, the impact of which is dependent on the design of the MAV.

Although vision is a rich sensor, in that it can provide different types of information, other sensors also can be used for reactive collision avoidance. LIDAR, for instance, has the advantage that it is less dependent on lighting conditions and can provide more accurate data for localization and navigation (Bachrach et al., 2011; Tripicchio et al., 2018). Alternatively, time-of-flight laser ranging sensors have also been proposed for reactive obstacle avoidance algorithms on small drones (Laković et al., 2019). These uni-directional sensors can sense whether an object appears along their line of sight (typically up to a few meters). Due to their small size and low power requirements, they can be used on tiny MAVs (Bitcraze, 2019)<sup>5</sup>.

## 5. INTRA-SWARM RELATIVE SENSING AND COLLISION AVOIDANCE

Once we have an MAV design that can perform basic safe flight, we begin to expand its capabilities toward collaboration in a swarm. Two fundamental challenges need to be considered in this domain. The first is relative localization. This is not only required to ensure intra-swarm collision avoidance, which is a basic safety requirement, but also to enable several swarm behaviors (Bouffanais, 2016). The design choice used for intra-swarm relative localization defines and constrains the motion of the MAVs relative to one another, which affects the swarming behavior that can be implemented. The second challenge is intra-swarm communication. Much like knowing the position of neighbors, the exchange of information between MAVs can help the swarm to coordinate (Valentini, 2017; Hamann, 2018). In this section, we explore the state of the art for relative localization (section 5.1), reactive collision avoidance maneuvers

(section 5.2), and we discuss intra-swarm communication technologies (section 5.3).

### 5.1. Relative Localization

In outdoor environments, relative position can be obtained via a combination of GNSS and intra-swarm communication. Global position information obtained via GNSS is communicated between MAVs and then used to extract relative position information. This has enabled connected swarms that can operate in formations or flocks (Chung et al., 2016; Yuan et al., 2017). An impressive recent display of this in the real world was put into practice by Vászrhelyi et al. (2018), who programmed a swarm of 30 MAVs to flock. The same concept can be applied to indoor environments if pre-fitted with, for example: external markers (Pestana et al., 2014), motion-tracking cameras (Kushleyev et al., 2013), antenna beacons (Ledergerber et al., 2015; Guo et al., 2016), or ultra sound beacons (Vedder et al., 2015). However, this dependency on external infrastructure limits the swarm to being operable only in areas that have been properly fitted to the task. Several tasks, especially the ones that involve exploration, cannot rely on these methods. In order to remove the dependency on external infrastructure, there is a need for technologies that allows the MAVs themselves to obtain a direct MAV-to-MAV relative location estimate. This is still an open challenge, with several technologies and sensors currently being developed.

One of the earlier solutions for direct relative localization on flying robots proposed the use of infrared sensors (Roberts et al., 2012). However, since infrared sensors are uni-directional, this used an array of sensors (both emitting and receiving) placed around the MAV in order to approach omni-directionality, making for a relatively heavy system. Alternatively, vision-based algorithms have once again been extensively explored. However, the robust visual detection of neighboring MAVs is not a simple task. The object needs to be recognized at different angles, positions, speeds, and sizes. Moreover, the image can be subject to blur or poor lighting conditions. One way to address this challenge is with the use of visual aids mounted on the MAVs, such as visual markers (Faigl et al., 2013; Krajník et al., 2014; Nægeli et al., 2014), colored balls (Roelofsen et al., 2015; Epstein and Feldman, 2018), or active markers, such as infrared markers (Faessler et al., 2014; Teixeira et al., 2018)<sup>6</sup> or Ultra Violet (UV) markers (Walter et al., 2018, 2019). Visual aids simplify the task and improve the detection accuracy and reliability. However, they are not as easily feasible on all designs, such as flapping wing MAVs or smaller quadrotors. Marker-less detection of other MAVs is very challenging, since other MAVs have to be detected against cluttered, possibly dynamic backgrounds while the detecting MAV is moving by itself as well. A successful current approach is to rely on stereo vision, where other drones can be detected because they “float” in the air unlike other objects like trees or buildings. Carrio et al. (2018) explored a deep learning algorithm for the detection of other MAVs in stereo-based disparity images. An alternative is to detect other MAVs in monocular still images. Like the detection in

<sup>5</sup>See [www.bitcraze.io/multi-ranger-deck/](http://www.bitcraze.io/multi-ranger-deck/)

<sup>6</sup>The solution by Teixeira et al. (2018) additionally uses communication between the MAVs.

stereo disparity images, this removes the difficulty of interpreting complex motion fields between frames, but it introduces the difficulty of detecting other, potentially (seemingly) small MAVs against background clutter. To solve the challenge, Opromolla et al. (2019) used a machine learning framework that exploited the knowledge that the MAVs were supposed to fly in formation. Their scheme used the knowledge of the formation in order to predict the expected position of a neighboring MAV and focus the vision-based detection on the expected region, thus simplifying the task. Employing a more end-to-end learning technique, Schilling et al. (2019) used imitation learning to autonomously learn a flocking behavior from camera images. Following the attribution method by Selvaraju et al. (2017), Schilling et al. studied the influence that each pixel of an input image had on the predicted velocity. It was shown that the parts of the image whereby neighboring MAVs could be seen were more influential, demonstrating that the network had implicitly learned to localize its neighbors. Despite the promising preliminary results, it is yet to be seen how it can handle other MAVs sizes or more cluttered backgrounds. Finally, it is possible to use the optic flow field for detecting other MAVs. This approach could have the benefit of generality, but it would require the calculation and interpretation of a complex, dense optic flow field. To our knowledge, this method has not yet been investigated.

From a swarming perspective, it may also be desirable to know the ID of a neighbor. However, IDs may be difficult to detect using vision without the aid of markers. This issue was explored by Stegagno et al. (2011), Cognetti et al. (2012), and Franchi et al. (2013) with fusion filters that infer IDs over time with the aid of communication. Moreover, cameras have a limited FOV. This limits the behaviors that can be achieved by the swarm. For instance, it may be limiting for surveillance tasks where quadrotors may need to look away from each other but can't or else they may collide or disperse. It can be addressed by placing several cameras around the MAVs (Schilling et al., 2019), but at the cost of additional mass, size, and power, which in turn creates new repercussions.

The use of vision is not only limited to directly recognizing other drones in the environment. With the aid of communication, two or more MAVs can also estimate their relative location indirectly by matching mutually observed features in the environment. The MAVs can compare their respective views and infer their relative location. In the most complete case, each MAV uses a SLAM algorithm to construct a map of its environment, which is then compared in full (as discussed in section 4, this can also be accomplished using other sensors, such as LIDAR, so this approach is not only reserved for vision). Although SLAM is a computationally expensive task, more easily handled centrally (Achtelik et al., 2012; Forster et al., 2013), it can also be run in a distributed manner, making for an infrastructure free system (Cunningham et al., 2013; Cieslewski et al., 2018; Lajoie et al., 2019). For a survey of collaborative visual SLAM, we refer the reader to the paper by Zou et al. (2019) and the sources therein. An additional benefit of collective map generation is that the MAVs benefit from the observations of their team-mates and can thus achieve a better collective map.

However, if the desired objective is only to achieve relative localization, the computations can be simplified. Instead of computing and matching an entire map, the MAVs need only to concern themselves with the comparison of mutually observed features in order to extract their relative geometric pose (Achtelik et al., 2011; Montijano et al., 2016). This requires that the images compared by the MAVs have sufficient overlap and can be uniquely identified.

An alternative stream of research leverages only communication between MAVs to achieve relative localization, while also using the antennas as relative range sensors. Here, we will refer to these methods as *communication-based ranging*. The advantage of this method is that it offers omni-directional information at a relatively low mass, power, and processing penalty, leveraging a technology that is likely available on even the smallest of MAVs. Szabo (2015) first proposed the use of signal strength to detect the presence of nearby MAVs and engage in avoidance maneuvers. Also for the purposes of collision avoidance, Coppola et al. (2018) implemented a beacon-less relative localization approach based on the signal strength between antennas, using the Bluetooth Low Energy connectivity already available on even the smaller drones. Guo et al. (2017) proposed a similar solution using UltraWide Band (UWB) antennas for relative ranging, which offer a higher resolution even at larger distances. However, this work used one of the drones as a reference beacon for the others. One commonality between the solutions by Guo et al. (2017) and Coppola et al. (2018) was that the MAVs were required to have a knowledge of North, which enabled them to compare each other's velocities along the same global axis. However, in practice this is a significant limitation due to the difficulties of reliably measuring North, especially if indoors, as already discussed in section 4.1.1. To tackle this, van der Helm et al. (2019) showed that, if using a high accuracy ranging antenna, such as UWB, then it is not necessary for the MAVs to measure a common North. However, selecting this option creates fundamental constraints on the high-level behaviors of the swarm. This issue is there for the case where North is known and when it is not, albeit the requirement when North is not known are more stringent. If North is known, at least one of the MAVs must be moving relative to the other for the relative localization to remain theoretically observable. If North is not known, all MAVs must be moving. The MAVs remain bound to trajectories that excite the filter (van der Helm et al., 2019). For the case where North is known, Nguyen et al. (2019) proposed that a portion of MAVs in the swarm should act as "observers" and perform trajectories that persistently excite the system.

Another solution is to use sound. Early research in this domain was performed by Tijs et al. (2010), who used a microphone to hear nearby MAVs. This was explored in more depth by Basiri (2015) using full microphone arrays for relative localization. A primary issue encountered was that the sound emitted by the listening quadrotor would mask the sound of the neighboring MAVs, which were also similar. This was addressed with the use of a "chirp" sound, which can then be easily heard by neighbors, in order to overcome this issue (Basiri et al., 2014, 2016). In recent work, Cabrera-Ponce et al. (2019) proposed the

use of a Convolutional Neural Network to detect the presence of nearby MAVs. This is done using a large scale microphone array (Ruiz-Espitia et al., 2018) featuring eight microphones based on the ManyEars framework (Grondin et al., 2013). Specific to sound sensors, the accuracy of the detection depends on how similar the sounds of other MAVs are. Moreover, the localization accuracy depends on the microphone setup. Most works use a microphone array, where the localization accuracy depends on the length of the baseline between microphones, which is inherently limited on small MAVs.

As it can be seen, several different techniques exist. Minimally, these technologies should enable neighboring MAVs to avoid collisions with one another. However, the particular choice of relative localization technology creates a fundamental constraint on the swarm behavior that can be achieved. For example, communication-based ranging methods have unobservable conditions depending on the MAVs' motion, and sound-based localization with microphone arrays will be less accurate when used on smaller MAVs. Similarly, certain swarm behaviors (e.g., one that requires known IDs, or long range distances) may place certain requirements on which technology is best to be used. In **Table 1**, we outline the major relative localization approaches with their advantages and disadvantages.

## 5.2. Intra-Swarm Collision Avoidance

Collision detection and avoidance of objects in the environment has already been discussed in section 4.2. As MAVs operate in teams, relative intra-swarm collision avoidance also becomes a safety-critical behavior that should be implemented. The complexity of this task is that it requires a collaborative maneuver between two or more MAVs.

MAVs operate in 3D space, and thus relative collision avoidance could be tackled by vertical separation. However, particularly in indoor environments where vertical space is limited, vertical avoidance maneuvers may cause undesirable aerodynamic interactions with other MAVs as well as other parts of the environment. For quadrotors, while aerodynamic influence is negligible when flying side-by-side, flying above another will create a disturbance for the lower one (Michael et al., 2010; Powers et al., 2013). Furthermore, emergency vertical maneuvers could also cause a quadrotor to fly too close to the ground, which creates a ground effect and pushes it upwards, or, if indoors, to fly too close to the ceiling, which creates a pulling effect toward the ceiling (Powers et al., 2013). Vertical avoidance may also corrupt the sensor readings of the MAV. For instance, height may be compromised if another MAV obstructs a sonar sensors. Overall, horizontal avoidance maneuvers are desired.

A popular algorithm for obstacle avoidance, provided that the robots know their relative position and velocity, is the Velocity Obstacle (VO) method (Fiorini and Shiller, 1998). The core idea is for a robot to determine a set of all velocities that will lead to collisions with the obstacle (a collision cone), and then choose a velocity outside of that set, usually the one that requires minimum change from the current velocity. VO has stemmed a number of variants specifically designed to deal with multi-agent avoidance, such as Reciprocal Velocity Obstacle (RVO) (van den Berg et al., 2008; van den Berg et al., 2011), Hybrid

Reciprocal Velocity Obstacle (HRVO) (Snape et al., 2009), and Optimal Reciprocal Collision Avoidance (ORCA) (Snape et al., 2011). These variants alter the set of forbidden velocities in order to address reciprocity, which may otherwise lead to oscillations in the behavior. These methods have been successfully applied on MAVs, both in a decentralized way as well as via centralized re-planners. They accounted for uncertainties by artificially increasing the perceived radii of the robots. Alonso-Mora et al. (2015) showed the successful use of RVO on a team of MAVs such that they may adjust their trajectory with respect to a reference. This was done using an external MCS for (relative) positioning. Coppola et al. (2018) showed a collision cone scheme with on-board relative localization, introducing a method to adjust the cone angle in order to better account for uncertainties in the relative localization estimates. A disadvantage of VO methods and its derivatives is scalability. If the flying area is limited and the airspace becomes too crowded, then it may become difficult for MAVs to find safe directions to fly toward (Coppola et al., 2018). Another avoidance algorithm, called Human-Like (HL), presents the advantage that the heading selection is decoupled from speed selection (Guzzi et al., 2013a; Guzzi et al., 2014), such that the MAVs only engage in a change in heading. HL has been found to be successful even when operating at relatively lower rates (Guzzi et al., 2013b). Although it has not been tested on MAVs, their tests also demonstrated generally better scalability properties.

Alternatively, attraction and repulsion forces between obstacles are also a valid algorithm for collision avoidance. This is a common technique which has been extensively studied in swarm research (Reynolds, 1987; Gazi and Passino, 2002; Gazi and Passino, 2004). If one wishes for the MAVs to flock, these attraction and repulsion forces can also be directly merged with the swarm controller (Vásárhelyi et al., 2018). One potential short-coming of this approach is that it can lead to equilibrium states whereby the swarm remains in a fixed final formation, although this can also be seen as a positive property that can be exploited (Gazi and Passino, 2011).

In summary, multiple methods exist for intra-swarm collision avoidance. Given sufficiently accurate relative locations, these methods are very successful. The main challenges here are: (1) how to deal with uncertainties and unobservable conditions deriving from the localization mechanism used by the drones, and (2) how to keep guaranteeing successful collision avoidance when the swarm scales up to very large numbers.

## 5.3. Intra-Swarm Communication

Direct sharing of information between neighboring robots is an enabler for swarm behaviors as well as relative sensing (Valentini, 2017; Hamann, 2018; Pitonakova et al., 2018). To achieve the desired effect, it needs to be implemented with scalability, robustness, and flexibility in mind. Common problems that can otherwise arise are: (1) the messaging rate between robots is too low (low scalability); (2) high packet loss (low robustness); (3) communication range is too low (low scalability and flexibility); (4) inability to adapt to a switching network topology (low flexibility) (Chamanbaz et al., 2017).

Solutions to the above depend on the application. With respect to hardware, the three main technologies in the state

**TABLE 1** | Current technologies in the state of the art for relative localization between MAVs, with their main advantages and disadvantages.

Technology	Sample references	Advantages	Disadvantages
Vision (direct, passive)	Faigl et al. (2013) Krajník et al. (2014) Nägeli et al. (2014) Roelofsen et al. (2015) Carrio et al. (2018)	<ul style="list-style-type: none"> <li>• Rich information</li> <li>• Passive sensor</li> <li>• Possible to extract ID (if using markers or having otherwise visually distinct MAVs)</li> <li>• Lightweight (depending on model)</li> <li>• Scalable (provided environment is not cluttered)</li> </ul>	<ul style="list-style-type: none"> <li>• Computationally expensive</li> <li>• Limited FOV</li> <li>• Dependent on lighting conditions</li> <li>• Dependent on visual clutter in environment</li> <li>• No IDs (if marker-less)</li> <li>• Needs visual line of sight</li> </ul>
Observation matching	Achtelik et al. (2011) Montijano et al. (2016)	<ul style="list-style-type: none"> <li>• No direct line of sight is needed</li> <li>• Thrives in visually cluttered environments</li> <li>• Includes IDs (via communication)</li> </ul>	<ul style="list-style-type: none"> <li>• Active sensor, requires communication</li> <li>• Requires sufficient visual overlap</li> <li>• Computationally expensive</li> <li>• Dependent on lighting conditions</li> </ul>
Communication-based ranging	Szabo (2015) Guo et al. (2017) Coppola et al. (2018) van der Helm et al. (2019) Nguyen et al. (2019)	<ul style="list-style-type: none"> <li>• Low mass</li> <li>• Omni-directional</li> <li>• Possible to extract IDs</li> <li>• Can work in visually cluttered environments</li> <li>• Enables communication of additional data</li> <li>• Scalable (provided that the sound environment is, or does not become, not cluttered)</li> <li>• Passive sensor</li> <li>• Omni-directional</li> <li>• Can work in visually cluttered environments</li> </ul>	<ul style="list-style-type: none"> <li>• Active sensor, requires communication</li> <li>• Needs relative excitation maneuvers</li> <li>• Noisy (depending on sensor)</li> <li>• Difficult to scale due to communication interference</li> <li>• Self-propeller noise</li> <li>• Limited range</li> <li>• Noise pollution (if using “chirps”)</li> <li>• No IDs (if chirp-less)</li> <li>• Limited angular accuracy due to limited baseline between microphones in the array</li> </ul>
Sound	Basiri et al. (2014, 2016) Cabrera-Ponce et al. (2019)	<ul style="list-style-type: none"> <li>• Can work in visually cluttered environments</li> </ul>	<ul style="list-style-type: none"> <li>• Limited range</li> <li>• Noise pollution (if using “chirps”)</li> <li>• No IDs (if chirp-less)</li> <li>• Limited angular accuracy due to limited baseline between microphones in the array</li> </ul>
Infra-red (sensor array)	Roberts et al. (2012)	<ul style="list-style-type: none"> <li>• Accurate</li> <li>• Computationally simple</li> <li>• Lower dependence on lighting conditions<sup>a</sup></li> <li>• Can work in visually cluttered environments</li> </ul>	<ul style="list-style-type: none"> <li>• Heavy</li> <li>• Needs visual line of sight</li> <li>• Many active sensors result in high energy expense</li> </ul>
Vision (direct, with active markers)	Faessler et al. (2014) Teixeira et al. (2018) Walter et al. (2018, 2019)	<ul style="list-style-type: none"> <li>• Accurate</li> <li>• Possible to extract IDs</li> <li>• Lower dependence on lighting conditions</li> <li>• Can work in visually cluttered environments</li> </ul>	<ul style="list-style-type: none"> <li>• Needs visual line of sight</li> <li>• Many active sensors result in high energy expense</li> </ul>

<sup>a</sup>Roberts et al. (2012) tested the sensor for 0, 500, and 10,000 lux and found <1% relative error between these lighting conditions. However, the sensor was not tested outdoors.

of the art are: Bluetooth, WiFi, and ZigBee (Bensky, 2019). All three operate in the 2.4 GHz band<sup>7</sup>. Bluetooth is energy efficient, but features a low maximum communication distances of  $\approx 10$ –20 m (indoors, depending on the environment and version). This makes it more important to establish a network that can adapt to a switching topology, as it is very likely to change during operations. The latest version of the Bluetooth standard, Bluetooth 5, features a higher range and a higher data-rate despite keeping a low power consumption. It also has longer advertising messages, such that, without pairing, asynchronous network nodes can exchange messages of 255 bytes instead of 31 (Collotta et al., 2018). Bluetooth antennas were used in the previously discussed work of (Coppola et al.,

2018) on a swarm of 3 MAVs to exchange data indoors and to measure their relative range. In comparison to Bluetooth, WiFi is known to be less energy efficient, but works more reliably at longer ranges and has a higher data throughput. Chung et al. (2016) used WiFi to enable a swarm of 50 MAVs to form an *ad-hoc* network. WiFi was also used by Vászrhelyi et al. (2018) in combination with an XBee module<sup>8</sup> using a proprietary communication protocol. ZigBee’s primary benefits are scalability (it can keep up to, theoretically, 64,000 nodes) and low power, although it has a low data communication rate (Bensky, 2019)<sup>9</sup>. Depending on the application, this may or may not be an issue depending on what the intra-swarm communication requirements are. Allred et al. (2007) used a

<sup>7</sup> WiFi also operates at other frequency bands. The 5 GHz band, for instance, is typically known to feature a lower interference (Verma et al., 2013). ZigBee can also operate at the 868 and 915 MHz frequency bands (Collotta et al., 2018).

<sup>8</sup>Not to be confused with ZigBee (Faludi, 2010).

<sup>9</sup>Note that Bluetooth Low Energy, a sub-version of the Bluetooth standard, also requires very little power. Tests by Collotta et al. (2018) return that Bluetooth 4.2 and 5.0 have a lower power consumption than ZigBee.

ZigBee module to enable communication on a flock of fixed wing MAVs due to its combination of low energy consumption and long range (offering “a range of over 1 mile at 60 mW”). For comparisons of technical details of these technologies we refer the reader to the detailed book by Bensky (2019), the MAV-focused review by Zufferey et al. (2013), as well as the earlier comparisons by Lee et al. (2007).

In addition to the technologies discussed above, there is also the possibility of enabling indirect communication via cellular networks. In the near future, 5G networks are expected to make it possible to have a reliable and high data throughput between several MAVs (Campion et al., 2018). Finally, the use of UWB can also gain more relevance in the future, especially because its additional capability to accurately measure the range between MAVs, as discussed in section 5.1, can be very helpful for swarms. One technological challenge is that communication needs power, and while this may be near-negligible for the bigger MAVs, it is not so for the smaller designs (Petricca et al., 2011). From this perspective, the communication-based relative localization discussed in section 5.1, which can also double as a communication device for MAVs, is an interesting solution if one desires a system that can achieve both goals simultaneously. However, using any relative localization approach that relies on communication means that having a stable connection among MAVs is an important requirement, and possibly a safety critical one. Moreover, high messaging rates also become important in order to have a high update rate.

## 6. SWARM-LEVEL CONTROL

We finally arrive at the “swarm” part of this paper. Once we have reliable MAVs that can safely fly in an environment, localize one another, and perhaps even communicate, we can begin to exploit them as a swarm. The complexity of this task stems from the fact that, due to the decentralized nature of the swarm, the local actions that a robot takes can have any number of repercussions at the global level. These cannot be known unless the system is fully observed and optimized for, which the individual robot cannot do.

This section discusses possible approaches to design MAV swarm behaviors. Prominent examples of behaviors are: flocking, formation flight, distributed sensing (e.g., mapping/surveillance), and collaborative transport and object manipulation<sup>10</sup>. Of these, formation flight receives significant attention. It can be useful for several applications, such as surveillance, mapping, or cinematography so as to collaboratively observe a scene (Mademlis et al., 2019). Additionally, it can also be used for collaborative transport (de Marina and Smeur, 2019), and it has even been shown that certain formations lead to energy efficient flight for groups (Weimerskirch et al., 2001). Flocking behaviors bear similar properties to formation flight, but with more “fluid” inter-agent behaviors that allow the swarm to re-organize according to their current neighborhood and the environment.

<sup>10</sup>Note that this list not exhaustive. Additionally, we will see that there may also be overlaps between these behaviors. For example, as explored in section 6.1, flocking behaviors may achieve fixed formations under certain equilibria.

Distributed sensing behaviors may require the swarm to travel in a formation or flock, but may also include behaviors in which the swarm distributes over pre-specified areas (Bähneemann et al., 2017) or disperses (McGuire et al., 2019). Collaborative transport and object manipulations take two forms. The first is that of MAVs individually foraging for different objects and bringing them to base (Bähneemann et al., 2017), the second is that of jointly carrying a load that is too heavy for the individual MAV to carry (Tagliabue et al., 2019). In order to achieve the behaviors above, and others, the MAVs can also engage in a number of more general swarm behaviors, such as distributed task allocation or collective decision making. For all cases, the challenge is to endow the MAVs with a controller that achieves the desired swarm behavior while also avoiding undesired results (Winfield et al., 2005, 2006).

Similarly to the review by Brambilla et al. (2013) (which the reader is referred to for a general overview of swarm robotics and engineering), we divide the design methods in two categories. The first, which we call “manual design methods,” refers to hand-crafted controllers that instigate a particular behavior in the swarm. These are discussed in section 6.1, where we provide an overview of the state of the art for different swarm behaviors. The second, which we refer to as “automatic design methods,” uses machine learning techniques in order to design and/or optimize the controller for an arbitrary goal. This is discussed in section 6.2. We discuss the advantages and disadvantages between the two, from the perspective of designing swarms of MAVs, in section 6.3.

### 6.1. Manual Design Methods

This is the “classical” strategy to control, whereby a swarm designer develops the controllers so as to achieve a desired global behavior. For swarm robotics, we differentiate between two approaches. One approach is to design local behaviors, analyze them, and then manually iterate until the swarm behaves as desired. Another approach is to make mathematical models of the robots and their interactions and then design a suitable controller that comes with a certain proof of convergence. The latter approach has some obvious advantages if one succeeds, but it makes the designer face the full complexity of swarm systems. Hence, such methods typically have limited applicability. For example, in the work of Izzo and Pettazzi (2007), the behavior is limited to only symmetrical formations of limited numbers of agents. The preferred approach is dependent on the swarm behavior that the designer wishes to achieve, under the constraints of the local properties of each MAV.

A large portion of methods focuses on formation control algorithms, whereby the goal is for the MAVs to form and/or keep a tight formation during flight. To hold a formation, the MAVs must hold a relative position or distance between given neighbors, such that they can move as one unit through space. See, for instance, the works of Quintero et al. (2013), Schiano et al. (2016), de Marina et al. (2017), Yuan et al. (2017), and de Marina and Smeur (2019). One advantage of flying in formation for MAV swarms is their predictability during operations. Several methods provide robust controllers with mathematical proofs that the formation can be achieved and maintained during flight.

A review dedicated to formation control algorithms for MAVs is provided by Oh et al. (2015). Chung et al. (2018) also discuss different methods.

There are applications for which a rigid formation is sub-optimal, undesired, or unnecessary, and it is better for the MAVs to move through space in a flock. Flocking behaviors were originally synthesized from the motion of animals in nature (Aoki, 1982), and were most famously formalized by Reynolds (1987) with the intent of simulating swarms in computer animations. The behavior is typically characterized by a combination of simple local rules: attraction forces, repulsion forces, heading alignment with neighbors, speed agreement with neighbors. This behavior naturally incorporates collision avoidance via the repulsion rule, and it has also been explored as a means to collectively navigate in an environment with obstacles, whereby the obstacles provide additional repulsion forces (Saska et al., 2014; Saska, 2015). Alternatively, the local rules can also be exploited to achieve formations by making use of equilibrium points between attraction and repulsion forces (Gazi, 2005). Depending on the way in which the rules are used, they can be incorporated into an iterative approach, or they can be made part of a mathematical regime combined with the model of the robot. An early real-world demonstration of distributed flocking was achieved by Hauert et al. (2011) with a swarm of ten fixed wing MAVs. The more recent work by Vásárhelyi et al. (2018) demonstrated outdoor flocking for a swarm of 30 quadrotors.

Concerning behaviors, such as distributed sensing, exploration, or mapping, there are several different types of solutions that have been developed specifically for MAVs. Typically, these are found to vary depending on the nature of the task, requiring the designer to make careful choices on the best algorithm to be used. Bähmann et al. (2017) and Spurný et al. (2019), aided by GNSS for positioning, divided a search area into multiple regions so that a team of three MAVs could efficiently explore it with a pre-planned trajectory. The recent work of McGuire et al. (2019) demonstrated a swarm of six Crazyflie MAVs performing an autonomous exploration task in an unknown indoor environment. Each MAV acted entirely locally based on a manually designed bug algorithm which enabled exploration as well as homing to a reference beacon.

## 6.2. Automatic Methods for Behavior Design and Optimization

In the last few decades, the increasing power of machine learning methods cannot be denied, with multiple examples in robotics, autonomous driving, smart-homes, and more. Machine learning techniques offer a way to automatically extract the local controller that can fulfill a task, relieving us from the need to design it ourselves. However, the problem shifts to devising algorithms that can efficiently and effectively discover the controllers for us. In this section, we discuss the possibilities based on two primary machine learning approaches in swarm intelligence research: Evolutionary Robotics (ER) and Reinforcement Learning (RL).

### 6.2.1. Evolutionary Robotics

ER uses the concept of *survival of the fittest* in order to efficiently search through the design space for an effective controller (Nolfi,

2002)<sup>11</sup>. It has been widely adopted in swarm robotics literature in order to evolve local robot controllers that optimize the performance of the swarm with respect to a global, swarm-level objective (Trianni, 2008). ER bypasses the analysis of the relation between the local controllers and the global behavior of the swarm. Instead, it optimizes the controllers “blindly” by means of several evaluations in an evolutionary process, which most often happens in simulation, but can also be performed in the real world (Eiben, 2014). Evolved solutions often exploit the robots’ bodies and environment, including the behaviors of other swarm members. Moreover, thanks to the blind optimization, not only the controller can be evolved, but also other factors, such as the communication between robots (Ampatzis et al., 2008). Likewise, ER offers a generic approach to generate swarm controllers of different types, including, but not limited to: neural networks (Trianni et al., 2003; Silva et al., 2015), grammar rules (Ferrante et al., 2013), behavior trees (Scheper et al., 2016; Jones et al., 2018, 2019), and state machines (Francesca et al., 2014). Although neural network architectures can be very powerful, the advantage of the latter methods is that they can be better understood by a designer, which makes it easier to cross the “reality gap” between simulation and the real world when deploying the controllers on the real robots (Jones et al., 2019). Crossing the reality gap is a major challenge in the field of ER and many different approaches have been investigated, also for neural networks. See Scheper (2019) for a more extensive discussion on these methods.

A major challenge for the effective use of ER, especially for swarm robotics, is the design of the fitness functions to be optimized (Francesca and Birattari, 2016). This is usually left to the designer’s ability to explicitly define the key elements that indicate the success of a behavior in a measurable and quantitative manner. It is not uncommon to see empirically defined parameters that represent certain desired elements, such as “safety” in the example of Duarte et al. (2016). As task complexity increases, so does the challenge of designing a fitness function. In the worst case, it may become *uninformative* or even *deceptive*, leading the algorithm to not finding the desired behavior (Silva et al., 2016). Different approaches have been proposed to tackle this issue, such as behavioral decomposition or incremental learning (Nelson et al., 2009). The risk with these strategies, however, is that the designer shapes the learning of the task too much, which may lead to sub-optimal performances. As an alternative strategy for learning complex tasks, Lehman and Stanley (2011) proposed novelty search, whereby the fitness is not defined by how well the task is performed, but by how “novel” a behavior is. This can lead to finding more unorthodox solutions, also for swarm robotics (Gomes et al., 2013). Potential drawbacks of this approach are that the search becomes less directed, and that the shaping shifts from defining a fitness function to defining what constitutes a “behavior.”

<sup>11</sup> Looking at the complexity achieved by natural swarming systems, it also seems intuitive that such complexity could be achieved automatically by mimicking an evolutionary process (Bouffanais, 2016). It is no surprise that a closely related discipline to ER is that of Artificial Life (AL), dedicated to artificially representing life-like processes, albeit with generally more open ended exploratory goals (Bedau, 2003; Trianni, 2014).

To conclude, the ER approach applied to swarming has the large advantage that it deals with complexity by actually bypassing it. However, this currently comes at the cost of needing many evaluations involving the simulation of not one but multiple robots, which leads to longer lasting evolutions. An additional problem of simulating a specific number of robots to evolve a swarm behavior is that the evolution may overfit the behaviors not only to the (simulation) environment, but also to the exact number of robots that were used during the evolution. A naive solution is to simulate different swarm sizes over the evolution, but this will take even more simulation time, and in any case, the number of robots will be limited, meaning that scalability is not guaranteed. Recent developments in this domain have seen the introduction of size-agnostic techniques (Coppola et al., 2019). Finally, although there are studies on online evolutionary learning for swarm robotics (Bredeche et al., 2018), online evolutionary strategies have yet to be explored (in practice) for MAVs.

### 6.2.2. Reinforcement Learning

With RL, a robot is made to learn by trial-and-error from interacting with its environment under a certain reward scheme. This approach teaches the robot an optimal mapping between a state and the action that it should take so as to maximize its final reward (Sutton and Barto, 2018). RL has been widely used in robotics, and it has thus also found its way to swarm robotics (Brambilla et al., 2013). The advantage of RL is that the robots can explore the environment and continuously adapt their behavior. Several techniques have been proposed over the years for multi-agent RL (Busoniu et al., 2008). However, within swarm robotics literature, it has generally received less attention than ER (Brambilla et al., 2013). A main difficulty with this approach is that, from the perspective of the individual robot, being in a swarm is a non-Markovian task, and each robot only has a partial observation of the full global state. A potential issue, for instance, is “state aliasing,” which refers to when multiple states appear to be the same from the perspective of the agent, even though they are not (McCallum, 1997). It has been demonstrated that ER can achieve better solutions for non-Markovian tasks (de Croon et al., 2005).

The solution to use RL with non-Markovian task leads to a Partially Observable Markov Decision Problem (POMDP). In this case, a robot keeps a history of its observations and thus extracts the most likely global state from them. RL can be applied to POMDPs (Ishii et al., 2005), yet features scalability issues (the so called “state explosion”), especially when ported to the swarm domain because the global state of the swarm, which it tries to estimate, can take exponentially many forms (Parsons and Wooldridge, 2002). In recent work, Hüttenrauch et al. (2019) proposed to use mean feature embeddings which encode a mean distribution of the agents. This compression is then invariant to the number of agents in the swarm. Another known difficulty of RL with respect to ER is the credit assignment problem. This refers to the challenge of decomposing the global rewards into local rewards for each robot, as the individual contribution of a single robot to a global task may not always be clearly determined (Brambilla et al., 2013). The credit assignment problem is also

manifested over time, as it is difficult to judge which prior action was most conducive.

In short, until now ER appears to be a more appropriate choice for learning control in swarms, as it allows robots to exploit non-Markovian properties of the problem (e.g., the states and behaviors of other robots). However, because of the reality gap, online learning methods may turn out very useful in the future, including RL methods.

## 6.3. Manual vs. Automatic Methods for MAV Swarms

A primary advantage of manual design methods for MAV swarms is that the solutions are generally better understood, given that they have to be designed and programmed manually. The algorithms that are developed can be analyzed, and in certain cases it can even be assessed whether the system will converge to the desired properties and even be resilient to faults (Saldaña et al., 2017; Saulnier et al., 2017). This is a particularly attractive property for MAV applications, where safety and predictability are a primary concern. A second advantage is that they carry a clearer breakdown of the requirements. For these reasons, it is not surprising that, to the best of our knowledge and as confirmed by Chung et al. (2018), most real-world implementations of MAV swarms to date have relied on primarily manually designed swarming algorithms. These advantages have also been acknowledged by the automatic design community, which has brought a general interest in using automatic approach to develop explicit controllers, such as state machines (Francesca et al., 2014, 2015) or behavior trees (Kuckling et al., 2018; Jones et al., 2019). In future work, the use of these methods could lead to a compromise between extracting an understandable controller and exploiting the power of automatic methods.

A challenge of designing an algorithm manually is in the need to ensure that it can work within the limitations of the system. For instance, if using a communication-based ranging relative localization system, the relative location estimate is only observable when both MAVs are moving in such a way that the system is excited (Nguyen et al., 2019). Alternatively, cameras can be limited by the FOV and be forced to keep a reference neighbor in the center (Nägeli et al., 2014). This may be undesirable for the final application of the swarm (e.g., surveillance), since the camera is kept pointing to other MAVs as opposed to interesting features in the environment. Examples, such as these serve to show how a manually designed algorithm can either fail to regard certain elements, or may not exploit the environment optimally so as to best deal with the limitations. An automatic method, on the other hand, could extract a controller that best deals with the limitations, possibly finding solutions that cannot be easily designed manually. For instance, ER studies show that evolved robot controllers can find behaviors that tightly exploit the sensory and motor capabilities of the given robot (Nolfi, 2002)—this is called sensory-motor coordination.

Despite their power, the application of automatic design methods to MAV swarms are relatively few. One of the first steps was done by Hauert et al. (2009) for the purposes of developing a flying communication network. In this case, the authors proposed to reverse engineer the behavior of an evolved neural network and subsequently program a similar behavior manually.

This approach provided original and “creative” insights that enabled them to design a viable and flexible behavior. In later work, Szabo (2015) applied evolutionary behavior trees to a team of MAVs for the purposes of collision avoidance, exploiting the increased readability of behavior trees. The MAVs only knew each other’s relative distance (not position) as measured by noisy Bluetooth signal strength, yet the evolved behavior was capable of reducing the number of collisions in a cluttered space. The automatically evolved behavior tree was not only simpler (fewer nodes/branches), but also performed better when compared to a manually designed one. Scheper and de Croon (2017) trained a neural network to form a triangle with a team of three MAVs, inspired by a similar task by Izzo et al. (2014). Although not aimed at MAVs, Izzo et al. (2014) had previously shown that an automatic method was able to extract a behavior with which homogeneous agents could self-organize into asymmetric patterns, whereas the previously developed manual approaches for the same system were limited to symmetric patterns (Izzo and Pettazzi, 2007). Scheper and de Croon (2017) additionally showed that evolving a controller at a higher level of abstraction does not necessarily compromise the ability of automatic methods to exploit an environment and sensory-motor relationships, yet helps to reduce the reality gap. The more recent work of Schilling et al. (2019) showed that it’s possible to learn a flocking behavior directly from camera images using imitation learning. This was demonstrated in a real world environment with two MAVs. This automatic approach was able to find a viable, collision-free behavior that could also localize neighbors.

The limited amount of works show that this field is still young. The extra challenge comes from the several constraints that flow from the lower levels as well as the additional cost and difficulty of real-world experimentation. Nevertheless, there are arguments to show that automatic methods may eventually provide a way to make the most out of the swarms (Francesca and Birattari, 2016). We expect that in the future, once both MAVs as well as automatic swarming design technologies become more mature, we will begin to see an increase of (experimental) works in this domain.

## 7. FURTHER CHALLENGES AND FUTURE DEVELOPMENTS TO BE MADE

### 7.1. Battery Recharging and Scheduling

As already discussed, flight time is a fundamental constraint for MAVs. Swarming can help to expand the flight time of the whole system, such that a portion of MAVs can recharge while others are still in operation. This is subject to two main challenges. The first is the design of the combined MAV + re-charging ecosystem, and the second is the distributed scheduling between drones. Research has already begun on this front, albeit to the best of our knowledge an automated and distributed recharging method for a swarm of MAVs has yet to be demonstrated outside of a controlled environment. Toksoz et al. (2011) and Lee et al. (2015) designed a battery swapping station to quickly exchange batteries on a quadrotor. The advantage of such a system is that the battery can be changed quickly. However, it also requires

an intricate design as well as highly accurate landing to ensure that the battery is properly replaced. Instead, a contact-based re-charging station, such as the one proposed by Leonard et al. (2014) offers a simpler system, albeit at the cost of a slower turn-over. The authors investigated its use for a multi-UAV system, whereby the MAVs queued their use of the charging stations via a prioritization function. Using a similar charging system, Mulgaonkar and Kumar (2014) demonstrated a system where three quadrotors take turns to surveil a target region, such that one operates while the other two recharge. Vasile and Belta (2014) and Leahy et al. (2016) proposed formal strategies based on temporal logic constraints to ensure that the MAVs would correctly queue for recharging. However, the experimental efforts focused on the case where only one MAV operates at a given time. Nowadays, commercial charging stations are also available (Brommer et al., 2018). This will likely accelerate the research progress. Wireless charging, albeit slower, is also an attractive choice as it softens the requirement on precision landing (Choi et al., 2016; Junaid et al., 2017).

Flight time can also be increased at the MAV design level by designing MAVs with on-board recharging or longer endurance. The capability for long endurance would allow the swarm to be more flexible and take on a more diverse set of missions. One possible method to increase the flight time is to use solar cells. These have mostly been applied to fixed wing designs, such as the “Skysailor” MAV (Noth and Siegwart, 2010), benefiting from efficient flight conditions and large wing areas. It can in fact be shown that the benefit of solar cells begins to have little effect on smaller platforms, due to the reduced surface area available (Bronz et al., 2009). This trend is even more prominent on quadrotors, which have higher energy requirements. As a solution, D’Sa et al. (2016) proposed an MAV design that can alternate between fixed wing and quadrotor mode, such that “*surplus energy collected and stored while in a fixed wing configuration is utilized while in a quadrotor configuration.*” Recently, Goh et al. (2019) demonstrated a fully solar-powered quadrotor. To meet the energy requirements, an area of 4 m<sup>2</sup> was required together with a reliance on ground-effects, meaning that the MAV was bound to low altitudes. A different solution is to use combustion engines (Zufferey et al., 2013; Nex and Remondino, 2014; Ross, 2014). They benefit from the high-energy density of fuel and can help to provide long endurance flight, although they are typically applied to larger drones in outdoor environments. Alternatively, fuel cells have also been explored as a power source for long endurance flight, with increasingly promising results in the recent years (Gong and Verstraete, 2017; De Wagter et al., 2019; Pan et al., 2019).

### 7.2. Swarm-Level Active Fault Detection

Active and decentralized fault detection should also play a fundamental role for the realization of MAV swarms<sup>12</sup>. If not catered to, then there is a risk that the erroneous actions of one MAV hinder the entire swarm (Bjerknes and Winfield, 2013).

<sup>12</sup> We differentiate between fault *detection* and fault *tolerance*. Fault tolerance refers to the ability of the system to be robust to faults. Fault detection refers to the ability of the robots in the swarm to detect issues, and thus possibly also cope with them.



Winfield and Nembrini (2006) applied the Failure Mode and Effect Analysis (FMEA) methodology to evaluate the reliability of an entire swarm based on its possible failure points. From such studies it can be evaluated whether, and to what extent, local failures can incapacitate the swarm. The question is how such faults can be detected and dealt with during operations. Doing so would create a system that is more robust to failures.

Li and Parker (2007) developed the Sensor Analysis based Fault Detection (SAFDetection). In this approach, a clustering algorithm is used to learn a model of the robots' expected behavior. This model is then used to determine whether the behavior of a robot in the swarm can be considered "normal" (i.e., falls within the learned model), or "abnormal," in which case a likely fault has been detected. A distributed version of the algorithm has also been developed (Li and Parker, 2009), in which case each robot learns its own behavior model locally and then shares it. This strategy scales better with the size of the swarm, as it parallelizes the clustering computations. The works by Tarapore et al. (2013, 2015a,b) also propose a strategy for normal/abnormal behavior classification by synthesizing the behavior of neighbors within a binary feature vector. In more recent work, Tarapore et al. (2017) proposed the use of a consensus algorithm so that the robots can collectively reach a decision on whether the behavior of a team-member can be considered normal or abnormal. This was also tested on a real robotic system (Tarapore et al., 2019). Qin et al. (2014) provide a review on this active area of research. Bringing these solutions to MAV swarms can largely improve the operational safety of the full system, which is paramount for deployment in the real world.

### 7.3. Controlling and Supervising Swarms of MAVs

A control interface should enable an operator to provide commands to the swarm, such as take-off and landing, the commencement of mission objectives, or the engagement of swarm-wide emergency procedures. All should be done in a direct and intuitive way to minimize the effort by the operator (Fuchs et al., 2014; Dousse et al., 2016). To this end, Nagi et al. (2014) explored the use of a gesture vocabulary which allows a human operator to instruct a team of MAVs. The human operator and the gestures are detected directly by the MAVs using their on-board camera. Thanks to their multiple viewpoints, they are able to discern the operator's commands in a distributed fashion. Tsykunov et al. (2018) explored how to use a haptic glove to control a team of drones as if they were all connected via a spring-damper system. Research has also focused on the development of gesture languages, as in the works of Soto-Gerrero and Ramirez-Torres (2016) and Couture et al. (2018). Virtual reality is also becoming an increasingly popular technology, and is beginning to be applied to the control of MAVs (Tsykunov and Tsetserukou, 2019; Vempati et al., 2019). Besides the above, a less technical, yet highly significant, challenge to overcome on this front is the (understandably) stringent legislation surrounding MAV flight, particularly in outdoor scenarios, often requiring at least one pilot per drone (the specifics vary based on the location) (Vincenzi et al., 2015). We refer the interested reader to Hocraffer

and Nam (2017) and the sources therein for a more thorough overview of the challenges and the current technologies for human control of aerial swarms.

## 8. DISCUSSION: HOW FAR ARE WE FROM MAV SWARMS?

Following the many topics discussed in this paper comes the inevitable question: how far are we from large scale aerial swarms that can cooperatively explore areas, carry heavier objects, and autonomously complete complex tasks without low level human-in-the-loop control? Despite the large amount of research and development that has been done to tackle the topics within this grander scheme, the field of robotics and the field of swarm intelligence are both still relatively young, and there remain advances to be made. In this paper, we discussed how the swarm behavior depends on the constraints set by lower level properties, and vice versa. This interdependency and iterative nature of design means that, if we wish to bring full-fledged MAV swarms to the real world, there must be a mutual understanding between the design levels as to what is required and what can be achieved in reality.

One of the main technologies required to make the leap from flying a single MAV to flying a decentralized swarm is an accurate and reliable intra-swarm relative localization technology. Even for those applications where cooperation is limited and each member in the swarm acts mostly independently, relative localization is still needed to ensure relative collision avoidance, which is a safety-critical requirement. As we have shown throughout this paper, several technologies are currently under exploration and it is still unclear which will prove most reliable and advantageous in the long run. As the choice of these systems very directly shapes the behavior of the swarm, the challenge of designing the swarm behavior needs to be tightly coupled to it, additionally to the way it is coupled to the design of the individual robots. As such, automatic design algorithms of swarm behaviors can provide a way to make the most out of the individual MAVs and their limitations, albeit at the potential cost of relying on less well-understood controllers.

Additionally, the on-going standardization of tools is expected to help the field to reach a new level of maturity (Nedjah and Junior, 2019). Systems, such as ROS (Quigley et al., 2009), Paparazzi (Mueller and Drouin, 2007; Brisset and Hattenberger, 2008), or PX4 (Meier et al., 2015) have now accelerated the process of prototyping and testing on real-world MAVs, and have also made it easier to share hardware/software advancements. Low cost programmable MAVs, such as the Crazyflie are also available, making it more feasible to experiment with large numbers of MAVs. Additionally, dedicated standards, such as MAVLink, which provides communication between software modules, are becoming increasingly popular (Dietrich et al., 2016), and full-stack frameworks have been developed to handle the entire pipeline (Sanchez-Lopez et al., 2016; Millan-Romera et al., 2019). The combination of these systems together with simulators, such as the well-known Gazebo (Koenig and Howard, 2004), ARGoS (Pinciroli et al., 2012), or AirSim (Shah

et al., 2018), further help to quickly prototype software in a realistic simulation environment. Combined with models and frameworks, such as hector-quadrotor (Meyer et al., 2012) or RotorS (Furrer et al., 2016), simulation environments can significantly accelerate the development time (Johnson and Mishra, 2002). Mairaj et al. (2019) provides an extensive review of several simulators for this purpose. Dedicated swarm languages, such as Buzz (Pinciroli and Beltrame, 2016) also provide a simpler prototyping framework dedicated to swarm robotics, which can also be applied to MAVs.

Finally, the prominent rise in popularity of MAVs in the last decade has brought about several technology accelerators. MAV focused robotics competitions, such as the Mohamed Bin Zayed International Robotics Challenge (MBZIRC) or the International Micro Air Vehicle (IMAV) competition have now also begun to integrate swarming or multi-robot elements (Pestana et al., 2014; Saska et al., 2016a; Bähnemann et al., 2017; Nieuwenhuisen et al., 2017; Spurný et al., 2019). This pushes researchers to take a technology out of the lab and into unknown environments, thereby increasing their robustness.

## REFERENCES

- Abeywardena, D., Kodagoda, S., Dissanayake, G., and Munasinghe, R. (2013). Improved state estimation in quadrotor MAVs: a novel drift-free velocity estimator. *IEEE Robot. Autom. Mag.* 20, 32–39. doi: 10.1109/MRA.2012.2225472
- Achtelik, M., Achtelik, M., Brunet, Y., Chli, M., Chatzichristofis, S., Decotignie, J., et al. (2012). “SFly: swarm of micro flying robots,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Vilamoura), 2649–2650. doi: 10.1109/IROS.2012.6386281
- Achtelik, M. W., Weiss, S., Chli, M., Dellaert, F., and Siegwart, R. (2011). “Collaborative stereo,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (San Francisco, CA), 2242–2248. doi: 10.1109/IROS.2011.6094866
- Afzal, M. H., Renaudin, V., and Lachapelle, G. (2011). “Magnetic field based heading estimation for pedestrian navigation environments,” in *2011 International Conference on Indoor Positioning and Indoor Navigation (Guimaraes)*, 1–10. doi: 10.1109/IPIN.2011.6071947
- Aguilar, W. G., Casaliglla, V. P., and Pólit, J. L. (2017). “Obstacle avoidance for low-cost UAVs,” in *2017 IEEE 11th International Conference on Semantic Computing (ICSC)* (San Diego, CA), 503–508. doi: 10.1109/ICSC.2017.96
- Alexis, K., Nikolakopoulos, G., Tzes, A., and Dritsas, L. (2009). “Coordination of helicopter UAVs for aerial forest-fire surveillance,” in *Applications of Intelligent Control to Engineering Systems*, ed K. P. Valavanis (Dordrecht: Springer Netherlands), 169–193.
- Allred, J., Hasan, A. B., Panichsakul, S., Pisano, W., Gray, P., Huang, J., et al. (2007). “Sensorflock: an airborne wireless sensor network of micro-air vehicles,” in *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems, SenSys '07* (New York, NY: Association for Computing Machinery), 117–129.
- Alonso-Mora, J., Nägeli, T., Siegwart, R., and Beardsley, P. (2015). Collision avoidance for aerial vehicles in multi-agent scenarios. *Auton. Robots* 39, 101–121. doi: 10.1007/s10514-015-9429-0
- Alvarez, H., Paz, L. M., Sturm, J., and Cremers, D. (2016). *Collision Avoidance for Quadrotors with a Monocular Camera*. Cham: Springer International Publishing, 195–209.
- Ampatzis, C., Tuci, E., Trianni, V., and Dorigo, M. (2008). Evolution of signaling in a multi-robot system: categorization and communication. *Adapt. Behav.* 16, 5–26. doi: 10.1177/1059712307087282

## 9. CONCLUSION

The challenges to solve before we can expect to see swarms of autonomous MAVs are many. They begin at the lowest level, forcing us to think of how the MAV design will impact the swarm behavior, and they end at the highest level, where we must design collective behaviors that best exploit our lower level designs, controllers, and sensors. In the last decade, the field of swarm robotics and MAV design have started to merge more and more, leading to increasingly impressive achievements. To go further, the tight and complex relationship between the low level and the high level needs to be appreciated in order to break into a new era of truly autonomous and distributed swarms of MAVs.

## AUTHOR CONTRIBUTIONS

This article has been set up based on discussions between all authors. It has been primarily written by MC with the additional help and advice of KM, CD, and GC.

- Aoki, I. (1982). A simulation study on the schooling mechanism in fish. *Nippon Suisan Gakkaishi* 48, 1081–1088.
- Augugliaro, F., Lupashin, S., Hamer, M., Male, C., Hehn, M., Mueller, M. W., et al. (2014). The flight assembled architecture installation: cooperative construction with flying machines. *IEEE Control Syst. Mag.* 34, 46–64. doi: 10.1109/MCS.2014.2320359
- Bachrach, A., Prentice, S., He, R., and Roy, N. (2011). RANGE-robust autonomous navigation in GPS-denied environments. *J. Field Robot.* 28, 644–666. doi: 10.1002/rob.20400
- Bähnemann, R., Schindler, D., Kamel, M., Siegwart, R., and Nieto, J. (2017). “A decentralized multi-agent unmanned aerial system to search, pick up, and relocate objects,” in *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)* (Shanghai), 123–128. doi: 10.1109/SSRR.2017.8088150
- Basiri, M. (2015). *Audio-based positioning and target localization for swarms of micro aerial vehicles* (Ph.D. thesis), École polytechnique fédérale de Lausanne, Lausanne. doi: 10.5075/epfl-thesis-6508
- Basiri, M., Schill, F., Floreano, D., and Lima, P. U. (2014). “Audio-based localization for swarms of micro air vehicles,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (Hong Kong), 4729–4734. doi: 10.1109/ICRA.2014.6907551
- Basiri, M., Schill, F., Lima, P., and Floreano, D. (2016). On-board relative bearing estimation for teams of drones using sound. *IEEE Robot. Autom. Lett.* 1, 820–827. doi: 10.1109/LRA.2016.2527833
- Beard, R. W. (2007). “State estimation for micro air vehicles,” in *Innovations in Intelligent Machines-1*, eds J. S. Chahl, L. C. Jain, A. Mizutani, and M. Sato-Ilic (Berlin; Heidelberg: Springer Berlin Heidelberg), 173–199.
- Bedau, M. A. (2003). Artificial life: organization, adaptation and complexity from the bottom up. *Trends Cogn. Sci.* 7, 505–512. doi: 10.1016/j.tics.2003.09.012
- Bensky, A. (2019). *Short-Range Wireless Communication*. Cambridge, MA: Elsevier. doi: 10.1016/c2017-0-02356-x
- Beyeler, A., Zufferey, J.-C., and Floreano, D. (2009). Vision-based control of near-obstacle flight. *Auton. Robots* 27:201. doi: 10.1007/s10514-009-9139-6
- Bitcraze, A. B. (2019). *Multi-Ranger Deck*. Available online at: <https://www.bitcraze.io/multi-ranger-deck/>
- Bjerknes, J. D., and Winfield, A. F. T. (2013). *On Fault Tolerance and Scalability of Swarm Robotic Systems*. Berlin; Heidelberg: Springer Berlin Heidelberg, 431–444.
- Bonabeau, E., and Théraulaz, G. (2008). Swarm smarts. *Sci. Am.* 18, 40–47. doi: 10.1038/scientificamerican0208-40sp

- Bouabdallah, S., Murrieri, P., and Siegwart, R. (2004). "Design and control of an indoor micro quadrotor," in *IEEE International Conference on Robotics and Automation (ICRA)*, Vol. 5 (New Orleans, LA), 4393–4398. doi: 10.1109/ROBOT.2004.1302409
- Bouabdallah, S., and Siegwart, R. (2007). "Full control of a quadrotor," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (San Diego, CA), 153–158. doi: 10.1109/IROS.2007.4399042
- Bouffanais, R. (2016). *Design and Control of Swarm Dynamics*. Singapore: Springer Singapore.
- Brambilla, M., Ferrante, E., Birattari, M., and Dorigo, M. (2013). Swarm robotics: a review from the swarm engineering perspective. *Swarm Intell.* 7, 1–41. doi: 10.1007/s11721-012-0075-2
- Bredeche, N., Haasdijk, E., and Prieto, A. (2018). Embodied evolution in collective robotics: a review. *Front. Robot. AI* 5:12. doi: 10.3389/frobot.2018.00012
- Briod, A., Klaptocz, A., Zufferey, J., and Floreano, D. (2012). "The AirBurr: a flying robot that can exploit collisions," in *2012 ICME International Conference on Complex Medical Engineering (CME)* (Kobe), 569–574. doi: 10.1109/ICME.2012.6275674
- Briod, A., Zufferey, J.-C., and Floreano, D. (2013). "Optic-flow based control of a 46g quadrotor," in *Workshop on Vision-Based Closed-Loop Control and Navigation of Micro Helicopters in GPS-Denied Environments, IROS 2013* (Tokyo).
- Brisset, P., and Hattenberger, G. (2008). "Multi-UAV control with the paparazzi system," in *Conference on Humans Operating Unmanned Systems (HUMOUS)* (Brest).
- Brommer, C., Malyuta, D., Hentzen, D., and Brockers, R. (2018). "Long-duration autonomy for small rotorcraft as including recharging," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid), 7252–7258. doi: 10.1109/IROS.2018.8594111
- Bronz, M., Moschetta, J. M., Brisset, P., and Gorraz, M. (2009). Towards a long endurance MAV. *Int. J. Micro Air Vehicles* 1, 241–254. doi: 10.1260/175682909790291483
- Brutschy, A., Garattoni, L., Brambilla, M., Francesca, G., Pini, G., Dorigo, M., et al. (2015). The TAM: abstracting complex tasks in swarm robotics research. *Swarm Intell.* 9, 1–22. doi: 10.1007/s11721-014-0102-6
- Bry, A., Richter, C., Bachrach, A., and Roy, N. (2015). Aggressive flight of fixed-wing and quadrotor aircraft in dense indoor environments. *Int. J. Robot. Res.* 34, 969–1002. doi: 10.1177/0278364914558129
- Busoniu, L., Babuska, R., and De Schutter, B. (2008). A comprehensive survey of multiagent reinforcement learning. *IEEE Trans. Syst. Man Cybernet. C Appl. Rev.* 38, 156–172. doi: 10.1109/TSMCC.2007.913919
- Cabrera-Ponce, A. A., Martinez-Carranza, J., and Rascon, C. (2019). "Detection of nearby UAVs using CNN and spectrograms," in *International Micro Air Vehicle Conference and Competition (IMAV)* (Madrid).
- Cadell, C. (2018). *Flight of Imagination: Chinese Firm Breaks Record With 1,374 Dancing Drones*. Reuters. Available online at: [www.reuters.com/article/us-china-drones/flight-of-imagination-chinese-firm-breaks-record-with-1374-dancing-drones-idUSKBN1I3189](http://www.reuters.com/article/us-china-drones/flight-of-imagination-chinese-firm-breaks-record-with-1374-dancing-drones-idUSKBN1I3189)
- Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., et al. (2016). Past, present, and future of simultaneous localization and mapping: toward the robust-perception age. *IEEE Trans. Robot.* 32, 1309–1332. doi: 10.1109/TRO.2016.2624754
- Campion, M., Ranganathan, P., and Faruque, S. (2018). "A review and future directions of UAV swarm communication architectures," in *2018 IEEE International Conference on Electro/Information Technology (EIT)* (Rochester, MI), 0903–0908. doi: 10.1109/EIT.2018.8500274
- Carrio, A., Bavle, H., and Campoy, P. (2018). Attitude estimation using horizon detection in thermal images. *Int. J. Micro Air Vehicles* 10, 352–361. doi: 10.1177/1756829318804761
- Carrio, A., Vemprala, S., Ripoll, A., Saripalli, S., and Campoy, P. (2018). "Drone detection using depth maps," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid), 1034–1037. doi: 10.1109/IROS.2018.8593405
- Chamanbaz, M., Mateo, D., Zoss, B. M., Tokić, G., Wilhelm, E., Bouffanais, R., et al. (2017). Swarm-enabling technology for multi-robot systems. *Front. Robot. AI* 4:12. doi: 10.3389/frobot.2017.00012
- Chen, Y., Zhao, H., Mao, J., Chirarattananon, P., Helbling, E. F., Hyun, N.-S. P., et al. (2019). Controlled flight of a microrobot powered by soft artificial muscles. *Nature* 575, 324–329. doi: 10.1038/s41586-019-1737-7
- Choi, C. H., Jang, H. J., Lim, S. G., Lim, H. C., Cho, S. H., and Gaponov, I. (2016). "Automatic wireless drone charging station creating essential environment for continuous drone operation," in *2016 International Conference on Control, Automation and Information Sciences (ICCAIS)* (Ansan), 132–136. doi: 10.1109/ICCAIS.2016.7822448
- Chung, S., Paranjape, A. A., Dames, P., Shen, S., and Kumar, V. (2018). A survey on aerial swarm robotics. *IEEE Trans. Robot.* 34, 837–855. doi: 10.1109/TRO.2018.2857475
- Chung, T. H., Clement, M. R., Day, M. A., Jones, K. D., Davis, D., and Jones, M. (2016). "Live-fly, large-scale field experimentation for large numbers of fixed-wing UAVs," in *2016 IEEE International Conference on Robotics and Automation (ICRA)* (Stockholm), 1255–1262. doi: 10.1109/ICRA.2016.7487257
- Cieslewski, T., Choudhary, S., and Scaramuzza, D. (2018). "Data-efficient decentralized visual slam," in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (Brisbane, QLD), 2466–2473. doi: 10.1109/ICRA.2018.8461155
- Cieslewski, T., and Scaramuzza, D. (2017). "Efficient decentralized visual place recognition from full-image descriptors," in *2017 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)* (Los Angeles, CA), 78–82. doi: 10.1109/MRS.2017.8250934
- Cognetti, M., Stegagno, P., Franchi, A., Oriolo, G., and Bühlhoff, H. H. (2012). "3-D mutual localization with anonymous bearing measurements," in *2012 IEEE International Conference on Robotics and Automation* (Saint Paul, MN), 791–798. doi: 10.1109/ICRA.2012.6225288
- Collotta, M., Pau, G., Talty, T., and Tonguz, O. K. (2018). Bluetooth 5: a concrete step forward toward the IoT. *IEEE Commun. Mag.* 56, 125–131. doi: 10.1109/MCOM.2018.1700053
- Coppola, M., Guo, J., Gill, E., and de Croon, G. C. H. E. (2019). The pagerank algorithm as a method to optimize swarm behavior through local analysis. *Swarm Intell.* 13, 277–319. doi: 10.1007/s11721-019-00172-z
- Coppola, M., McGuire, K. N., Scheper, K. Y. W., and de Croon, G. C. H. E. (2018). On-board communication-based relative localization for collision avoidance in micro air vehicle teams. *Auton. Robots* 42, 1787–1805. doi: 10.1007/s10514-018-9760-3
- Couture, N., Bottecchia, S., Chaumette, S., Ceconello, M., Rekalde, J., and Desainte-Catherine, M. (2018). "Using the soundpainting language to fly a swarm of drones," in *Advances in Human Factors in Robots and Unmanned Systems*, ed J. Chen (Cham: Springer International Publishing), 39–51.
- Cunningham, A., Indelman, V., and Dellaert, F. (2013). "Ddf-sam 2.0: consistent distributed smoothing and mapping," in *2013 IEEE International Conference on Robotics and Automation* (Karlsruhe), 5220–5227. doi: 10.1109/ICRA.2013.6631323
- de Croon, G. C. H. E., de Weerd, E., De Wagter, C., Remes, B. D. W., and Ruijsink, R. (2012). The appearance variation cue for obstacle avoidance. *IEEE Trans. Robot.* 28, 529–534. doi: 10.1109/TRO.2011.2170754
- de Croon, G. C. H. E., Perçin, M., Remes, B. D. W., Ruijsink, R., and Wagter, C. D. (2016). *The DelFly*. Dordrecht: Springer Netherlands. doi: 10.1007/978-94-017-9208-0
- de Croon, G. C. H. E., Van Dartel, M. F., and Postma, E. O. (2005). "Evolutionary learning outperforms reinforcement learning on non-markovian tasks," in *Workshop on Memory and Learning Mechanisms in Autonomous Robots, 8th European Conference on Artificial Life* (Canterbury).
- de Croon, G. C. H. E., Wagter, C. D., Remes, B. D. W., and Ruijsink, R. (2012). Sub-sampling: real-time vision for micro air vehicles. *Robot. Auton. Syst.* 60, 167–181. doi: 10.1016/j.robot.2011.10.001
- de Marina, H. G., and Smeur, E. (2019). "Flexible collaborative transportation by a team of rotorcraft," in *2019 International Conference on Robotics and Automation (ICRA)* (Montreal, QC), 1074–1080. doi: 10.1109/ICRA.2019.8794316
- de Marina, H. G., Sun, Z., Bronz, M., and Hattenberger, G. (2017). "Circular formation control of fixed-wing UAVs with constant speeds," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Vancouver, BC), 5298–5303. doi: 10.1109/IROS.2017.8206422

- De Wagter, C., Remes, B., Ruisink, R., van Tienen, F., and van der Horst, E. (2019). "Design and testing of a vertical take-off and landing uav optimized for carrying a hydrogen fuel-cell with pressure tank," in *International Micro Air Vehicle Conference and Competition (IMAV)* (Madrid).
- Dietrich, T., Andreyev, O., Zimmermann, A., and Mitschele-Thiel, A. (2016). "Towards a unified decentralized swarm management and maintenance coordination based on MAVLink," in *2016 International Conference on Autonomous Robot Systems and Competitions (ICARSC)* (Braganca), 124–129. doi: 10.1109/ICARSC.2016.64
- Doer, C., Scholz, G., and Trommer, G. F. (2017). Indoor laser-based slam for micro aerial vehicles. *Gyros. Navig.* 8, 181–189. doi: 10.1134/S2075108717030038
- Dousse, N., Heitz, G., and Floreano, D. (2016). Extension of a ground control interface for swarms of small drones. *Artif. Life Robot.* 21, 308–316. doi: 10.1007/s10015-016-0302-9
- D'Sa, R., Jenson, D., Henderson, T., Kilian, J., Schulz, B., Calvert, M., et al. (2016). "SUAV:Q—an improved design for a transformable solar-powered UAV," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Daejeon), 1609–1615. doi: 10.1109/IROS.2016.7759260
- Duarte, M., Costa, V., Gomes, J., Rodrigues, T., Silva, F., Oliveira, S. M., et al. (2016). Evolution of collective behaviors for a real swarm of aquatic surface robots. *PLoS ONE* 11:e151834. doi: 10.1371/journal.pone.0151834
- Dupeyroux, J., Serres, J. R., and Viollet, S. (2019). AntBot: a six-legged walking robot able to home like desert ants in outdoor environments. *Sci. Robot.* 4:eau0307. doi: 10.1126/scirobotics.aau0307
- Dusha, D., Mejias, L., and Walker, R. (2011). Fixed-wing attitude estimation using temporal tracking of the horizon and optical flow. *J. Field Robot.* 28, 355–372. doi: 10.1002/rob.20387
- Eiben, A. E. (2014). Grand challenges for evolutionary robotics. *Front. Robot. AI* 1:4. doi: 10.3389/frobot.2014.00004
- Engelen, S., Gill, E., and Verhoeven, C. (2014). On the reliability, availability, and throughput of satellite swarms. *IEEE Trans. Aerospace Electron. Syst.* 50, 1027–1037. doi: 10.1109/TAES.2014.120711
- Epstein, D., and Feldman, D. (2018). Quadcopter tracks quadcopter via real-time shape fitting. *IEEE Robot. Autom. Lett.* 3, 544–550. doi: 10.1109/LRA.2017.2773668
- Faessler, M., Fontana, F., Forster, C., and Scaramuzza, D. (2015). "Automatic re-initialization and failure recovery for aggressive flight with a monocular vision-based quadrotor," in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (Seattle, WA), 1722–1729. doi: 10.1109/ICRA.2015.7139420
- Faessler, M., Mueggler, E., Schwabe, K., and Scaramuzza, D. (2014). "A monocular pose estimation system based on infrared leds," in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (Hong Kong), 907–913. doi: 10.1109/ICRA.2014.6906962
- Faigl, J., Krajník, T., Chudoba, J., Přečůl, L., and Saska, M. (2013). "Low-cost embedded system for relative localization in robotic swarms," in *2013 IEEE International Conference on Robotics and Automation* (Karlsruhe), 993–998. doi: 10.1109/ICRA.2013.6630694
- Falanga, D., Kim, S., and Scaramuzza, D. (2019a). How fast is too fast? The role of perception latency in high-speed sense and avoid. *IEEE Robot. Autom. Lett.* 4, 1884–1891. doi: 10.1109/LRA.2019.2898117
- Falanga, D., Kleber, K., Mintchev, S., Floreano, D., and Scaramuzza, D. (2019b). The foldable drone: a morphing quadrotor that can squeeze and fly. *IEEE Robot. Autom. Lett.* 4, 209–216. doi: 10.1109/LRA.2018.2885575
- Faludi, R. (2010). *Building Wireless Sensor Networks: With ZigBee, XBee, Arduino, and Processing*. Sebastopol, CA: O'Reilly Media, Inc.
- Ferrante, E., Duéñez Guzmán, E., Turgut, A. E., and Wenseleers, T. (2013). "GESwarm: grammatical evolution for the automatic synthesis of collective behaviors in swarm robotics," in *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation, GECCO '13* (New York, NY: ACM), 17–24.
- Fiorini, P., and Shiller, Z. (1998). Motion planning in dynamic environments using velocity obstacles. *Int. J. Robot. Res.* 17, 760–772.
- Flocchini, P., Prencipe, G., Santoro, N., and Widmayer, P. (2008). Arbitrary pattern formation by asynchronous, anonymous, oblivious robots. *Theor. Comput. Sci.* 407, 412–447. doi: 10.1016/j.tcs.2008.07.026
- Floreano, D., Mintchev, S., and Shintake, J. (2017). "Foldable drones: from biology to technology," in *Bioinspiration, Biomimetics, and Bioreplication 2017*, Vol. 10162, eds M. Knez, A. Lakhtakia, and R. J. Martin-Palma (Portland, OR: International Society for Optics and Photonics, SPIE), 1–6. doi: 10.1117/12.2259931
- Floreano, D., Pericet-Camara, R., Viollet, S., Ruffier, F., Brückner, A., Leitl, R., et al. (2013). Miniature curved artificial compound eyes. *Proc. Natl. Acad. Sci. U.S.A.* 110, 9267–9272. doi: 10.1073/pnas.1219068110
- Floreano, D., and Wood, R. J. (2015). Science, technology and the future of small autonomous drones. *Nature* 521:460. doi: 10.1038/nature14542
- Forster, C., Lynen, S., Kneip, L., and Scaramuzza, D. (2013). "Collaborative monocular slam with multiple micro aerial vehicles," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Tokyo), 3962–3970. doi: 10.1109/IROS.2013.6696923
- Francesca, G., and Birattari, M. (2016). Automatic design of robot swarms: achievements and challenges. *Front. Robot. AI* 3:29. doi: 10.3389/frobot.2016.00029
- Francesca, G., Brambilla, M., Brutschy, A., Garattoni, L., Miletich, R., Podevijn, G., et al. (2015). Automode-chocolate: automatic design of control software for robot swarms. *Swarm Intell.* 9, 125–152. doi: 10.1007/s11721-015-0107-9
- Francesca, G., Brambilla, M., Brutschy, A., Trianni, V., and Birattari, M. (2014). Automode: a novel approach to the automatic design of control software for robot swarms. *Swarm Intell.* 8, 89–112. doi: 10.1007/s11721-014-0092-4
- Franchi, A., Oriolo, G., and Stegagno, P. (2013). Mutual localization in multi-robot systems using anonymous relative measurements. *Int. J. Robot. Res.* 32, 1302–1322. doi: 10.1177/0278364913495425
- Fuchs, C., Borst, C., de Croon, G. C. H. E., van Paassen, M. M. R., and Mulder, M. (2014). An ecological approach to the supervisory control of UAV swarms. *Int. J. Micro Air Vehicles* 6, 211–229. doi: 10.1260/1756-8293.6.4.211
- Furrer, F., Burri, M., Achtelik, M., and Siegwart, R. (2016). *RotorS—A Modular Gazebo MAV Simulator Framework*. Cham: Springer International Publishing, 595–625.
- Gabrich, B., Saldaña, D., Kumar, V., and Yim, M. (2018). "A flying gripper based on cuboid modular robots," in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (Brisbane, QLD), 7024–7030. doi: 10.1109/ICRA.2018.8460682
- Gandhi, D., Pinto, L., and Gupta, A. (2017). "Learning to fly by crashing," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Vancouver, BC), 3948–3955. doi: 10.1109/IROS.2017.8206247
- Garnier, S., Gautrais, J., and Theraulaz, G. (2007). The biological principles of swarm intelligence. *Swarm Intell.* 1, 3–31. doi: 10.1007/s11721-007-0004-y
- Gazi, V. (2005). Swarm aggregations using artificial potentials and sliding-mode control. *IEEE Trans. Robot.* 21, 1208–1214. doi: 10.1109/TRO.2005.853487
- Gazi, V., and Passino, K. M. (2002). "A class of attraction/repulsion functions for stable swarm aggregations," in *Proceedings of the 41st IEEE Conference on Decision and Control* Vol. 3 (Las Vegas, NV), 2842–2847. doi: 10.1109/CDC.2002.1184277
- Gazi, V., and Passino, K. M. (2004). Stability analysis of social foraging swarms. *IEEE Trans. Syst. Man Cybernet. B Cybernet.* 34, 539–557. doi: 10.1109/TSMCB.2003.817077
- Gazi, V., and Passino, K. M. (2011). *Swarm Stability and Optimization*. Berlin; Heidelberg: Springer. doi: 10.1007/978-3-642-18041-5
- Ghadiok, V., Goldin, J., and Ren, W. (2012). On the design and development of attitude stabilization, vision-based navigation, and aerial gripping for a low-cost quadrotor. *Auton. Robots* 33, 41–68. doi: 10.1007/s10514-012-9286-z
- Giernacki, W., Skwarczyński, M., Witwicki, W., Wroński, P., and Kozierski, P. (2017). "Crazyflie 2.0 quadrotor as a platform for research and education in robotics and control engineering," in *22nd International Conference on Methods and Models in Automation and Robotics (MMAR)* (Miedzyzdroje), 37–42. doi: 10.1109/MMAR.2017.8046794
- Goh, C. S., Kuan, J. R., Yeo, J. H., Teo, B. S., and Danner, A. (2019). A fully solar-powered quadcopter able to achieve controlled flight out of the ground effect. *Prog. Photovolt. Res. Appl.* 27, 869–878. doi: 10.1002/ppp.3169
- Gomes, J., Urbano, P., and Christensen, A. L. (2013). Evolution of swarm robotics systems with novelty search. *Swarm Intell.* 7, 115–144. doi: 10.1007/s11721-013-0081-z

- Gong, A., and Verstraete, D. (2017). Fuel cell propulsion in small fixed-wing unmanned aerial vehicles: Current status and research needs. *Int. J. Hydrogen Energy* 42, 21311–21333. doi: 10.1016/j.ijhydene.2017.06.148
- Green, W. E., and Oh, P. Y. (2006). “Autonomous hovering of a fixed-wing micro air vehicle,” in *Proceedings 2006 IEEE International Conference on Robotics and Automation (ICRA)* (Orlando, FL), 2164–2169. doi: 10.1109/ROBOT.2006.1642024
- Grondin, F., Létourneau, D., Ferland, F., Rousseau, V., and Michaud, F. (2013). The ManyEars open framework. *Auton. Robots* 34, 217–232. doi: 10.1007/s10514-012-9316-x
- Grzonka, S., Grisetti, G., and Burgard, W. (2009). “Towards a navigation system for autonomous indoor flying,” in *2009 IEEE International Conference on Robotics and Automation* (Kobe), 2878–2883. doi: 10.1109/ROBOT.2009.5152446
- Guinness World Records (2018). *Most Unmanned Aerial Vehicles (UAVs) Airborne Simultaneously*. Available online at: [www.guinnessworldrecords.com/world-records/373319-most-C2%A0unmanned-aerial-C2%A0vehicles-C2%A0uavs-airborne-C2%A0simultaneously-C2%A05-kg-or-less](http://www.guinnessworldrecords.com/world-records/373319-most-C2%A0unmanned-aerial-C2%A0vehicles-C2%A0uavs-airborne-C2%A0simultaneously-C2%A05-kg-or-less)
- Guinness World Records (2019). *Most Unmanned Aerial Vehicles (UAVs) Airborne Simultaneously From a Single Computer (Indoors)*. Available online at: [www.guinnessworldrecords.com/world-records/507534-most-uavs-controlled-from-a-single-computer](http://www.guinnessworldrecords.com/world-records/507534-most-uavs-controlled-from-a-single-computer)
- Guo, K., Qiu, Z., Meng, W., Xie, L., and Teo, R. (2017). Ultra-wideband based cooperative relative localization algorithm and experiments for multiple unmanned aerial vehicles in GPS denied environments. *Int. J. Micro Air Vehicles* 9, 169–186. doi: 10.1177/1756829317695564
- Guo, K., Qiu, Z., Miao, C., Zaini, A. H., Chen, C.-L., Meng, W., et al. (2016). Ultra-wideband-based localization for quadcopter navigation. *Unmanned Syst.* 4, 23–34. doi: 10.1142/S2301385016400033
- Gupte, S., Mohandas, P. I. T., and Conrad, J. M. (2012). “A survey of quadrotor unmanned aerial vehicles,” in *Proceedings of IEEE Southeastcon* (Orlando, FL: IEEE), 1–6. doi: 10.1109/secon.2012.6196930
- Guzzi, J., Giusti, A., Gambardella, L. M., and Di Caro, G. A. (2013a). “Local reactive robot navigation: a comparison between reciprocal velocity obstacle variants and human-like behavior,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Tokyo), 2622–2629. doi: 10.1109/IROS.2013.6696726
- Guzzi, J., Giusti, A., Gambardella, L. M., and Di Caro, G. A. (2014). “Bioinspired obstacle avoidance algorithms for robot swarms,” in *Bio-Inspired Models of Network, Information, and Computing Systems*, eds G. A. Di Caro and G. Theraulaz (Cham: Springer International Publishing), 120–134.
- Guzzi, J., Giusti, A., Gambardella, L. M., Theraulaz, G., and Di Caro, G. A. (2013b). “Human-friendly robot navigation in dynamic environments,” in *2013 IEEE International Conference on Robotics and Automation* (Karlsruhe), 423–430. doi: 10.1109/ICRA.2013.6630610
- Hamann, H. (2018). *Swarm Robotics: A Formal Approach*. Cham: Springer International Publishing. doi: 10.1007/978-3-319-74528-2
- Hauert, S., Leven, S., Varga, M., Ruini, F., Cangelosi, A., Zufferey, J., et al. (2011). “Reynolds flocking in reality with fixed-wing robots: Communication range vs. maximum turning rate,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (San Francisco, CA), 5015–5020. doi: 10.1109/IROS.2011.6095129
- Hauert, S., Zufferey, J.-C., and Floreano, D. (2009). Evolved swarming without positioning information: an application in aerial communication relay. *Auton. Robots* 26, 21–32. doi: 10.1007/s10514-008-9104-9
- Heng, L., Meier, L., Tanskanen, P., Fraundorfer, F., and Pollefeys, M. (2011). “Autonomous obstacle avoidance and maneuvering on a vision-guided map using on-board processing,” in *2011 IEEE International Conference on Robotics and Automation* (Shanghai), 2472–2477. doi: 10.1109/ICRA.2011.5980095
- Ho, H. W., de Croon, G. C. H. E., and Chu, Q. (2017). Distance and velocity estimation using optical flow from a monocular camera. *Int. J. Micro Air Vehicles* 9, 198–208. doi: 10.1177/1756829317695566
- Hocraffer, A., and Nam, C. S. (2017). A meta-analysis of human-system interfaces in unmanned aerial vehicle (UAV) swarm management. *Appl. Ergonom.* 58, 66–80. doi: 10.1016/j.apergo.2016.05.011
- Honegger, D., Meier, L., Tanskanen, P., and Pollefeys, M. (2013). “An open source and open hardware embedded metric optical flow CMOS camera for indoor and outdoor applications,” in *2013 IEEE International Conference on Robotics and Automation (ICRA)* (Karlsruhe), 1736–1741. doi: 10.1109/ICRA.2013.6630805
- Huang, A. S., Bachrach, A., Henry, P., Krainin, M., Maturana, D., Fox, D., et al. (2017). *Visual Odometry and Mapping for Autonomous Flight Using an RGB-D Camera*. Cham: Springer International Publishing, 235–252.
- Hüttenrauch, M., Šošić, A., and Neumann, G. (2019). Deep reinforcement learning for swarm systems. *J. Mach. Learn. Res.* 20, 1–31.
- Ishii, S., Fujita, H., Mitsutake, M., Yamazaki, T., Matsuda, J., and Matsuno, Y. (2005). A reinforcement learning scheme for a partially-observable multi-agent game. *Mach. Learn.* 59, 31–54. doi: 10.1007/s10994-005-0461-8
- Itasse, M., Moschetta, J.-M., Ameho, Y., and Carr, R. (2011). Equilibrium transition study for a hybrid MAV. *Int. J. Micro Air Vehicles* 3, 229–245. doi: 10.1260/1756-8293.3.4.229
- Izzo, D., and Pettazzi, L. (2007). Autonomous and distributed motion planning for satellite swarm. *J. Guidance Control Dyn.* 30, 449–459. doi: 10.2514/1.22736
- Izzo, D., Simões, L. F., and de Croon, G. C. H. E. (2014). An evolutionary robotics approach for the distributed control of satellite formations. *Evol. Intell.* 7, 107–118. doi: 10.1007/s12065-014-0111-9
- Johnson, E., and Mishra, S. (2002). “Flight simulation for the development of an experimental UAV,” in *AIAA Modeling and Simulation Technologies Conference and Exhibit* (Monterey, CA). doi: 10.2514/6.2002-4975
- Jones, S., Studley, M., Hauert, S., and Winfield, A. (2018). *Evolving Behaviour Trees for Swarm Robotics*. Cham: Springer International Publishing, 487–501.
- Jones, S., Winfield, A. F., Hauert, S., and Studley, M. (2019). Onboard evolution of understandable swarm behaviors. *Adv. Intell. Syst.* 1:1900031. doi: 10.1002/aisy.201900031
- Junaid, A. B., Konoiko, A., Zweiri, Y., Sahinkaya, M. N., and Seneviratne, L. (2017). Autonomous wireless self-charging for multi-rotor unmanned aerial vehicles. *Energies* 10:803. doi: 10.3390/en10060803
- Jung, S., Liu, C., and Ariyur, K. B. (2013). “Absolute orientation for a uav using celestial objects,” in *AIAA Infotech@Aerospace (I@A) Conference* (Boston, MA).
- Karásek, M., Muijres, F. T., De Wagter, C., Remes, B. D. W., and de Croon, G. C. H. E. (2018). A tailless aerial robotic flapper reveals that flies use torque coupling in rapid banked turns. *Science* 361, 1089–1094. doi: 10.1126/science.aat0350
- Kendoul, F., Fantoni, I., and Nonami, K. (2009a). Optic flow-based vision system for autonomous 3D localization and control of small aerial vehicles. *Robot. Auton. Syst.* 57, 591–602. doi: 10.1016/j.robot.2009.02.001
- Kendoul, F., Nonami, K., Fantoni, I., and Lozano, R. (2009b). An adaptive vision-based autopilot for mini flying machines guidance, navigation and control. *Auton. Robots* 27, 165–188. doi: 10.1007/s10514-009-9135-x
- Koenig, N., and Howard, A. (2004). “Design and use paradigms for Gazebo, an open-source multi-robot simulator,” in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, Vol. 3 (Sendai), 2149–2154. doi: 10.1109/IROS.2004.1389727
- Kong, L., Sheng, J., and Teredesai, A. (2014). “Basic micro-aerial vehicles (MAVs) obstacles avoidance using monocular computer vision,” in *2014 13th International Conference on Control Automation Robotics Vision (ICARCV)* (Singapore), 1051–1056. doi: 10.1109/ICARCV.2014.7064451
- Kornatowski, P. M., Mintchev, S., and Floreano, D. (2017). “An origami-inspired cargo drone,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Vancouver, BC), 6855–6862. doi: 10.1109/IROS.2017.8206607
- Krajník, T., Nitsche, M., Faigl, J., Vaněk, P., Saska, M., Přeučil, L., et al. (2014). A practical multirobot localization system. *J. Intell. Robot. Syst.* 76, 539–562. doi: 10.1007/s10846-014-0041-x
- Krajník, T., Vonásek, V., Fišer, D., and Faigl, J. (2011). “AR-Drone as a platform for robotic research and education,” in *Research and Education in Robotics-EUROBOT 2011*, eds D. Obržálek and A. Gottscheber (Berlin; Heidelberg: Springer Berlin Heidelberg), 172–186.
- Kuckling, J., Ligot, A., Bozhinoski, D., and Birattari, M. (2018). “Behavior trees as a control architecture in the automatic modular design of robot swarms,” in *Swarm Intelligence*, eds M. Dorigo, M. Birattari, C. Blum, A. L. Christensen, A. Reina, and V. Trianni (Cham: Springer International Publishing), 30–43.
- Kumar, V., and Michael, N. (2012). Opportunities and challenges with autonomous micro aerial vehicles. *Int. J. Robot. Res.* 31, 1279–1291. doi: 10.1177/0278364912455954

- Kushleyev, A., Mellinger, D., Powers, C., and Kumar, V. (2013). Towards a swarm of agile micro quadrotors. *Auton. Robots* 35, 287–300. doi: 10.1007/s10514-013-9349-9
- Lajoie, P.-Y., Ramtoula, B., Chang, Y., Carbone, L., and Beltrame, G. (2019). Door-slam: distributed, online, and outlier resilient slam for robotic teams. *arXiv [preprint]*. arXiv:1909.12198v2. doi: 10.1109/LRA.2020.2967681
- Laković, N., Brkić, M., Batinić, B., Bajić, J., Rajs, V., and Kulundžić, N. (2019). “Application of low-cost VL53L0X ToF sensor for robot environment detection,” in *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)* (East Sarajevo), 1–4. doi: 10.1109/INFOTEH.2019.8717779
- Lamers, K., Tijmons, S., De Wagter, C., and de Croon, G. (2016). “Self-supervised monocular distance learning on a lightweight micro air vehicle,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Daejeon), 1779–1784. doi: 10.1109/IROS.2016.7759284
- Lammering, T., Anton, E., and Henke, R. (2012). “Technology assessment on aircraft-level: modeling of innovative aircraft systems in conceptual aircraft design,” in *10th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)* (Fort Worth, TX). doi: 10.2514/6.2010-9072
- Leahy, K., Zhou, D., Vasile, C.-I., Oikonomopoulos, K., Schwager, M., and Belta, C. (2016). Persistent surveillance for unmanned aerial vehicles subject to charging and temporal logic constraints. *Autonom. Robots* 40, 1363–1378. doi: 10.1007/s10514-015-9519-z
- Lecoeur, J., Dacke, M., Floreano, D., and Baird, E. (2019). The role of optic flow pooling in insect flight control in cluttered environments. *Sci. Rep.* 9:7707. doi: 10.1038/s41598-019-44187-2
- Ledergerber, A., Hamer, M., and D’Andrea, R. (2015). “A robot self-localization system using one-way ultra-wideband communication,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Hamburg), 3131–3137. doi: 10.1109/IROS.2015.7353810
- Lee, D., Zhou, J., and Lin, W. T. (2015). “Autonomous battery swapping system for quadcopter,” in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)* (Denver, CO). doi: 10.1109/ICUAS.2015.7152282
- Lee, J., Su, Y., and Shen, C. (2007). “A comparative study of wireless protocols: Bluetooth, uwb, zigbee, and wi-fi,” in *IECON 2007–33rd Annual Conference of the IEEE Industrial Electronics Society* (Taipei), 46–51. doi: 10.1109/IECON.2007.4460126
- Lehman, J., and Stanley, K. O. (2011). Abandoning objectives: evolution through the search for novelty alone. *Evol. Comput.* 19, 189–223. doi: 10.1162/EVCO\_a\_00025
- Lehnert, C., and Corke, P. (2013). “ $\mu$ av–design and implementation of an open source micro quadrotor,” in *Australasian Conference on Robotics and Automation (ACRA2013)*, eds J. Katupitiya, J. Guivant, and R. Eaton (Sydney, NSW: University of New South Wales; Australian Robotics & Automation Association), 1–8.
- Leishman, R. C., Macdonald, J. C., Beard, R. W., and McLain, T. W. (2014). Quadrotors and accelerometers: State estimation with an improved dynamic model. *IEEE Control Syst. Mag.* 34, 28–41. doi: 10.1109/MCS.2013.2287362
- Leonard, J., Savvaris, A., and Tsourdos, A. (2014). Energy management in swarm of unmanned aerial vehicles. *J. Intel. Robot. Sys.* 74, 233–250. doi: 10.1007/s10846-013-9893-8
- Lew, T., Emmei, T., Fan, D. D., Bartlett, T., Santamaria-Navarro, A., Thakker, R., et al. (2019). “Contact inertial odometry: collisions are your friend,” in *International Symposium on Robotics Research (ISRR)* (Hanoi).
- Li, X., and Parker, L. E. (2007). “Sensor analysis for fault detection in tightly-coupled multi-robot team tasks,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation* (Roma), 3269–3276. doi: 10.1109/ROBOT.2007.363977
- Li, X., and Parker, L. E. (2009). “Distributed sensor analysis for fault detection in tightly-coupled multi-robot team tasks,” in *2009 IEEE International Conference on Robotics and Automation* (Kobe), 3103–3110. doi: 10.1109/ROBOT.2009.5152389
- Lindsey, Q., Mellinger, D., and Kumar, V. (2012). Construction with quadrotor teams. *Autonom. Robots* 33, 323–336. doi: 10.1007/s10514-012-9305-0
- Liu, W., Loiano, G., Mohta, K., Daniilidis, K., and Kumar, V. (2018). “Semi-dense visual-inertial odometry and mapping for quadrotors with swap constraints,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (Brisbane, QLD), 3904–3909. doi: 10.1109/ICRA.2018.8463163
- López, E., Barea, R., Gómez, A., Saltos, Á., Bergasa, L. M., Molinos, E. J., et al. (2016). “Indoor slam for micro aerial vehicles using visual and laser sensor fusion,” in *Robot 2015: Second Iberian Robotics Conference*, eds L. P. Reis, A. P. Moreira, P. U. Lima, L. Montano, and V. Muñoz-Martinez (Cham: Springer International Publishing), 531–542.
- Macdonald, J., Leishman, R., Beard, R., and McLain, T. (2014). Analysis of an improved IMU-based observer for multirotor helicopters. *J. Intell. Robot. Syst.* 74, 1049–1061. doi: 10.1007/s10846-013-9835-5
- Mademlis, I., Nikolaidis, N., Tefas, A., Pitas, I., Wagner, T., and Messina, A. (2019). Autonomous uav cinematography: a tutorial and a formalized shot-type taxonomy. *ACM Comput. Surv.* 52, 105:1–105:33. doi: 10.1145/3347713
- Mahony, R., Kumar, V., and Corke, P. (2012). Multirotor aerial vehicles: modeling, estimation, and control of quadrotor. *IEEE Robot. Autom. Mag.* 19, 20–32. doi: 10.1109/MRA.2012.2206474
- Mairaj, A., Baba, A. I., and Javaid, A. Y. (2019). Application specific drone simulators: recent advances and challenges. *Simul. Modell. Pract. Theory* 94, 100–117. doi: 10.1016/j.simpat.2019.01.004
- Martinelli, A. (2011). “Closed-form solution for attitude and speed determination by fusing monocular vision and inertial sensor measurements,” in *2011 IEEE International Conference on Robotics and Automation (ICRA)* (Shanghai), 4538–4545. doi: 10.1109/ICRA.2011.5980081
- Martinoli, A., and Easton, K. (2003). “Modeling swarm robotic systems,” in *Experimental Robotics VIII*, eds B. Siciliano and P. Dario (Berlin; Heidelberg: Springer Berlin Heidelberg), 297–306.
- Matthies, L., Brockers, R., Kuwata, Y., and Weiss, S. (2014). “Stereo vision-based obstacle avoidance for micro air vehicles using disparity space,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (Hong Kong), 3242–3249. doi: 10.1109/ICRA.2014.6907325
- McCallum, R. (1997). *Reinforcement learning with selective perception and hidden state* (Ph.D. thesis), University of Rochester, Rochester, NY, United States.
- McGuire, K., de Croon, G., Wagter, C. D., Tuyls, K., and Kappen, H. (2017). Efficient optical flow and stereo vision for velocity estimation and obstacle avoidance on an autonomous pocket drone. *IEEE Robot. Autom. Lett.* 2, 1070–1076. doi: 10.1109/LRA.2017.2658940
- McGuire, K. N., De Wagter, C., Tuyls, K., Kappen, H. J., and de Croon, G. C. H. E. (2019). Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment. *Sci. Robot.* 4:eaa9710. doi: 10.1126/scirobotics.aaw9710
- Meier, L., Honegger, D., and Pollefeys, M. (2015). “PX4: a node-based multithreaded open source robotics framework for deeply embedded platforms,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (Seattle, WA), 6235–6240. doi: 10.1109/ICRA.2015.7140074
- Melhuish, C., and Welsby, J. (2002). “Gradient ascent with a group of minimalist real robots: implementing secondary swarming,” in *IEEE International Conference on Systems, Man and Cybernetics*, Vol. 2 (Yasmine Hammamet), 509–514. doi: 10.1109/ICSMC.2002.1173464
- Meyer, J., Sendobry, A., Kohlbrecher, S., Klingauf, U., and von Stryk, O. (2012). “Comprehensive simulation of quadrotor UAVs using ROS and Gazebo,” in *Simulation, Modeling, and Programming for Autonomous Robots*, eds I. Noda, N. Ando, D. Brugalí, and J. J. Kuffner (Berlin; Heidelberg: Springer Berlin Heidelberg), 400–411.
- Michael, N., Mellinger, D., Lindsey, Q., and Kumar, V. (2010). The GRASP multiple micro-UAV testbed. *IEEE Robot. Autom. Mag.* 17, 56–65. doi: 10.1109/MRA.2010.937855
- Michelson, R. C., and Reece, S. (1998). “Update on flapping wing micro air vehicle research—ongoing work to develop a flapping wing, crawling entomopter,” in *13th Bristol International RPV/UAV Systems Conference Proceedings*, Vol. 30 (Bristol), 30–31.
- Millan-Romera, J. A., Perez-Leon, H., Castillejo-Calle, A., Maza, I., and Ollero, A. (2019). “ROS-MAGNA, a ROS-based framework for the definition and management of multi-UAS cooperative missions,” in *International Conference on Unmanned Aircraft Systems (ICUAS)* (Atlanta, GA), 1477–1486. doi: 10.1109/ICUAS.2019.8797829
- Mintchev, S., de Rivaz, S., and Floreano, D. (2017). Insect-inspired mechanical resilience for multicopters. *IEEE Robot. Autom. Lett.* 2, 1248–1255. doi: 10.1109/LRA.2017.2658946

- Mirjan, A., Augugliaro, F., D'Andrea, R., Gramazio, F., and Kohler, M. (2016). *Building a Bridge with Flying Robots*. Cham: Springer International Publishing, 34–47.
- Mohr, B. B., and Fitzpatrick, D. L. (2008). Micro air vehicle navigation system. *IEEE Aerosp. Electron. Syst. Mag.* 23, 19–24. doi: 10.1109/MAES.2008.4493438
- Montijano, E., Cristofalo, E., Zhou, D., Schwager, M., and Sagüés, C. (2016). Vision-based distributed formation control without an external positioning system. *IEEE Trans. Robot.* 32, 339–351. doi: 10.1109/TRO.2016.2523542
- Moore, R. J. D., Dantu, K., Barrows, G. L., and Nagpal, R. (2014). “Autonomous MAV guidance with a lightweight omnidirectional vision sensor,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (Hong Kong), 3856–3861. doi: 10.1109/ICRA.2014.6907418
- Mueggler, E., Baumli, N., Fontana, F., and Scaramuzza, D. (2015). “Towards evasive maneuvers with quadrotors using dynamic vision sensors,” in *2015 European Conference on Mobile Robots (ECMR)* (Lincoln, UK), 1–8. doi: 10.1109/ECMR.2015.7324048
- Mueller, M., and Drouin, A. (2007). “Paparazzi—the free autopilot. Build your own UAV,” in *24th Chaos Communication Congress* (Berlin), 27–30.
- Mulgaonkar, Y., Cross, G., and Kumar, V. (2015). “Design of small, safe and robust quadrotor swarms,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (Seattle, WA), 2208–2215. doi: 10.1109/ICRA.2015.7139491
- Mulgaonkar, Y., and Kumar, V. (2014). “Autonomous charging to enable long-endurance missions for small aerial robots,” in *Micro- and Nanotechnology Sensors, Systems, and Applications VI*, Vol. 9083, eds T. George, M. S. Islam, and A. K. Dutta (Baltimore, MD: International Society for Optics and Photonics, SPIE), 404–418.
- Mulgaonkar, Y., Makineni, A., Guerrero-Bonilla, L., and Kumar, V. (2018). Robust aerial robot swarms without collision avoidance. *IEEE Robot. Autom. Lett.* 3, 596–603. doi: 10.1109/LRA.2017.2775699
- Mulgaonkar, Y., Whitzer, M., Morgan, B., Kroninger, C. M., Harrington, A. M., and Kumar, V. (2014). “Power and weight considerations in small, agile quadrotors,” in *Micro- and Nanotechnology Sensors, Systems, and Applications VI*, Vol. 9083, eds T. George, M. S. Islam, and A. K. Dutta (Baltimore, MD: International Society for Optics and Photonics, SPIE), 376–391. doi: 10.1117/12.2051112
- Nägeli, T., Conte, C., Domahidi, A., Morari, M., and Hilliges, O. (2014). “Environment-independent formation flight for micro aerial vehicles,” in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Chicago, IL), 1141–1146. doi: 10.1109/IROS.2014.6942701
- Nagi, J., Giusti, A., Gambardella, L. M., and Di Caro, G. A. (2014). “Human-swarm interaction using spatial gestures,” in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Chicago, IL), 3834–3841. doi: 10.1109/IROS.2014.6943101
- Nascimento, T. P., and Saska, M. (2019). Position and attitude control of multi-rotor aerial vehicles: a survey. *Annu. Rev. Control* 48, 129–146. doi: 10.1016/j.arcontrol.2019.08.004
- Nedjah, N., and Junior, L. S. (2019). Review of methodologies and tasks in swarm robotics towards standardization. *Swarm Evol. Comput.* 50:100565. doi: 10.1016/j.swevo.2019.100565
- Nelson, A. L., Barlow, G. J., and Doitsidis, L. (2009). Fitness functions in evolutionary robotics: a survey and analysis. *Robot. Autom. Syst.* 57, 345–370. doi: 10.1016/j.robot.2008.09.009
- Newcombe, R. A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., et al. (2011). “Kinectfusion: real-time dense surface mapping and tracking,” in *2011 10th IEEE International Symposium on Mixed and Augmented Reality* (Basel), 127–136. doi: 10.1109/ISMAR.2011.6092378
- Nex, F., and Remondino, F. (2014). Uav for 3D mapping applications: a review. *Appl. Geomat.* 6, 1–15. doi: 10.1007/s12518-013-0120-x
- Nguyen, T., Qiu, Z., Nguyen, T. H., Cao, M., and Xie, L. (2019). Distance-based cooperative relative localization for leader-following control of MAVs. *IEEE Robot. Autom. Lett.* 4, 3641–3648. doi: 10.1109/LRA.2019.2926671
- Nieuwenhuisen, M., Beul, M., Rosu, R. A., Quenzel, J., Pavlichenko, D., Houben, S., et al. (2017). “Collaborative object picking and delivery with a team of micro aerial vehicles at MBZIRC,” in *2017 European Conference on Mobile Robots (ECMR)* (Paris), 1–6. doi: 10.1109/ECMR.2017.8098676
- Nolfi, S. (2002). Power and the limits of reactive agents. *Neurocomputing* 42, 119–145. doi: 10.1016/S0925-2312(01)00598-7
- Noth, A., and Siegwart, R. (2010). *Solar-Powered Micro-Air Vehicles and Challenges in Downscaling*. Berlin; Heidelberg: Springer Berlin Heidelberg, 285–297.
- Obert, E. (2009). *Aerodynamic Design of Transport Aircraft*. Amsterdam: IOS Press.
- Odelga, M., Stegagno, P., and Bühlhoff, H. H. (2016). “Obstacle detection, tracking and avoidance for a teleoperated UAV,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)* (Stockholm), 2984–2990. doi: 10.1109/ICRA.2016.7487464
- Oh, K.-K., Park, M.-C., and Ahn, H.-S. (2015). A survey of multi-agent formation control. *Automatica* 53, 424–440. doi: 10.1016/j.automatica.2014.10.022
- Olejnik, D., Duisterhof, B., Karásek, M., Scheper, K. Y. W., van Dijk, T., and de Croon, G. C. H. E. (2019). “A tailless flapping wing mav performing monocular visual servoing tasks,” in *International Micro Air Vehicle Conference and Competition (IMAV)* (Madrid).
- Oleynikova, H., Honegger, D., and Pollefeys, M. (2015). “Reactive avoidance using embedded stereo vision for MAV flight,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (Seattle, WA), 50–56. doi: 10.1109/ICRA.2015.7138979
- Opromolla, R., Fasano, G., Rufino, G., Grassi, M., and Savvaris, A. (2016). “Lidar-inertial integration for UAV localization and mapping in complex environments,” in *2016 International Conference on Unmanned Aircraft Systems (ICUAS)* (Arlington, VA), 649–656. doi: 10.1109/ICUAS.2016.7502580
- Opromolla, R., Inchingolo, G., and Fasano, G. (2019). Airborne visual detection and tracking of cooperative UAVs exploiting deep learning. *Sensors* 19:4332. doi: 10.3390/s19194332
- Oung, R., and D'Andrea, R. (2011). The distributed flight array. *Mechatronics* 21, 908–917. doi: 10.1016/j.mechatronics.2010.08.003
- Palunko, I., Fierro, R., and Cruz, P. (2012). “Trajectory generation for swing-free maneuvers of a quadrotor with suspended payload: a dynamic programming approach,” in *2012 IEEE International Conference on Robotics and Automation* (Saint Paul, MN), 2691–2697. doi: 10.1109/ICRA.2012.6225213
- Pan, Z., An, L., and Wen, C. (2019). Recent advances in fuel cells based propulsion systems for unmanned aerial vehicles. *Appl. Energy* 240, 473–485. doi: 10.1016/j.apenergy.2019.02.079
- Parsons, S., and Wooldridge, M. (2002). Game theory and decision theory in multi-agent systems. *Autonom. Agents Multi Agent Syst.* 5, 243–254. doi: 10.1007/978-1-4615-1107-6
- Pascoal, A., Kaminer, I., and Oliveira, P. (2000). Navigation system design using time-varying complementary filters. *IEEE Trans. Aerosp. Electron. Syst.* 36, 1099–1114. doi: 10.1109/7.892661
- Pestana, J., Sanchez-Lopez, J. L., de la Puente, P., Carrio, A., and Campoy, P. (2014). “A vision-based quadrotor swarm for the participation in the 2013 international micro air vehicle competition,” in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)* (Orlando, FL), 617–622. doi: 10.1109/ICUAS.2014.6842305
- Petricca, L., Ohlckers, P., and Grinde, C. (2011). Micro-and nano-air vehicles: state of the art. *Int. J. Aerosp. Eng.* 2011:214549. doi: 10.1155/2011/214549
- Pincirolì, C., and Beltrame, G. (2016). “Buzz: an extensible programming language for heterogeneous swarm robotics,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Daejeon), 3794–3800. doi: 10.1109/IROS.2016.7759558
- Pincirolì, C., Trianni, V., O'Grady, R., Pini, G., Brutschy, A., Brambilla, M., et al. (2012). Argos: a modular, parallel, multi-engine simulator for multi-robot systems. *Swarm Intell.* 6, 271–295. doi: 10.1007/s11721-012-0072-5
- Pitonakova, L., Crowder, R., and Bullock, S. (2018). Information exchange design patterns for robot swarm foraging and their application in robot control algorithms. *Front. Robot. AI* 5:47. doi: 10.3389/frobot.2018.00047
- Portelli, G., Ruffier, F., Roubieu, F. L., and Franceschini, N. (2011). Honeybees' speed depends on dorsal as well as lateral, ventral and frontal optic flows. *PLoS ONE* 6:e19486. doi: 10.1371/journal.pone.0019486
- Powers, C., Mellinger, D., Kushleyev, A., Kothmann, B., and Kumar, V. (2013). “Influence of aerodynamics and proximity effects in quadrotor flight,” in *Experimental Robotics: The 13th International Symposium on Experimental Robotics*, eds J. P. Desai, G. Dudek, O. Khatib, and V. Kumar (Heidelberg: Springer International Publishing), 289–302.
- Preiss, J. A., Honig, W., Sukhatme, G. S., and Ayanian, N. (2017). “Crazyswarm: A large nano-quadcopter swarm,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)* (Singapore), 3299–3304. doi: 10.1109/ICRA.2017.7989376

- Qin, L., He, X., and Zhou, D. (2014). A survey of fault diagnosis for swarm systems. *Syst. Sci. Control Eng.* 2, 13–23. doi: 10.1080/21642583.2013.873745
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., et al. (2009). “ROS: an open-source robot operating system,” in *ICRA Workshop on Open Source Software*, Vol. 3 (Kobe), 5.
- Quintero, S. A. P., Collins, G. E., and Hespanha, J. P. (2013). “Flocking with fixed-wing UAVs for distributed sensing: a stochastic optimal control approach,” in *2013 American Control Conference* (Washington, DC), 2025–2031. doi: 10.1109/ACC.2013.6580133
- Rasmussen, J. (1983). Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE Trans. Syst. Man Cybernet.* 13, 257–266.
- Remes, B. D. W., Esden-Tempski, P., Van Tienen, F., Smeur, E., De Wagter, C., and De Croon, G. C. H. E. (2014). “Lisa-S 2.8 g autopilot for GPS-based flight of MAVs,” in *International Micro Air Vehicle Conference and Competition (IMAV)* (Delft).
- Reynolds, C. W. (1987). “Flocks, herds and schools: a distributed behavioral model,” in *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '87* (New York, NY: ACM), 25–34.
- Roberts, J. F., Stirling, T., Zufferey, J.-C., and Floreano, D. (2012). 3-D relative positioning sensor for indoor flying robots. *Autonom. Robots* 33, 5–20. doi: 10.1007/s10514-012-9277-0
- Roelofsen, S., Gillet, D., and Martinoli, A. (2015). “Reciprocal collision avoidance for quadrotors using on-board visual detection,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Hamburg), 4810–4817. doi: 10.1109/IROS.2015.7354053
- Ross, P. E. (2014). Open-source drones for fun and profit. *IEEE Spectr.* 51, 54–59. doi: 10.1109/MSPEC.2014.6745885
- Ross, S., Melik-Barkhudarov, N., Shankar, K. S., Wendel, A., Dey, D., Bagnell, J. A., et al. (2013). “Learning monocular reactive UAV control in cluttered natural environments,” in *2013 IEEE International Conference on Robotics and Automation* (Karlsruhe), 1765–1772. doi: 10.1109/ICRA.2013.6630809
- Ruiz-Espitia, O., Martínez-Carranza, J., and Rascon, C. (2018). “AIRA-UAS: an evaluation corpus for audio processing in unmanned aerial system,” in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)* (Dallas, TX), 836–845. doi: 10.1109/ICUAS.2018.8453466
- Sa, I., and Corke, P. (2014). “Vertical infrastructure inspection using a quadcopter and shared autonomy control,” in *Field and Service Robotics: Results of the 8th International Conference*, eds K. Yoshida and S. Tadokoro (Berlin; Heidelberg: Springer Berlin Heidelberg), 219–232.
- Sabatini, A. M., and Genovese, V. (2013). A stochastic approach to noise modeling for barometric altimeters. *Sensors* 13, 15692–15707. doi: 10.3390/s131115692
- Saha, S., Natraj, A., and Waharte, S. (2014). “A real-time monocular vision-based frontal obstacle detection and avoidance for low cost UAVs in GPS denied environment,” in *2014 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology* (Yogyakarta), 189–195. doi: 10.1109/ICARES.2014.7024382
- Şahin, E. (2005). “Swarm robotics: from sources of inspiration to domains of application,” in *Swarm Robotics*, eds E. Şahin and W. M. Spears (Berlin; Heidelberg: Springer), 10–20.
- Şahin, E., Girgin, S., Bayindir, L., and Turgut, A. E. (2008). *Swarm Robotics*. Berlin; Heidelberg: Springer Berlin Heidelberg, 87–100.
- Saldaña, D., Gabrich, B., Li, G., Yim, M., and Kumar, V. (2018). “ModQuad: the flying modular structure that self-assembles in midair,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (Brisbane, QLD), 691–698. doi: 10.1109/ICRA.2018.8461014
- Saldaña, D., Gupta, P. M., and Kumar, V. (2019). Design and control of aerial modules for inflight self-disassembly. *IEEE Robot. Autom. Lett.* 4, 3410–3417. doi: 10.1109/LRA.2019.2926680
- Saldaña, D., Prorok, A., Sundaram, S., Campos, M. F. M., and Kumar, V. (2017). “Resilient consensus for time-varying networks of dynamic agents,” in *2017 American Control Conference (ACC)* (Seattle, WA), 252–258. doi: 10.23919/ACC.2017.7962962
- Sanchez-Lopez, J. L., Fernández, R. A. S., Bavle, H., Sampedro, C., Molina, M., Pestana, J., et al. (2016). “AEROSTACK: an architecture and open-source software framework for aerial robotics,” in *2016 International Conference on Unmanned Aircraft Systems (ICUAS)* (Arlington, VA), 332–341. doi: 10.1109/ICUAS.2016.7502591
- Santamaria-Navarro, A., Solà, J., and Andrade-Cetto, J. (2015). “High-frequency MAV state estimation using low-cost inertial and optical flow measurement units,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Hamburg), 1864–1871. doi: 10.1109/IROS.2015.7353621
- Saska, M. (2015). “MAV-swarms: unmanned aerial vehicles stabilized along a given path using onboard relative localization,” in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)* (Denver, CO), 894–903. doi: 10.1109/ICUAS.2015.7152376
- Saska, M., Baca, T., Spurný, V., Loianno, G., Thomas, J., Krajník, T., et al. (2016a). “Vision-based high-speed autonomous landing and cooperative objects grasping-towards the MBZIRC competition,” in *Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems-Vision-based High Speed Autonomous Navigation of UAVs (Workshop)* (Daejeon), 9–14.
- Saska, M., Baca, T., Thomas, J., Chudoba, J., Preucil, L., Krajník, T., et al. (2017). System for deployment of groups of unmanned micro aerial vehicles in gps-denied environments using onboard visual relative localization. *Autonom. Robots* 41, 919–944. doi: 10.1007/s10514-016-9567-z
- Saska, M., Vakula, J., and Přeucil, L. (2014). “Swarms of micro aerial vehicles stabilized under a visual relative localization,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (Hong Kong), 3570–3575. doi: 10.1109/ICRA.2014.6907374
- Saska, M., Vonásek, V., Chudoba, J., Thomas, J., Loianno, G., and Kumar, V. (2016b). Swarm distribution and deployment for cooperative surveillance by micro-aerial vehicles. *J. Intell. Robot. Syst.* 84, 469–492. doi: 10.1007/s10846-016-0338-z
- Saulnier, K., Saldaña, D., Prorok, A., Pappas, G. J., and Kumar, V. (2017). Resilient flocking for mobile robot teams. *IEEE Robot. Autom. Lett.* 2, 1039–1046. doi: 10.1109/LRA.2017.2655142
- Scaramuzza, D., Achtelik, M. C., Doitsidis, L., Friedrich, F., Kosmatopoulos, E., Martinelli, A., et al. (2014). Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in GPS-denied environments. *IEEE Robot. Autom. Mag.* 21, 26–40. doi: 10.1109/MRA.2014.2322295
- Scaramuzza, D., and Zhang, Z. (2019). Visual-inertial odometry of aerial robots. *arXiv [preprint]*. arXiv:1906.03289.
- Schauwecker, K., Ke, N. R., Scherer, S. A., and Zell, A. (2012). “Markerless visual control of a quad-rotor micro aerial vehicle by means of on-board stereo processing,” in *Autonomous Mobile Systems 2012*, eds P. Levi, O. Zweigle, K. Häußermann, nd B. Eckstein (Berlin; Heidelberg: Springer Berlin Heidelberg), 11–20.
- Schauwecker, K., and Zell, A. (2014). On-board dual-stereo-vision for the navigation of an autonomous MAV. *J. Intell. Robot. Syst.* 74, 1–16. doi: 10.1007/s10846-013-9907-6
- Scheper, K. (2019). *Abstraction as a tool to bridge the reality gap in evolutionary robotics* (Ph.D. thesis), Delft University of Technology, Delft, Netherlands.
- Scheper, K. Y. W., and de Croon, G. C. H. E. (2017). Abstraction, sensory-motor coordination, and the reality gap in evolutionary robotics. *Artif. Life* 23, 124–141. doi: 10.1162/ARTL\_a\_00227
- Scheper, K. Y. W., Tijmons, S., de Visser, C. C., and de Croon, G. C. H. E. (2016). Behavior trees for evolutionary robotics. *Artif. Life* 22, 23–48. doi: 10.1162/ARTL\_a\_00192
- Schiano, F., Franchi, A., Zelazo, D., and Giordano, P. R. (2016). “A rigidity-based decentralized bearing formation controller for groups of quadrotor UAVs,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Daejeon), 5099–5106.
- Schilling, F., Lecoœur, J., Schiano, F., and Floreano, D. (2019). Learning vision-based flight in drone swarms by imitation. *IEEE Robot. Autom. Lett.* 4, 4523–4530. doi: 10.1109/LRA.2019.2935377
- Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). “Grad-cam: visual explanations from deep networks via gradient-based localization,” in *2017 IEEE International Conference on Computer Vision (ICCV)* (Venice), 618–626. doi: 10.1109/ICCV.2017.74
- Shah, S., Dey, D., Lovett, C., and Kapoor, A. (2018). “AirSim: high-fidelity visual and physical simulation for autonomous vehicles,” in *Field and Service Robotics*, eds M. Hutter and R. Siegwart (Cham: Springer International Publishing), 621–635.



- Shen, S., Michael, N., and Kumar, V. (2011). "Autonomous multi-floor indoor navigation with a computationally constrained MAV," in *2011 IEEE International Conference on Robotics and Automation* (Shanghai), 20–25. doi: 10.1109/ICRA.2011.5980357
- Shen, S., Mulgaonkar, Y., Michael, N., and Kumar, V. (2014). "Multi-sensor fusion for robust autonomous flight in indoor and outdoor environments with a rotorcraft MAV," in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (Hong Kong), 4974–4981. doi: 10.1109/ICRA.2014.6907588
- Shi, J., He, B., Zhang, L., and Zhang, J. (2016). "Vision-based real-time 3D mapping for UAV with laser sensor," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Daejeon), 4524–4529. doi: 10.1109/IROS.2016.7759666
- Shilov, K. (2014). "The next generation design of autonomous mav flight control system SMARTAP," in *International Micro Air Vehicle Conference and Competition (IMAV)* (Delft).
- Silva, F., Duarte, M., Correia, L., Oliveira, S. M., and Christensen, A. L. (2016). Open issues in evolutionary robotics. *Evol. Comput.* 24, 205–236. doi: 10.1162/EVCO\_a\_00172
- Silva, F., Urbano, P., Correia, L., and Christensen, A. L. (2015). odNEAT: an algorithm for decentralised online evolution of robotic controllers. *Evol. Comput.* 23, 421–449. doi: 10.1162/EVCO\_a\_00141
- Smisek, J., Jancosek, M., and Pajdla, T. (2013). *3D With Kinect*. London: Springer London, 3–25.
- Snape, J., van den Berg, J., Guy, S. J., and Manocha, D. (2009). "Independent navigation of multiple mobile robots with hybrid reciprocal velocity obstacles," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* (St. Louis, MO), 5917–5922. doi: 10.1109/IROS.2009.5354821
- Snape, J., van den Berg, J., Guy, S. J., and Manocha, D. (2011). The hybrid reciprocal velocity obstacle. *IEEE Trans. Robot.* 27, 696–706. doi: 10.1109/TRO.2011.2120810
- Soto-Gerrero, D., and Ramirez-Torres, J.-G. (2016). "A human-machine interface with unmanned aerial vehicles," in *Robotics and Mechatronics*, eds S. Zeghloul, M. A. Laribi, and J. P. Gazeau (Cham: Springer International Publishing), 233–242.
- Spurný, V., Bába, T., Saska, M., Pěnička, R., Krajník, T., Thomas, J., et al. (2019). Cooperative autonomous search, grasping, and delivering in a treasure hunt scenario by a team of unmanned aerial vehicles. *J. Field Robot.* 36, 125–148. doi: 10.1002/rob.21816
- Stancliff, S. B., Dolan, J. M., and Trebi-Ollennu, A. (2006). "Mission reliability estimation for multirobot team design," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Beijing), 2206–2211. doi: 10.1109/IROS.2006.282562
- Stegagno, P., Basile, M., Bühlhoff, H. H., and Franchi, A. (2014). "A semi-autonomous uav platform for indoor remote operation with visual and haptic feedback," in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (Hong Kong), 3862–3869. doi: 10.1109/ICRA.2014.6907419
- Stegagno, P., Cognetti, M., Franchi, A., and Oriolo, G. (2011). "Mutual localization using anonymous bearing measurements," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems* (San Francisco, CA), 469–474. doi: 10.1109/IROS.2011.6094635
- Suleiman, A., Zhang, Z., Carlone, L., Karaman, S., and Sze, V. (2018). "Navion: a fully integrated energy-efficient visual-inertial odometry accelerator for autonomous navigation of nano drones," in *2018 IEEE Symposium on VLSI Circuits* (Honolulu, HI), 133–134. doi: 10.1109/VLSIC.2018.8502279
- Sutton, R. S., and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. 2nd Edn. Cambridge, MA: MIT Press.
- Swatman, R. (2016a). *Intel Stuns During CES Keynote With Record for Most Drones Airborne Simultaneously—Watch Incredible Footage*. Guinness World Records. Available online at: <https://www.guinnessworldrecords.com/news/brand-or-agency/2016/1/intel-stuns-during-ces-keynote-with-record-for-most-drones-airborne-simultaneous-411677>
- Swatman, R. (2016b). *Intel Launches 500 Drones Into Sky and Breaks World Record in Spectacular Style*. Guinness World Records. Available online at: <https://www.guinnessworldrecords.com/news/2016/11/intel-launches-500-drones-into-sky-and-breaks-world-record-in-spectacular-style-449886>
- Szabo, T. (2015). *Autonomous collision avoidance for swarms of MAVs: based solely on RSSI measurements* (Master's thesis), Delft University of Technology, Delft, Netherlands.
- Tagliabue, A., Kamel, M., Siegart, R., and Nieto, J. (2019). Robust collaborative object transportation using multiple MAVs. *Int. J. Robot. Res.* 38, 1020–1044. doi: 10.1177/0278364919854131
- Tarapore, D., Christensen, A. L., Lima, P. U., and Carneiro, J. (2013). "Abnormality detection in multiagent systems inspired by the adaptive immune system," in *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems, AAMAS '13* (Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems), 23–30.
- Tarapore, D., Christensen, A. L., and Timmis, J. (2015a). "Abnormality detection in robots exhibiting composite swarm behaviours," in *The 2019 Conference on Artificial Life*, Vol. 27 (York), 406–413.
- Tarapore, D., Christensen, A. L., and Timmis, J. (2017). Generic, scalable and decentralized fault detection for robot swarms. *PLoS ONE* 12:e182058. doi: 10.1371/journal.pone.0182058
- Tarapore, D., Lima, P. U., Carneiro, J., and Christensen, A. L. (2015b). To err is robotic, to tolerate immunological: fault detection in multirobot systems. *Bioinspir. Biomim.* 10:016014. doi: 10.1088/1748-3190/10/1/016014
- Tarapore, D., Timmis, J., and Christensen, A. L. (2019). Fault detection in a swarm of physical robots based on behavioral outlier detection. *IEEE Trans. Robot.* 35, 1516–1522. doi: 10.1109/TRO.2019.2929015
- Teixeira, L., Maffra, F., Moos, M., and Chli, M. (2018). Vi-rpe: visual-inertial relative pose estimation for aerial vehicles. *IEEE Robot. Autom. Lett.* 3, 2770–2777. doi: 10.1109/LRA.2018.2837687
- Thurrowgood, S., Soccol, D., Moore, R. J. D., Bland, D., and Srinivasan, M. V. (2009). "A vision based system for attitude estimation of UAVs," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* (St. Louis, MO), 5725–5730. doi: 10.1109/IROS.2009.5354041
- Tijs, E., de Croon, G. C. H. E., Wind, J., Remes, B., De Wagter, C., de Bree, H. E., et al. (2010). "Hear-and-avoid for micro air vehicles," in *International Micro Air Vehicle Conference and Competition (IMAV)* (Braunschweig), Vol. 69.
- Toksoz, T., Redding, J., Michini, M., Michini, B., How, J. P., Vavrana, M., et al. (2011). "Automated battery swap and recharge to enable persistent UAV missions," in *Infotech@Aerospace 2011*, Vol. 21 (St. Louis, MO). doi: 10.2514/6.2011-1405
- Trianni, V. (2008). *Evolutionary Swarm Robotics: Evolving Self-organising Behaviours in Groups of Autonomous Robots*, Vol. 108. Berlin; Heidelberg: Springer. doi: 10.1007/978-3-540-77612-3
- Trianni, V. (2014). Evolutionary robotics: model or design? *Front. Robot. AI* 1:13. doi: 10.3389/frobt.2014.00013
- Trianni, V., and Campo, A. (2015). *Fundamental Collective Behaviors in Swarm Robotics*. Berlin; Heidelberg: Springer Berlin Heidelberg, 1377–1394.
- Trianni, V., Grof, R., Labella, T. H., Şahin, E., and Dorigo, M. (2003). "Evolving aggregation behaviors in a swarm of robots," in *Advances in Artificial Life*, eds W. Banzhaf, J. Ziegler, T. Christaller, P. Dittrich, and J. T. Kim (Berlin; Heidelberg: Springer Berlin Heidelberg), 865–874.
- Tripicchio, P., Satler, M., Unetti, M., and Avizzano, C. A. (2018). Confined spaces industrial inspection with micro aerial vehicles and laser range finder localization. *Int. J. Micro Air Vehicles* 10, 207–224. doi: 10.1177/1756829318757471
- Troub, B., DePineuil, B., and Montalvo, C. (2017). Simulation analysis of a collision-tolerant micro-airship fleet. *Int. J. Micro Air Vehicles* 9, 297–305. doi: 10.1177/1756829317705326
- Tsykunov, E., Labazanov, L., Tleugazy, A., and Tsetserukou, D. (2018). "Swarmtouch: tactile interaction of human with impedance controlled swarm of nano-quadrotors," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid), 4204–4209. doi: 10.1109/IROS.2018.8594424
- Tsykunov, E., and Tsetserukou, D. (2019). "WiredSwarm: high resolution haptic feedback provided by a swarm of drones to the user's fingers for VR interaction," in *VRST '19: 25th ACM Symposium on Virtual Reality Software and Technology* (New York, NY), 102:1–102:2. doi: 10.1145/3359996.3364789
- Valentini, G. (2017). *Achieving Consensus in Robot Swarms: Design and Analysis of Strategies for the Best-of-n Problem*, Vol. 706. Cham: Springer International Publishing. doi: 10.1007/978-3-319-53609-5

- van den Berg, J., Guy, S. J., Lin, M., and Manocha, D. (2011). "Reciprocal n-body collision avoidance," in *Robotics Research*, eds C. Pradalier, R. Siegwart, and G. Hirzinger (Berlin; Heidelberg, Springer Berlin Heidelberg), 3–19.
- van den Berg, J., Lin, M., and Manocha, D. (2008). "Reciprocal velocity obstacles for real-time multi-agent navigation," in *2008 IEEE International Conference on Robotics and Automation* (Pasadena, CA), 1928–1935. doi: 10.1109/ROBOT.2008.4543489
- van der Helm, S., Coppola, M., McGuire, K. N., and de Croon, G. C. H. E. (2019). On-board range-based relative localization for micro air vehicles in indoor leader-follower flight. *Autonom. Robots*. doi: 10.1007/s10514-019-09843-6
- van Hecke, K., de Croon, G., van der Maaten, L., Hennes, D., and Izzo, D. (2018). Persistent self-supervised learning: from stereo to monocular vision for obstacle avoidance. *Int. J. Micro Air Vehicles* 10, 186–206. doi: 10.1177/1756829318756355
- Vásárhelyi, G., Virágh, C., Somorjai, G., Nepusz, T., Eiben, A. E., and Vicsek, T. (2018). Optimized flocking of autonomous drones in confined environments. *Sci. Robot.* 3:aat3536. doi: 10.1126/scirobotics.aat3536
- Vasile, C. I., and Belta, C. (2014). "An automata-theoretic approach to the vehicle routing problem," in *Robotics: Science and Systems* (Berkeley, CA). doi: 10.15607/rss.2014.x.045
- Vedder, B., Eriksson, H., Skarin, D., Vinter, J., and Jonsson, M. (2015). "Towards collision avoidance for commodity hardware quadcopters with ultrasound localization," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)* (Denver, CO), 193–203. doi: 10.1109/ICUAS.2015.7152291
- Vempati, A. S., Khurana, H., Kabelka, V., Flueckiger, S., Siegwart, R., and Beardsley, P. (2019). A virtual reality interface for an autonomous spray painting UAV. *IEEE Robot. Autom. Lett.* 4, 2870–2877. doi: 10.1109/LRA.2019.2922588
- Verma, L., Fakharzadeh, M., and Choi, S. (2013). Wifi on steroids: 802.11ac and 802.11ad. *IEEE Wireless Commun.* 20, 30–35. doi: 10.1109/MWC.2013.6704471
- Vincenzi, D. A., Terwilliger, B. A., and Ison, D. C. (2015). Unmanned aerial system (UAS) human-machine interfaces: new paradigms in command and control. *Proc. Manuf.* 3, 920–927. doi: 10.1016/j.promfg.2015.07.139
- Voskuijl, M., van Bogaert, J., and Rao, A. G. (2018). Analysis and design of hybrid electric regional turboprop aircraft. *CEAS Aeronaut. J.* 9, 15–25. doi: 10.1007/s13272-017-0272-1
- Walter, V., Staub, N., Franchi, A., and Saska, M. (2019). UVDAR system for visual relative localization with application to leader-follower formations of multirotor UAVs. *IEEE Robot. Autom. Lett.* 4, 2637–2644. doi: 10.1109/LRA.2019.2901683
- Walter, V., Staub, N., Saska, M., and Franchi, A. (2018). "Mutual localization of UAVs based on blinking ultraviolet markers and 3D time-position hough transform," in *IEEE 14th International Conference on Automation Science and Engineering (CASE)* (Munich), 298–303. doi: 10.1109/COASE.2018.8560384
- Weimerskirch, H., Martin, J., Clerquin, Y., Alexandre, P., and Jiraskova, S. (2001). Energy saving in flight formation. *Nature* 413, 697–698. doi: 10.1038/35099670
- Weinstein, A., Cho, A., Loianno, G., and Kumar, V. (2018). Visual inertial odometry swarm: an autonomous swarm of vision-based quadrotors. *IEEE Robot. Autom. Lett.* 3, 1801–1807. doi: 10.1109/LRA.2018.2800119
- Weiss, S., Achtelik, M., Kneip, L., Scaramuzza, D., and Siegwart, R. (2011). Intuitive 3D maps for MAV terrain exploration and obstacle avoidance. *J. Intell. Robot. Syst.* 61, 473–493. doi: 10.1007/s10846-010-9491-y
- Winfield, A. F., and Nembrini, J. (2006). Safety in numbers: fault-tolerance in robot swarms. *Int. J. Modell. Identif. Control* 1:30. doi: 10.1504/IJMIC.2006.008645
- Winfield, A. F. T., Harper, C. F., and Nembrini, J. (2006). "Towards the application of swarm intelligence in safety critical systems," in *The First Institution of Engineering and Technology International Conference on System Safety* (London), 7.
- Winfield, A. F. T., Harper, C. J., and Nembrini, J. (2005). *Towards Dependable Swarms and a New Discipline of Swarm Engineering*. Berlin; Heidelberg: Springer Berlin Heidelberg, 126–142.
- Wood, R., Nagpal, R., and Wei, G.-Y. (2013). Flight of the robobees. *Sci. Am.* 308, 60–65. doi: 10.1038/scientificamerican0313-60
- Yang, G.-Z., Bellingham, J., Dupont, P. E., Fischer, P., Floridi, L., Full, R., et al. (2018). The grand challenges of science robotics. *Sci. Robot.* 3. doi: 10.1126/scirobotics.aar7650
- Yousif, K., Bab-Hadiashar, A., and Hoseinnezhad, R. (2015). An overview to visual odometry and visual SLAM: applications to mobile robotics. *Intell. Ind. Syst.* 1, 289–311. doi: 10.1007/s40903-015-0032-7
- Yuan, Q., Zhan, J., and Li, X. (2017). Outdoor flocking of quadcopter drones with decentralized model predictive control. *ISA Trans.* 71, 84–92. doi: 10.1016/j.isatra.2017.07.005
- Zou, D., Tan, P., and Yu, W. (2019). Collaborative visual slam for multiple agents: a brief survey. *Virtual Reality Intell. Hardware* 1, 461–482. doi: 10.1016/j.vrih.2019.09.002
- Zufferey, J.-C., Beyeler, A., and Floreano, D. (2010). Autonomous flight at low altitude using light sensors and little computational power. *Int. J. Micro Air Vehicles* 2, 107–117. doi: 10.1260/1756-8293.2.2.107
- Zufferey, J.-C., Hauert, S., Stirling, T., Leven, S., Roberts, J., and Floreano, D. (2013). "Aerial collective systems," in *Handbook of Collective Robotics* (New York, NY: Pan Stanford), 609–659. doi: 10.1201/b14908-21

**Conflict of Interest:** The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2020 Coppola, McGuire, De Wagter and de Croon. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.