

Reynolds flocking in reality with fixed-wing robots: communication range vs. maximum turning rate

Sabine Hauert^{*1}, Severin Leven^{*1}, Maja Varga¹, Fabio Ruini²
Angelo Cangelosi², Jean-Christophe Zufferey¹, Dario Floreano¹

Abstract—The success of swarm behaviors often depends on the range at which robots can communicate and the speed at which they change their behavior. Challenges arise when the communication range is too small with respect to the dynamics of the robot, preventing interactions from lasting long enough to achieve coherent swarming. To alleviate this dependency, most swarm experiments done in laboratory environments rely on communication hardware that is relatively long range and wheeled robotic platforms that have omnidirectional motion. Instead, we focus on deploying a swarm of small fixed-wing flying robots. Such platforms have limited payload, resulting in the use of short-range communication hardware. Furthermore, they are required to maintain forward motion to avoid stalling and typically adopt low turn rates because of physical or energy constraints. The tradeoff between communication range and flight dynamics is exhaustively studied in simulation in the scope of Reynolds flocking and demonstrated with up to 10 robots in outdoor experiments.

I. INTRODUCTION

Swarm behavior is often dependent on the ability of robots to interact. In most swarm experiments, interactions rely on communication hardware to broadcast messages to neighboring robots. However, interactions can be compromised if the robots are not able to communicate at a range that is suitable with respect to the speed at which they can react. As shown in Fig. 1, a robot that has a small communication range will need to aggressively alter its trajectory to prevent communication breaks. Likewise, a robot that is not able to maneuver quickly will need to communicate at longer ranges.

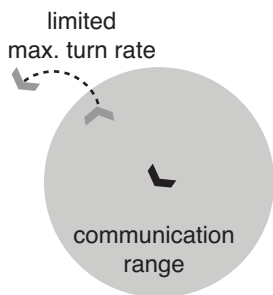


Fig. 1. Schematic illustrating the relationship between motion dynamics and communication range on swarm interactions. Here the robot shown in grey attempts to align heading with the black robot by turning as fast as possible. Because the robot has limited turning rate, communication is lost during the maneuver.

This problem is often alleviated in laboratory swarm experiments since communication ranges are typically large enough for office environments and robots are usually capable of omnidirectional motion, meaning they can rapidly change their direction, stop and turn on the spot [1]–[5]. In reality however, robots may be required to deploy over larger areas. Their communication range will be limited by energy, weight and cost constraints and motion dynamics will depend on physical and energy constraints of the platform [6]. There is therefore a need to study the impact of communication range and motion dynamics on the success of swarm behavior so that optimal design choices can be made.

In particular, we consider using fixed-wing flying robots that are subject to strict design choices in terms of communication hardware and flight dynamics because they are required to be lightweight, low-cost, safe and easy to use. These robots need to maintain forward motion to avoid stalling and can not make sharp turns. Furthermore, they are equipped with light-weight and low-cost WiFi dongles that intrinsically limit their communication range [7].

To study the tradeoff between communication range and motion dynamics on swarm behavior we consider a scenario where robots need to flock. Aerial flocking allows several robots to move as one without the burden of deploying a single large drone. As an advantage, the failure of any one robot does not cause the entire mission to fail (robustness) and the number of robots can be increased to extend the capability of the swarm (scalability). Flocking can be used to distribute communication, sensing or computational payloads across different robots [8] or to generate visually pleasing aerial patterns [9].

Rules driving bird-like flocking were proposed by Craig W. Reynolds [10] and can be summarized as follows:

- Alignment: robots align their velocity to the average velocity of neighboring robots.
- Cohesion: robots are attracted to the average position of neighboring robots.
- Separation: robots are repulsed from neighboring robots.

To avoid that robots fly away in real-world experiments, we also consider the well studied “migration” rule to attract the flock towards a predefined “migration point” in the environment [11]. Each rule contributes to the final velocity of the robot as shown in Fig. 2. Robots use communication to transmit their absolute position and heading to neighboring robots. In this paper, we only consider 2D flocking with robots flying at constant altitude. The emergent equilibrium

1. Laboratory of Intelligent Systems, Ecole Polytechnique Fédérale de Lausanne, Switzerland jean-christophe.zufferey@epfl.ch

2. Centre for Robotics and Neural Systems, University of Plymouth, UK

*These authors contributed equally to this work.

in the swarm allows the robots to remain at a constant distance from one another, thus avoiding collisions or large separations, while having the group advance in a common direction.

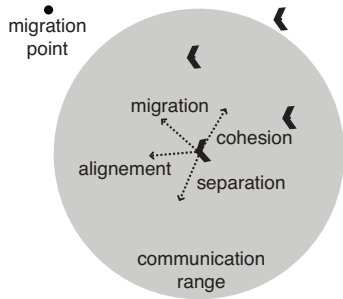


Fig. 2. Local alignment, cohesion, separation and migration velocity vectors acting on the motion of flocking robots.

Reynolds flocking has been extensively studied in the literature using swarms of self-propelled particles [1]–[5] or ground robots [12]–[14] that have omnidirectional motion. Recently, many researchers have considered using Reynolds flocking as a basis to deploy aerial swarms in simulation [15]–[17]. However, only few demonstrations of aerial flocking exist in reality with one noticeable example by Welsby et al. who deployed four blimps in an indoor environment [18]. Work with aerial robots typically assumes helicopter or blimp dynamics, which can be approximated as omnidirectional motion and little work has been done to consider challenges with real-world communication. The impact of non-holonomic motion on the convergence of flocking was considered in theoretical work by Tanner et al., although robots in this system were able to stop [19]. Furthermore, the influence of communication on the cohesion of flocks was studied in work by Şamiloğlu et al. [20]. Instead, we aim at considering both the impact of communication range and motion constraints on the success of aerial flocking in reality.

II. MATERIALS & METHODS

To explore the tradeoff between communication range and motion dynamics on aerial flocking, extensive research is done in simulation and demonstrated in reality using up to 10 fixed-wing robots.

A. OUTDOOR AERIAL SWARM SETUP

All the necessary software and hardware to perform fully autonomous outdoor experiments with aerial swarms was developed by the authors in the scope of the SMAVNET project¹ and in collaboration with sensefly² [7], [21].

In order to enable large outdoor aerial swarms, individual robots need to be inexpensive, safe and easy to operate by non-experts. The entire swarm should be deployable by a single person.

1) **PLATFORM:** We have developed flying platforms that are lightweight (420 g) and fly rather slowly (≤ 20 m/s). Thus, the limited kinetic energy (under 100 J) leads to a very low risk of serious damage or injury to 3rd parties in case of a failure [22]. The robots shown in Fig. 3 have an 80 cm wingspan, are built out of durable and flexible foam material and can be deployed by hand-launch. An electric motor is mounted at the back and two elevons (combined ailerons and elevator) serve as control surfaces. Each robot is equipped with an autopilot that ensures low-level control of altitude, airspeed and turn rate. Embedded in the autopilot is a micro-controller that runs a minimalist control strategy [23] based on input from only three sensors: one gyroscope and two pressure sensors. An interface is provided for receiving commands either from an autopilot-internal waypoint navigation module or from an embedded Linux board running swarm algorithms³.



Fig. 3. Aerial swarm composed of 10 flying-wing robots.

The output of the swarm controller, namely a desired turn rate, is sent as control command to the autopilot. In order to determine position and log flight trajectories, robots are further equipped with a u-blox⁴ LEA-5H GPS module. For *a priori* collision avoidance between robots, individual flight altitudes during the experiments remained constant with a spacing of 10 m and a ceiling at 150 m. The airspeed command remained constant at 12 m/s. An example of a typical swarm control architecture can be seen in Fig. 4.

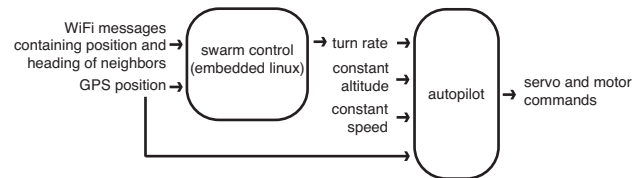


Fig. 4. Controller architecture with physically separated autopilot and swarm control.

¹<http://lis.epfl.ch/smavs>

²<http://sensefly.com>

³Colibri PXA270 by Toradex, <http://www.toradex.com>

⁴<http://www.u-blox.com>

2) **COMMUNICATION:** An off-the-shelf USB WiFi dongle, which is connected to the Linux computer, enables inter-robot communication. Netgear⁵ WNDA3100 dongles were used that implement the 802.11n standard and transmit in the 5 GHz band. This is interesting with respect to transmissions in the 2.4 GHz band because it allows for less interference with the considerable number of devices currently used in this band. Dongles are configured for ad-hoc mode and have two selectable line-of-sight communication ranges of 50 m and 300 m for the purpose of this paper as shown in Fig. 5. Reducing the communication range is done by discarding messages that have a signal strength below a predefined threshold in the WiFi drivers.

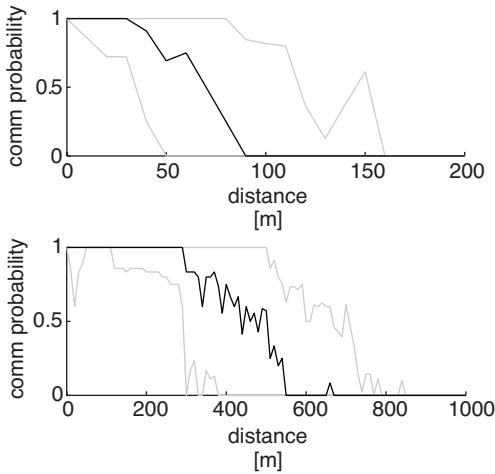


Fig. 5. Probability of receiving a message from another robot as a function of distance between the robots. Measures were taken during in-flight experiments with 10 robots for two different WiFi driver settings [21]. The three lines represent the lower quartile, median and upper quartile.

3) **SWARM EXPERIMENTATION PROCEDURE:** An experimental protocol was designed to perform successful swarm experiments in a safe and efficient manner using a single operator. It follows years of experience with hundreds of flights and can be adapted to most swarm systems. This setup allows for experiments of around 30 min (battery charge) with at least 10 robots. The protocol starts by booting all the robots on the ground and waiting for them to have a GPS fix, which typically takes 1-3 minutes. During this time, the operator prepares a ground station that will be used to monitor and interact with the swarm [24]. The robots are then launched one after the other by tilting their nose to the sky, which is assimilated to a take-off command. This is much quicker than having to click a button on the interface for every robot and allows the operator to move away from the computer. After take-off, robots navigate to a “stand-by” waypoint which they continuously circle. Flocking is then initiated by broadcasting a command called “swarm” from the ground station to the robots. On reception of this message, robots adopt turn rates sent from swarm algorithms implemented on the embedded Linux board to the autopilot (Fig. 4). Experiments can be stopped at any

time by sending the swarm command “stand-by”, at which point the robots return to the previous waypoint. This allows for quick swarm experiments that can be stopped at any time. The interface can also be used to change controller parameters online, updates can be made at the level of the swarm or the individual. When experiments are finished, the operator initiates automatic landing by sending the swarm command “land”. The robots then need to be retrieved around the landing site. Carefree operation is ensured through safety modes that prevent robots from escaping and that make robots land in case of low battery voltage. Notice that the entire swarm would still operate and land safely if the ground station is turned off.

B. SIMULATION

Experiments in simulation were conducted using a 3D simulator which realistically models robot trajectories, sensors, and communication described in the previous section. The simulator is event-based in order to model the fact that each robot has its own internal clock and that communication is in general asynchronous.

Robot motion is approximated using a first order model that produces trajectories similar to those performed by the actual robots [21]. Equations 1 through 3 modify the position (x, y, z) of the platforms after each time-step of duration dt based on a constant speed v and turn rate ω .

$$x(t) = x(t - dt) + v \cdot \cos(\omega \cdot dt) \cdot dt \quad (1)$$

$$y(t) = y(t - dt) + v \cdot \sin(\omega \cdot dt) \cdot dt \quad (2)$$

$$z(t) = \text{constant} \quad (3)$$

More in detail, robots fly at a speed of 12 m/s, affected by uniform noise in the range $[-1, 1]$ m/s. An additional permanent bias of ± 1 m/s is added to the speed, as is often the case in reality. Since fixed-wing platforms are unable to hover or make sharp turns, their turn rate is limited to the range $[-1, 1]$ rad/s. Uniform noise in the range $[-\pi/36, \pi/36]$ rad/s is added to the turn rate of the robot and a smoothing function ensures that it can not be modified abruptly.

Having a realistic communication model is essential for the credibility of our experiment because of real-life challenges brought on by highly dynamic systems, signal propagation uncertainties and network topologies prone to packet collisions. For this purpose, the simulator implements lower layers of the open systems interconnection (OSI) model, namely the network layer, data-link layer and physical layer for 802.11b wireless communications. The shadowing propagation model was used to probabilistically determine the range of inter-robot transmissions [25]. Parameters of the model can be tuned to achieve any desired range.

C. FLOCKING CONTROLLER

During flocking, robots are steered by changing their turn rate proportionally to the error between their current heading and their desired heading. The desired heading is given by the weighted sum of four vectors:

⁵<http://www.netgear.com>

- **Alignment:** mean velocity of all robots within communication range. This vector is normalized and given a weight of 1.
- **Cohesion:** vector pointing towards the center of mass of all robots within communication range. This vector is normalized and given a weight of 1.
- **Separation:** each robot within the communication range generates a repulsion vector that is inversely proportional to the distance separating the two robots. These vectors are then summed and the resulting vector is normalized and given a weight of 1.2 (smaller weights were not able to generate stable formations).
- **Migration:** vector pointing to the migration point. This vector is scaled by a factor 1/500. This means that if the robot is further than 500 m away from the migration point, the norm of the vector is greater than one, making this force dominant. Distances below 500 m produce a force with norm smaller than 1.

D. PERFORMANCE CRITERIA

Exhaustively exploring the impact of communication range and flight dynamics on aerial flocking is done in simulation. For each combination of parameters, with communication ranges in the range [0 m, 300 m] and turn rates between [0 rad/s, 1 rad/s], we run 10 independent swarm deployments that each last 15 min. The capacity of the robots to flock coherently is quantified using two measures.

The first shows how well robots can align their headings by measuring, for each parameter set, the standard deviation on all robot headings averaged over the last minute of all trials. Heading deviation is often used in the literature to characterize swarm convergence [26]. Measures are taken at the end of trials to ensure flocks have had time to form.

The second measure quantifies the capacity of robots to remain at the same relative distance from one another. For this purpose, the distance of every robot to all the other robots is measured and stored in a distance matrix A. This distance matrix is then measured again after 1 second and stored in matrix B. The absolute difference between both matrices ($|A - B|$) is then taken. We then compute the mean over all elements of this matrix, over each second interval during the last minute of the 10 trials. This measure is interesting because it does not require any a-priori knowledge about the formation that will emerge.

Experiments in simulation are then used as a basis to select three interesting parameter sets that will be used during experiments in reality. The first set aims to achieve stable flocking. We then consider two extreme parameter sets where the communication range or the maximum turn rate is small, causing incoherent flocking. The performance in reality is measured and compared to results in simulation and swarm behaviors are qualitatively analyzed.

III. RESULTS

A. PERFORMANCE

Fig. 6 shows the tradeoff between communication range and maximum turn rate in simulation. As expected robots

with small communication ranges of below 80 m and low maximum turn rates near 0.1 rad/s are not able to align their heading. As a worst case scenario, one can consider situations where robots can not turn (turn rate = 0) or communicate (communication range = 0). These cases lead to a standard deviation of around 1.1 rad. Instead, large communication ranges (300 m) and large maximum turn rates (1 rad/s) provide excellent heading alignment as shown by the near-0 standard deviation on heading. A clean gradient between these two frontiers shows the importance of considering both communication range and maximum turn rate when designing swarm systems. The same gradient can be observed when measuring the robots' ability to build a stable flock as shown in Fig. 7 .

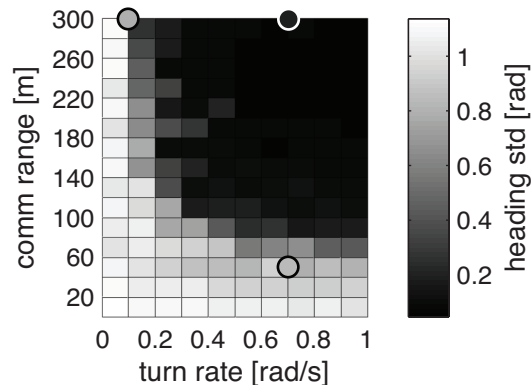


Fig. 6. Standard deviation on all robot headings averaged over the last minute of 10 simulated trials lasting 15 min. Good heading alignment expected in flocking is shown by low standard deviations. Three data points taken from experiments in reality are shown by circles.

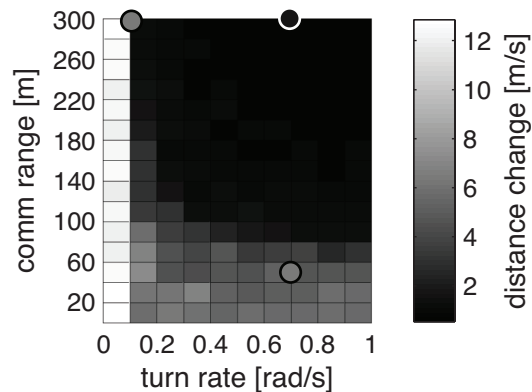


Fig. 7. Change in distance between all robots over one second, averaged over the last minute of 10 simulated trials lasting 15 min. Stable formations expected in flocking is shown by small changes in distance among the robots. Three data points taken from experiments in reality are shown by circles.

B. COMPARISON WITH REALITY

Coherent flocking with our setup can be achieved by using a communication range of 300 m and a maximum turn rate of 0.7 rad/s as shown in Fig. 7 and Fig. 6. Maximum turn

rates of 1 rad/s, were not chosen due to energy considerations since energy consumption increases with the tangent of the robot's roll angle. Fig. 8 shows snapshots of flock formations in simulation and reality. In both cases, robots move in a stable circular formation. During the entire experiment of 15 min, the flock remained coherent. The standard deviation on turn rate during the last minute of the trial in reality was of 0.197 rad while the change in distance between robots was of 2 m/s. Both values confirm the capacity of the flock to achieve heading alignment and a stable topology. These values compare well with those in simulation as shown in Fig. 7 and Fig. 6.

Fig. 9 shows the case where the maximum turn rate of the robots is small (0.1 rad/s), preventing them from making sharp turns. The result is that the robots fly much further away from the migration point and are unable to form a single coherent flock. In reality, robots had trouble keeping a turn rate of 0.1 and were often at lower turn rates. This explains that robots in reality would move further from the migration point. This further reinforces the idea that small changes in robot dynamics, occurring naturally in real world applications, can hinder an entire swarm system. The standard deviation on turn rate during the last minute of the trial in reality was of 0.80 rad while the change in distance between robots was of 6.21 m/s. These values are, as expected, worse than those measured in reality for communication ranges of 300 m and turn rates of 0.7. Notice the agreement between these measures and those taken in simulation (Fig. 7 and Fig. 6).

Finally, Fig. 10 shows the impact of small communication ranges (50 m) on the behavior of the flock. The result is that the robots form several mini-flocks rather than a single coherent one. The standard deviation on turn rate during the last minute of the trial in reality was of 0.85 rad while the change in distance between robots was of 6.55 m/s. These values are, as expected, worse than those measured in reality for communication ranges of 300 m and turn rates of 0.7.

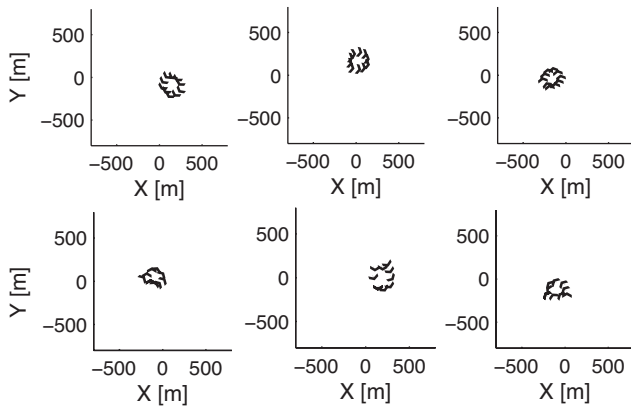


Fig. 8. Flock formations in simulation (top) and in reality (bottom) when robots have a communication range of 300 m and a maximum turn rate of 0.7 rad/s. Snapshots are taken at one minute intervals. The migration point is positioned at (0,0). Nine robots were launched during the experiment in reality.

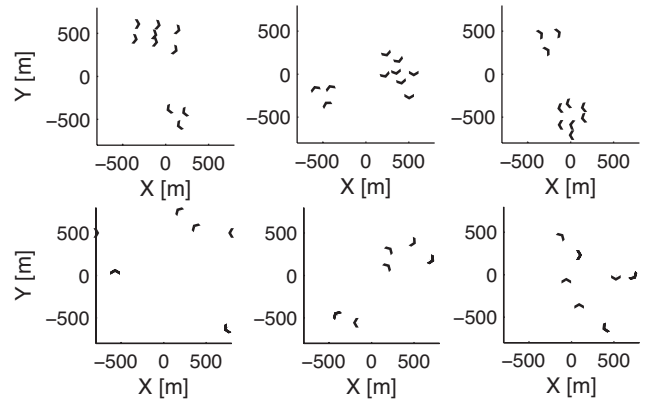


Fig. 9. Flock formations in simulation (top) and in reality (bottom) when robots have a communication range of 300 m and a maximum turn rate of 0.1 rad/s. Snapshots are taken at one minute intervals. The migration point is positioned at (0,0). Seven robots were launched during the experiment in reality.

As before, there is a clear correspondence with predictions made in simulation (Fig. 7 and Fig. 6).

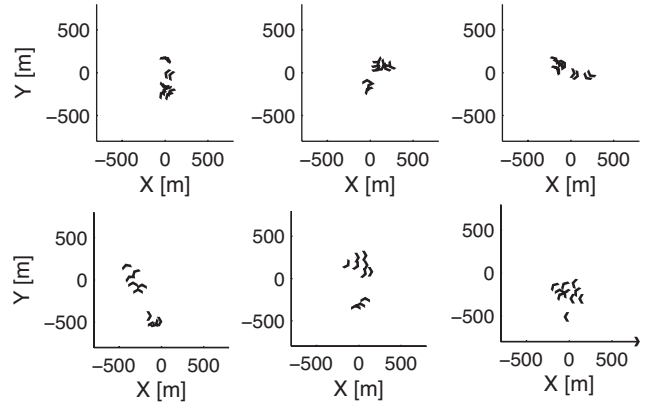


Fig. 10. Flock formations in simulation (top) and in reality (bottom) when robots have a communication range of 50 m and a maximum turn rate of 0.7 rad/s. Snapshots are taken at one minute intervals. The migration point is positioned at (0,0). Ten robots were launched during the experiment in reality.

The video submitted in supplementary material summarizes the results from this paper and the experimental setup.

IV. DISCUSSION

In general, flocking performance can be improved by increasing the maximum turn rate of the robots and their communication range. To set an upper bound on the maximum turn rate, one needs to consider the dynamics of the robot platform and the required energy needed to perform sharp turns. Likewise, the communication range should be limited based on energy, payload, and monetary constraints.

However, optimal settings for communication range might be altered when large numbers of robots are present. For example, large communication ranges could hinder the scalability of the swarm because many robots need to exchange information within a neighborhood. To avoid scalability issues while still maintaining good performance, one can

choose the smallest communication range for which coherent flocking is achieved. In our system, good parameters could be a maximum turn rate of 0.7 rad/s and a communication range of 150 m. An adequate choice in communication range and the distributed nature of Reynolds flocking should provide good scalability to the system. Using these parameters, preliminary results in simulation demonstrated coherent flocks of at least up to 100 robots.

V. CONCLUSION

Swarm behaviors often require robots to exchange information locally using wireless communication. Robots react to such messages by altering their trajectory. Challenges arise when the robot motion is disruptive to communication between robots. Such disruptions typically do not occur in the current literature because robots are assumed to have omnidirectional motion and relatively large communication ranges. In this paper, we considered the deployment of swarms of fixed-wing robots. This requires building flying robots that are low-cost, lightweight and energy efficient. As a result, the robots are not able to communicate at long ranges and their flight dynamics require them to maintain forward motion to avoid stalling and prevent them from making sharp turns. It therefore becomes crucial to consider the impact of both communication range and flight dynamics on swarm behaviors. As a case study, we implemented Reynolds rules to achieve robot flocking. Experiments in simulation clearly show the challenges in maintaining coherent flocks when the communication range is too small with respect to the maximum turn rate of the robots and when the turn rate is too small with respect to the communication range. Furthermore we dissected swarm behaviors when flocking worked and when parameters prevented flocking. Results are given in simulation and demonstrated thanks to three experiments involving up to 10 fully autonomous flying robots in outdoor experiments. The ability to deploy such large aerial swarms was made possible thanks to a new type of robotic platform, experimental protocol and swarm interface that enabled single operator operation [21]. In the future, we hope to investigate 3D flocking and a large variety of swarm algorithms applied to aerial systems.

VI. ACKNOWLEDGMENTS

This work is supported by armasuisse, competence sector Science + Technology for the Swiss Federal Department of Defense, Civil Protection and Sports and by the SNFS Sinergia project “Swarmix”, grant number CRSI22-133059. Fabio Ruini and Angelo Cangelosi were supported by EOARD grant 07-3075.

REFERENCES

- [1] A. Jadbabaie, J. Lin, and A. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.
- [2] N. Moshtagh and A. Jadbabaie, “Distributed geodesic control laws for flocking of nonholonomic agents,” *IEEE Transactions on Automatic Control*, vol. 52, no. 4, pp. 681–686, 2007.
- [3] R. Olfati-Saber, “Flocking for multi-agent dynamic systems: Algorithms and theory,” *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.
- [4] H. Tanner, A. Jadbabaie, and G. Pappas, “Stable flocking of mobile agents, Part I: Fixed topology,” in *Proceedings of the IEEE Conference on Decision and Control*, vol. 2. IEEE, 2004, pp. 2010–2015.
- [5] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, “Novel type of phase transition in a system of self-driven particles,” *Physical Review Letters*, vol. 75, no. 6, pp. 1226–1229, 1995.
- [6] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*, 2nd ed. Wiley, October 2003.
- [7] S. Hauert, “Evolutionary Synthesis of Communication-Based Aerial Swarms,” Ph.D. dissertation, Lausanne, 2010.
- [8] R. De Nardi, O. Holland, J. Woods, and A. Clark, “SwarMAV: A swarm of miniature aerial vehicles,” in *Proceedings of the 21st International UAV Systems Conference*, 2006.
- [9] J. Nembrini, N. Reeves, E. Poncet, A. Martinoli, and A. F. T. Winfield, “Mascarillon: Flying Swarm Intelligence for Architectural Research,” in *IEEE Swarm-Intelligence Symposium SIS’05*, 2005, pp. 225–232.
- [10] C. W. Reynolds, “Flocks, herds and schools: a distributed behavioral model,” in *SIGGRAPH Computer Graphics*, vol. 21. New York: ACM Press, 1987, pp. 25–34.
- [11] B. Crowther, “Rule-based guidance for flight vehicle flocking,” *Proceedings of the Institute of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 218, no. 2, pp. 111–124, 2004.
- [12] A. Hayes and P. Dormiani-Tabatabaei, “Self-organized flocking with agent failure: Off-line optimization and demonstration with real robots,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 4, 2002, pp. 3900–3905.
- [13] A. Turgut, H. Çelikkanat, F. Gökçe, and E. Şahin, “Self-organized flocking in mobile robot swarms,” *Swarm Intelligence*, vol. 2, no. 2, pp. 97–120, 2008.
- [14] C. Moeslinger, T. Schmickl, and K. Crailsheim, “A minimalist flocking algorithm for swarm robots,” in *Proceedings of the European Conference on Artificial Life*, vol. 9. Citeseer, 2009.
- [15] P. Basu, J. Redi, and V. Shurbanov, “Coordinated flocking of UAVs for improved connectivity of mobile ground nodes,” in *Proceedings of the IEEE Military Communications Conference*, vol. 3. Piscataway: IEEE Press, 2004, pp. 1628–1634.
- [16] D. Chang, S. Shadden, J. Marsden, and R. Olfati-Saber, “Collision avoidance for multiple agent systems,” in *Proceedings of the 42nd IEEE Conference on Decision and Control*. Piscataway: IEEE Press, 2003, pp. 539–543.
- [17] R. De Nardi, “Flocking of UAVs Software model and limited vision simulations,” Ph.D. dissertation, University of Essex, 2004.
- [18] J. Welsby and C. Melhuish, “Autonomous minimalist following in three dimensions: A study with small-scale dirigibles,” *Proceedings of Towards Intelligent Mobile Robots. Technical Report Series, Manchester University, Department of Computer Science*, 2001.
- [19] H. Tanner, A. Jadbabaie, and G. Pappas, “Flocking in teams of nonholonomic agents,” *Cooperative Control*, pp. 458–460, 2004.
- [20] A. Şamiloğlu, V. Gazi, and A. Koku, “Effects of asynchronism and neighborhood size on clustering in self-propelled particle systems,” *Computer and Information Sciences*, pp. 665–676, 2006.
- [21] S. Leven, “Enabling Large-Scale Collective Systems in Outdoor Aerial Robotics,” Ph.D. dissertation, Ecole Polytechnique Fédérale de Lausanne (EPFL), March 2011.
- [22] K. Dalamagkidis, K. P. Valavanis, and L. A. Piegl, “UAS safety assessment and functional requirements,” in *On Integrating Unmanned Aircraft Systems into the National Airspace System*. New York: Springer, 2009, ch. 5, pp. 73–75.
- [23] S. Leven, J.-C. Zufferey, and D. Floreano, “A minimalist control strategy for small UAVs,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. Piscataway: IEEE Press, October 2009, pp. 2873–2878.
- [24] A. Beyeler, S. Magnenat, and A. Habersaat, “Ishtar: a flexible and light-weight software for remote data access,” in *Proceedings of the European Micro Air Vehicle Conference and Competition*, Braunschweig, Germany, July 2008.
- [25] L. Fenton, “The sum of log-normal probability distributions in scatter transmission systems,” *Proceedings of the IEEE Transactions on Communications Systems*, vol. 8, no. 1, pp. 57–67, 1960.
- [26] H. Tanner, A. Jadbabaie, and G. Pappas, “Stable flocking of mobile agents part II: Dynamic topology,” in *Proceedings of the IEEE Conference on Decision and Control*, vol. 2. IEEE, 2004, pp. 2016–2021.