2016 IEEE International Conference on Robotics and Automation (ICRA)
Stockholm, Sweden, May 16-21, 2016

Live-Fly, Large-Scale Field Experimentation for Large Numbers of Fixed-Wing UAVs

Timothy H. Chung, Michael R. Clement, Michael A. Day, Kevin D. Jones, Duane Davis, Marianna Jones
Advanced Robotic Systems Engineering Laboratory
Naval Postgraduate School
Monterey, California USA
{thchung,mrclemen,maday,kdjones,dtdavi1,mjjones@nps.edu}

Abstract— **In this paper, we present extensive advances in live-fly field experimentation capabilities of large numbers of fixed-wing aerial robots, and highlight both the enabling technologies as well as the challenges addressed in such large-scale flight operations. We showcase results from recent field tests, including the *autonomous launch, flight, and landing of 50 UAVs*, which illuminate numerous operational lessons learned and generate rich multi-UAV datasets. We detail the design and open architecture of the testbed, which intentionally leverages low-cost and open-source components, aimed at promoting continued advances and alignment of multi-robot systems research and practice.**

I. Introduction

Robotics and unmanned systems are increasingly a critical element of current and future civilian and military applications, and their capabilities and associated technologies necessarily must continue to rapidly evolve to keep pace with increasingly expanding mission sets. The maturation of unmanned systems technology, most notably in unmanned aerial vehicles (UAVs), has benefited substantially from the confluence of relatively lower cost, increasingly easier access, and wider active engagement by open-source communities, leading to significant advances in the development and deployment of large numbers of autonomous systems.

Over the past decade, numerous research programs have developed fixed-wing, multi-UAV systems for outdoor field experimentation, similar to that showcased in this paper. Ranging from emphasis on platform and embedded systems design [1], [2], [3] to more focus on cooperative control algorithms [4], [5], [6], these previous efforts highlighted system designs for few (i.e., two to five) aerial robots fielded at a time. Other research projects, notably [7], are aimed at exploring swarm control and swarm flight operations, representing some of the latest advancements in operations of increasingly larger number of outdoor UAVs.

However, realization of these many-robot systems in real-world contexts, especially for increasingly larger numbers of robots, face not only technological hurdles, such as efficient coordination algorithms in lossy network environments, but also other key integration issues, including human factors for human-swarm interaction and logistics considerations for operating large numbers of robots. Such challenges merit a systems approach to holistically consider the various development and integration efforts across sub-system components, as well as potentially identify a need for increased community collaboration, through shared benchmark problems, consistent performance metrics, and possibly via an open or common testbed architecture.

Main contributions of this paper include description of recent advances in live-fly field experimentation of large numbers of aerial robots and highlight the enabling technologies supporting this multi-UAV testbed infrastructure, with which recent field tests successfully demonstrated 50 fixed-wing aerial robots conducting cooperative autonomous flight operations. Significant contributions in hardware, software, networking, human, and logistics systems are presented as well. Furthermore, this paper also presents an architecture, founded on open principles, that addresses not only those metrics relevant for collective autonomous systems, but also facilitates definition of operationally relevant measures of effectiveness for such large-scale experimentation capabilities. This architecture also promotes the creation, aggregation, and dissemination of various unique datasets of potential value to the research and developer community.

The multi-UAV systems architecture is presented in the following section (Section II), to include description of the hardware, software, networking, human-swarm interaction, and logistics infrastructure necessary to enable field tests of this magnitude. An overview of the field experimental setup and high-level summary of the recent 50-UAV mission is described in Section III, after which we characterize and analyze the system performance. Key lessons learned of broader value to the multi-robot systems community are summarized in Section IV, with the envisioned avenues of future research and development also outlined within.

II. Systems Architecture

A. Concept of Operations for Live-Fly Swarm UAVs

In developing the capabilities to conduct live-fly experiments, we identified the fundamental phases of operations, which both enabled and informed the design and implementation of the architecture as well as enabling technologies, such as operator interfaces and/or collective behavior algorithms. These operational phases are:

- **Pre-Flight**: Preparation of UAVs by Pre-Flight Technicians, including individual system health checks, proper mission and software loading, and battery and camera installation;
- **Launch**: Individual sequential deployment of UAVs via the Launch Operator and Launch Crew, using either a

bungee launcher or, more recently, an electric-powered launching system

- **Ingress**: Safe transit of individual UAVs after automatic takeoff to flight operating area, including climbing to (possibly pre-defined) appropriate altitudes for deconflicted operations;
- **Swarm Ready**: State of individual UAVs standing by for assignment to a specific sub-swarm prior to or while awaiting initiation of a swarm behavior;
- **Swarm Active**: State of individual UAVs within a sub-swarm actively executing a specified swarm behavior, such as leader-follower or swarm search;
- **Egress**: Safe transit out of the flight operating area to a holding area prior to transfer to the landing queue
- **Landing**: Sequential landing of UAVs, either by individual or collective (i.e., sub-swarm) command, including autonomous landing routine for descent and final approach to target landing point;
- **Recovery and Post-Flight**: Retrieval and power-down of UAVs from landing area by Recovery Crew, and post-flight inspection for maintenance and/or repair needs by Post-Flight Technicians.

Definition of these phases have informed and reflected substantial development efforts in both technologies and processes, and highlight the tight integration of the myriad systems comprising the field experimentation capability described herein.

B. Flight Systems

Figure 1 illustrates the NPS ARSENL's *ZephyrII* UAV. The airframe selection was driven by several criteria of varying significance, including: mission capabilities, size, weight, flight speed, configuration, endurance, and cost, among others.

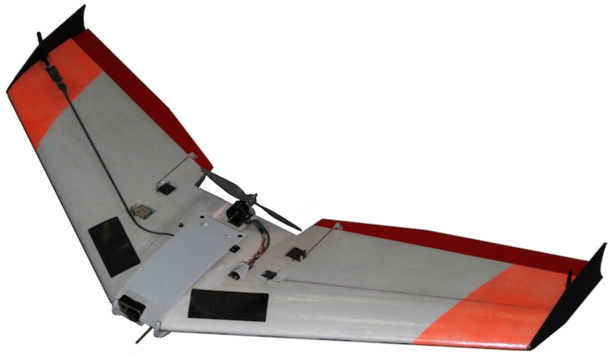


Fig. 1. Picture of NPS *ZephyrII* UAV, a low-cost yet capable system leveraging open-source and commercially available components.

Size and weight were more a function of required payload capacity (a moving target) and endurance, but with a further condition to keep the airframe as small and light as possible to minimize system cost and the potential risk in the event of an accident. The configuration was driven primarily by flight performance and durability, but was also influenced by system cost and complexity. Factors like cost, complexity and packing dimensions become significant when the fleet size gets large, 50 or more aircraft in this case. Early on a goal was set to try to keep the per aircraft system cost (excluding labor) to US$1000 or less.

TABLE I

OPERATIONAL SPECIFICATIONS FOR CURRENT GENERATION (GEN 7) OF THE NPS *ZephyrII* UAV PLATFORM

Wingspan	1.45 meters
Nominal Endurance	50 minutes
Cruise Speed	18 m/s
Takeoff Weight	2.5 kg

In order to meet this goal, the system is comprised almost entirely from off-the-shelf hobby equipment, with the addition of a few printed plastic parts and a printed circuit board to consolidate much of the power and signal routing of the propulsion and flight control systems. The primary components are listed in Table II. Polycarbonate 3D printed parts are used to facilitate a reduction of labor in the assembly process, ease of use in specific areas, and improved durability in others. For example, a printed nose section includes a socket to hold the camera and the pitot-static tube. The use of printed plastic makes it possible to have a spring-loaded latch to hold the camera securely in place, and to allow the pitot-static tube to slide into the wing to reduce the possibility of damage to the tube during transport or in a rough landing. The printed part uses about $8 of material, but saves an hour or more of labor (per plane), and improves servicability in the field.

TABLE II

LISTING OF CORE FLIGHT SYSTEM COMPONENTS

Airframe	Ritewing *ZephyrII* (EPO foam core)
Avionics	Pixhawk Autopilot
Autonomy CPU	Hardkernel ODroid U3
Telemetry link	3DR Telemetry Radio (915MHz)
Payload link	Alfa WiFi USB radio adapter (2.4GHz)
Front-facing camera	GoPro Hero 3
Power	2 × Thunder Power ProLite 3S 5000mAHr
Propulsion	OS 3820-1200W
ESC	Castle Creations EdgeLite 50
Propeller	APC 11x5.5
Servos	2 × Hitec HS5245MG
GPS/Compass	3DR uBlox LEA-6H/compass
Airspeed sensor	MS4525DO/3DR Pitot-Static tube
RC TX/RX	Spektrum DX9 and satellite Rx

The floor of the avionics bay is fitted with a printed circuit board (PCB) which routes power from the batteries to the various systems and includes several key components. The end result is a part that is assembled off-site at low cost and greatly reduces the wiring clutter within the bay. The PCB includes plugs to mate with the battery leads, a plug to route power to the electronic speed control (ESC), voltage and current sensing for the autopilot, voltage regulation for the autopilot, payload and servos, and a digital airspeed sensor. The autopilot and payload computer are attached to the PCB using printed parts, such that the PCB can be pulled from the aircraft and utilized as an entire, standalone avionics package for benchtop testing and HITL testing. The PCB is shown in Figure 2.

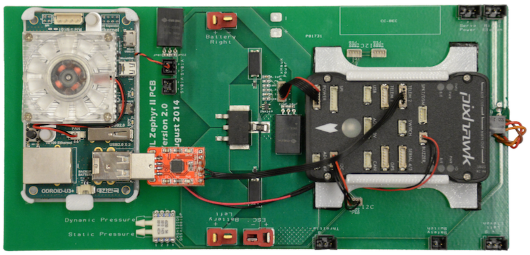


Fig. 2. ARSENL's custom printed circuit board for *ZephyrII*, providing power distribution, airspeed and power sensing, wiring interfaces, and 3D-printed mounts for autopilot and autonomy computer modules.

C. Software Systems

In addition to the autopilot, which handles lower-level guidance, navigation, and control tasks, we have implemented an "autonomy payload" to handle higher-level planning and coordination tasks. The payload computer (also termed a "companion computer" in the community), an ODroid U3, provides compute power similar to that of a modern smartphone and is capable of running Linux and the Robot Operating System (ROS). All of the payload's software components exclusively leverage ROS for inter-component communications, with two exceptions: an *autopilot bridge* that serves as the intermediary between the autopilot and other ROS-speaking components, and a *network bridge* that facilitates plane-to-plane and plane-to-ground communication.

ROS supports both publish-subscribe and client-server communication models, though the publish-subscribe model is more commonly used. The autonomy payload "core" comprises the following ROS nodes:

- `/autopilot`: The MAVLink-ROS bridge, which publishes autopilot position and status messages in ROS form, and forwards ROS command messages via the MAVLink messaging standard to the autopilot.
- `/network`: The ROS-Network bridge, which enables aircraft sharing their position and status with other flying aircraft and ground stations, and relays ground station network commands to the aircraft payload.
- `/swarm_tracker`: All aircraft share position and status information across a communications network, as described below. Position updates received from other aircraft are aggregated by the swarm tracker node, and a dead-reckoned snapshot of the entire swarm is produced at 10 Hz. This data product is used by algorithms that implement cooperative behaviors.
- `/swarm_manager`: This node is responsible for directing the individual aircraft's interaction with the swarm or subswarm through parameterization, activation, and deactivation available control modes. As control modes and swarm behaviors are implemented, this node's functionality will be expanded as required to manage individual aircraft participation in swarm activities.
- `/ctlr_selector`: This node is responsible for switching between various autonomous behaviors and for ensuring that any preconditions for safe execution of a behavior are met prior to its activation. It also monitors for and prevents various illegal situations, such as multiple conflicting behaviors being activated simultaneously. Upon detection of an unexpected or potentially unsafe controller state, the node deactivates all payload controllers and directs the aircraft to a safe loiter position.

Atop this core architecture, we develop various "controllers" that enable individual aircraft to participate in solitary or cooperative behaviors. Two examples of controllers are a "waypoint sequencer" that emulates the autopilot behavior of traveling along a given sequence of (latitude, longitude, altitude) points, and a "follower" that causes an aircraft to follow another specified "leader" aircraft based on the leader's communicated position.

Individual controllers are implemented using a deterministic finite automata model with three reachable states as depicted in Figure 3. This model ensures safe activation and deactivation of controllers by preventing transition to any state in which a controller is active despite being loaded with invalid parameters.

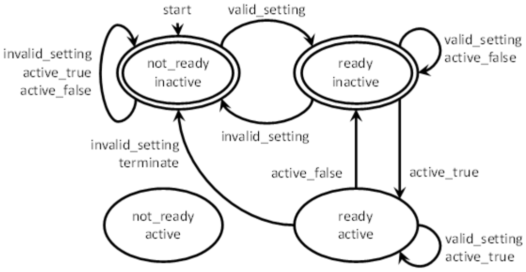


Fig. 3. Illustration of the deterministic finite automata model used to govern controller activation onboard each UAV.

At present, all controllers implement waypoint-driven navigation; that is, they generate (latitude, longitude, altitude) tuples that are sent via the autopilot bridge to the autopilot, which in turn updates the present waypoint to that new position. Effectively, this means that the controller "moves the carrot" and it is up to the autopilot (the rabbit) to determine how to navigate to the carrot. While numerous methods for smooth trajectory tracking using waypoint-based navigation have been developed and implemented [8], [9], we expect further development to focus on enhancing rate-based approaches (i.e., forward speed, turn rate, and climb rate) for control to enable aggressive flight, a desired direction going forward.

D. Network Systems

While conventional communications architectures often rely on a single, high data rate connection to the ground for each aircraft, such paradigms suffer when attempting to scale to significantly large numbers. Furthermore, in the context of collaborative autonomy, planes must be generally aware of the states (e.g., positions) of other planes and able either to receive succinct commands from a ground station or to coordinate the collective behavior directly with other planes.

As such, all aircraft are equipped with three communications systems: an 802.11n wireless radio that participates in an ad hoc (optionally meshed) network with other aircraft and ground stations; a radio control (RC) receiver for manually piloted flight; and a serial "telemetry" radio that provides two-way communication directly with the autopilot onboard a single plane. We rely solely on the 802.11n network as our primary communications link for command and coordination; we reserve the RC and telemetry links for debugging and emergency override as auxiliary means to communicate with individual aircraft.

In addressing the unique needs for collaborative autonomy for large numbers, we have prototyped a custom protocol for inter-aircraft and ground-to-aircraft communications, implemented on top of UDP/IP. The majority of the command and coordination messages we wish to send are different from other UAS command protocols such as MAVLink in that we command the "swarm," not individual aircraft. Commands are frequently sent as a single broadcast to many or all aircraft, informing them of the overall intent; they must then negotiate among themselves to determine aircraft-specific roles and parameters. Rather than repurpose another protocol's existing messages with new semantics or extend it with a plethora of new message variants, we are able to form a distinct set of messages and a common header that best suits the mission set. Using this custom protocol, UAVs transmit broadcast messages over the 2.4GHz link, providing state updates (to include, e.g., GPS latitude, GPS longitude, and barometric altitude) at 10Hz, and status messages (which contain UAV and sub-swarm identifiers as well as other system health flags) at 2Hz.

E. Human-Swarm Interaction Systems

Interacting with a large number of aerial robots necessarily requires a shift in perspective from traditional approaches for controlling one or a few UAVs, transitioning from a pilot or operator role towards more of a "mission manager" level of interaction. While numerous ground control stations have been developed intended for multi-UAV mission sets, the majority of them are still largely predicated on providing flight-system level situational awareness and control of individual elements within the collective. These approaches are often unable to scale well with increasing numbers of UAVs, due to the overwhelming cognitive workload for the operator, let alone the implications for excessive demands on the network bandwidth for transmitting heavy streams of telemetry data and individual UAV commands. Experiences in our previous field experiments also qualitatively corroborate these findings of cognitive overload and operator fatigue due to, e.g., task and/or context switching, preventing expansion past 5-6 UAVs in simultaneous operations when using conventional approaches and interfaces. In this context, traditional thought proposes the employment of a team of operators to achieve larger numbers, with each operator wholly responsible for a subset, e.g., 5-10 UAVs each, of the total collective (see Figure 4(a)). In this case, the operator team size is forced to increase with the number of robots, which begins to be undesirable, if not infeasible, for larger numbers, e.g., beyond ten, of UAVs.

Instead, we choose to functionally decompose the responsibilities of the swarm mission management into that of system health monitoring, managed by the "swarm monitor," and swarm behavior execution, managed by the "swarm operator," as illustrated in Figure 4(b).

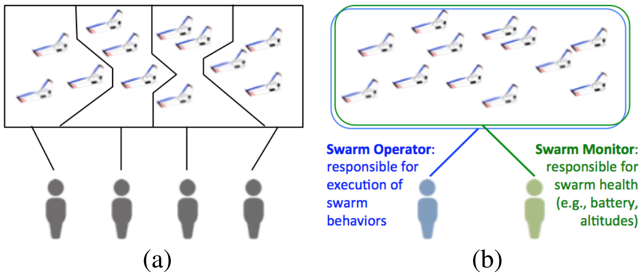

Swarm Operator: responsible for execution of swarm behaviors
Swarm Monitor: responsible for swarm health (e.g., battery, altitudes)
(a)
(b)

Fig. 4. Live-fly flight operations with larger numbers of UAVs have been demonstrated by shifting away from (a) partitioning the swarm into sub-teams, each with a dedicated operator, to (b) decomposing the responsibilities into *health monitoring* and *swarm behavior execution*.

This two-person partitioning by *function* rather than by *platforms*, with their respective tailored user interfaces, has successfully enabled live-fly experimentation with larger swarm sizes, including the recent 50-UAV mission. Active ongoing efforts seek to develop more intuitive and multi-modal interfaces encouraging this "swarm monitor" and "swarm operator" perspective.

F. Logistics Systems

In addition (and perhaps more important) to the enabling technologies for large numbers of UAVs, the logistics of maintaining, preparing, and deploying the UAV fleet pose significant challenges. The many facets of the logistics challenges when addressing the magnitude of operations and scale of experimentation include: UAV fleet maintenance and management (e.g., tracking repairs and configuration); software systems management (e.g., updating onboard firmware or mission description files); supply-chain considerations (e.g., availability of replacement parts); and transportation and storage (e.g., how and/or where to fit 80+ UAVs).

One of the major challenges to large-scale, large-number, swarm UAV field experimentation capabilities is the need to conduct pre-flight procedures to prepare aircraft for flight operations. These critical processes include: mechanical inspection of the airframes; functional checks of control surfaces and communication links; verification of system parameters and settings; calibration of sensors (e.g., barometer); and installation of batteries and camera. While such procedures are part of preparing any flight system, when scaling to large numbers, the time necessary to perform these checks becomes an issue. As an example, even if each aircraft can be cleared for flight in 20 minutes (which is significantly shorter than most operational UAV pre-flight times), preparation of 50 UAVs would take over sixteen hours! Further accounting for operational availability (as determined from previous experience) of the platforms, i.e., needing to "prep four UAVs to fly three," a single pre-flight station would require over 22 hours to prepare 67 UAVs.

Through continued refinement and iteration, as well as with significant development efforts devoted to in-house logistics automation software that simplifies the process while ensuring quality of checks, these pre-flight procedures have been substantially streamlined. With three pre-flight stations in parallel, the pre-flight phase is allotted nearly four hours with pre-flight checks taking less than 10 minutes per UAV.

Another problem that is faced when performing field tests with large numbers of UAVs is the endurance limit of the platforms imposed by the battery capacities. With about 50 minutes of flight time possible, the launch of 50 aircraft becomes a severe challenge. As with pre-flight checks, much attention has been on streamlining the launch procedures for expediency, including a "launch operator interface" software utility to accelerate necessary system status checks prior to launch. Another enabling technology is the design, implementation, and field demonstration of a robotic UAV launcher, shown in Figure 5, which leverages an electric motor to propel the UAV. In addition to being able to physically launch a UAV nearly every three seconds, the launcher itself carries an onboard computer running ROS to: interface with the launch motor speed controller; listen to forward and aft proximity sensors for personnel safety; read RFID tags affixed to each UAV when resting on the launcher rails; run a teleoperation node to maneuver the launcher itself, e.g., to re-position into the wind; and provide wireless remote control via game controller.

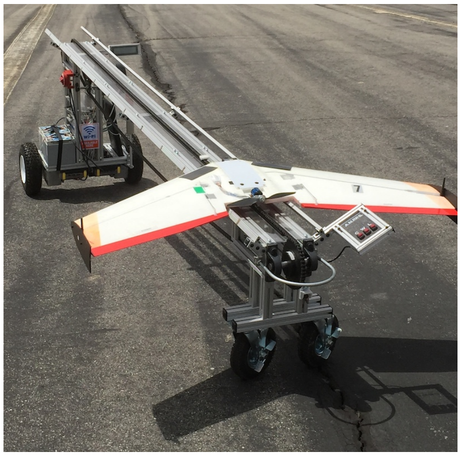


Fig. 5. Picture of the ARSENL *Automated Multi-Plane Propulsion System* (AMPPS) robotic launcher, used in the 50-UAV mission for rapid launch. AMPPS uses an electric motor to launch the UAV, and is equipped with, e.g., ROS-enabled sensors, differential drive motors, and gamepad for triggering launch and teleoperation.

The launcher also reduces the number of personnel necessary to conduct the physical launch, as compared to the bungee-based launcher, which further enhances the field experimentation capabilities.

G. Swarm Mission Planning

Several key considerations are of operational relevance when constructing a mission for large numbers of fixed-wing aircraft in live-fly field experimentation. Some of these issues include: sufficient airspace for maneuvering aircraft, minimal risk to personnel and property, high likelihood of containment, adequate communications link quality and range, as well as others. Further, given the different phases of swarming operations (see Section II-A), spatial configuration of staging regions and designated areas for swarm flight operations must also be considered.

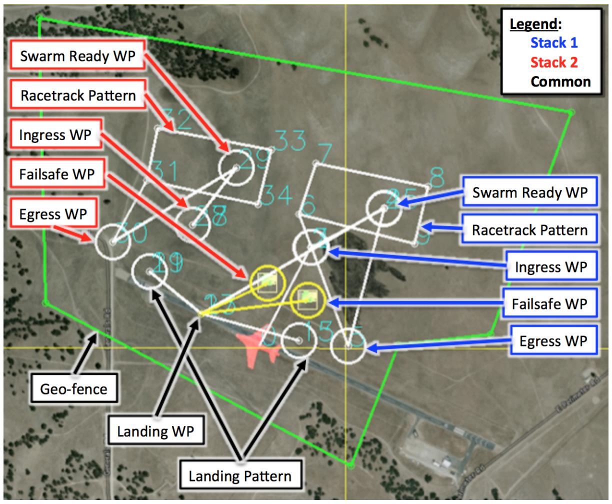


Fig. 6. Illustration of the two-stack configuration for the swarm mission, designed to support the desired concept of swarm operations and to facilitate safe yet compact flight experiments. Each UAV is assigned to an altitude-deconflicted slot within one of the two stacks.

Figure 6 illustrates the mission plan, which involves two laterally deconflicted "stacks" of 25 UAVs each, with UAVs allocated to different 15-meter altitude slots within each stack. This configuration allows for a balance between safety of operations (e.g., mitigate uncertainty present in GPS altitude measurements) while keeping the effective airspace and swarm footprint relatively compact. Twenty-five UAVs are deployed first to Stack 1, with the next 25 UAVs deployed to Stack 2.

Both stacks have their respective ingress, egress, and failsafe locations, as well as a standard "racetrack" pattern and the Swarm Ready waypoint, where aircraft awaiting tasking for collective behavior maintain an orbit. Two landing patterns are included in the mission, with selection of one occurring just prior to landing depending on the prevailing wind direction, with a targeted landing location common to all UAVs. Furthermore, a software-enabled failsafe for containment is provided by the geo-fence feature in the open-source autopilot software.

III. Swarm Analysis and Experimental Results

Recent field experiments have demonstrated substantial and rapid advances in the team's ability to deploy large numbers of aerial robots, culminating in the live-fly flight tests with fifty UAVs described in this paper. Figure 7 illustrates the evolution of the experiment, which includes the rapid sequential launch of UAVs, followed by a period lasting nearly 9 minutes and 45 seconds where 50 aircraft are simultaneously aloft and autonomously operated, and the subsequent aggregate landing of the fleet concluding the flight operations. One can note the anomaly occurring near 12:49:00, where an airborne UAV exhibited anomalous behaviors and was manually extracted from the flight stack and safely landed from flight. To still enable 50 UAVs aloft, we launched an extra aircraft for a total of 51 sorties. Having only one UAV exhibit faulty behaviors is credited to extensive and careful pre-flight procedures; for this test, 68 total aircraft were prepared, in anticipation of needing four flight-ready UAVs for every three we wanted to fly.

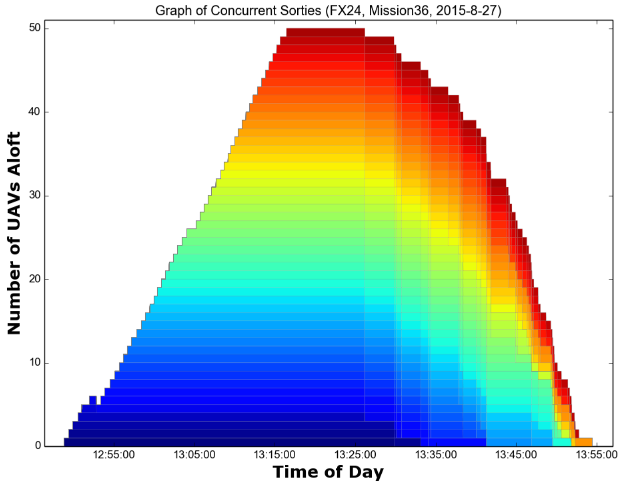


Fig. 7. Illustration of the evolution of the 50-UAV flight experiment, showing the number of sorties (i.e., deployed UAVs) aloft and depicting the sequential launch, effective mission time of 9:45 minutes where all 50 UAVs are in flight, and the sequential landing and recovery of all aircraft.

A. Launch Rate

Despite the challenges faced by the system during the employment phases of operation, significant attention must be given to *deployment* concepts of getting large numbers of aerial robots into the sky. Furthermore, the processes for launching these systems are also constrained by other components, such as battery endurance once the UAVs are aloft. The key measure of performance defined for the deployment of the swarm fleet is the mean time between launches, or:

$$MTBL = \frac{1}{N}\left(\tau^{1}_{\text{launch}} + \sum_{i=2}^{N}\left(\tau^{i}_{\text{launch}} - \tau^{i-1}_{\text{launch}}\right)\right),$$

where τ^{i}_{launch} is the time until the launch of the i^{th} UAV since commencing launch operations for N UAVs total.

For the 50-UAV mission, battery endurance limitations required a mean time between launches of 30-45 seconds to result in a total launch time that would leave sufficient time remaining for swarming behavior execution as well as safe egress and landing of all aircraft. Figure 8 illustrates the time to launch each UAV, with a resulting mean time between launches $MTBL \approx 33.5$ seconds, satisfying our operational performance requirements.

B. Swarm Behaviors

1) Leader-Follower: Numerous approaches exist for cooperative control for formations, to include leader-follower configurations, e.g., [10], [11]. To motivate the development of multi-UAS capabilities, we employ a simple model that accounts for the leader's predicted position and the follower's desired relative geometry in the plane, with relative altitude separations controlled independently.

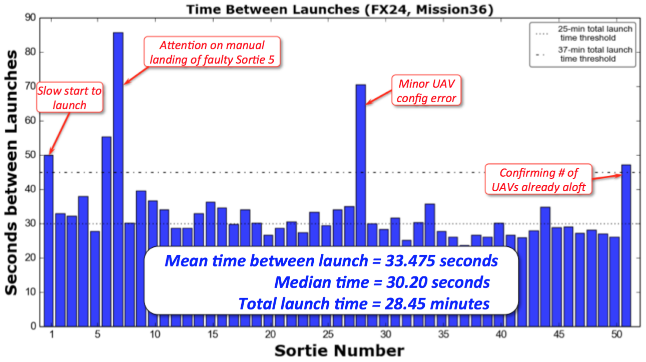
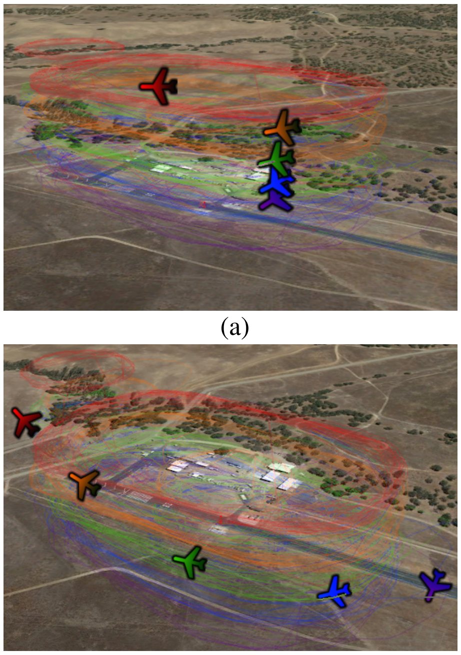


Fig. 8. Launch times for the 50-UAV mission, with a mean time between launches of 33.5 seconds. This resulted in a total launch time of just over 28 minutes, providing sufficient time for executing instances of collective behaviors, even with infrequent unexpected delays in the launch sequence (as annotated in red).

For multiple aircraft, we model the leader-follower relationships between UAVs as defined by an acyclic directed graph, such that each aircraft (except for the root leader) has one and only one leader. While many configurations are possible, we consider two basic configurations for live-fly validation as depicted in Figure 10.



(a)

(b)

Fig. 9. Live-fly examples of leader-follower of two basic configurations, where (a) all followers trail a single leader, or (b) followers are in a (close) trail formation

Leader-election methods can also be utilized and, for the case of our basic implementation used in experiments detailed in this paper, are conducted by each UAV individually identifying the highest altitude member of its sub-swarm based on that UAV's internal model of all swarm element states (i.e., via the `swarm_manager` node, c.f. Section II-C). This leader-election mechanism (ideally) leads to the single-leader configuration illustrated in Figure 10.

We can assess the formation following behavior using mean position error as a function of time, $MPE(t)$ (Equation 1), over all sub-swarm members as well as its time-average, $AMPE$ (Equation 2), e.g., [12]):

$$MPE(t) = \frac{1}{N}\sum_{i=1}^{N} \|\boldsymbol{p}_{\boldsymbol{G}}^{i}(t) - \boldsymbol{p}_{\boldsymbol{F}}^{i}(t)\|, \tag{1}$$

$$AMPE = \frac{1}{T_N}\sum_{t=1}^{T_N} MPE(t), \tag{2}$$

where N is the number of robots performing the formation keeping during a length of time denoted T_N. Note that during a given flight experiment, sub-swarms may have different numbers of robots for different durations, so further consideration should be given to define appropriate measures of performance over any given experiment.

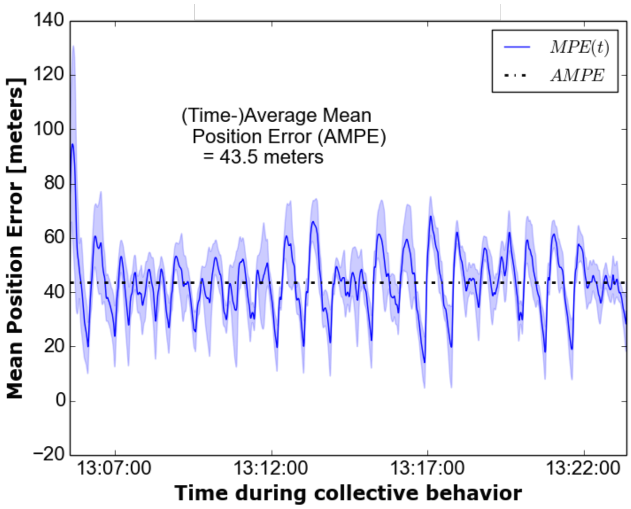
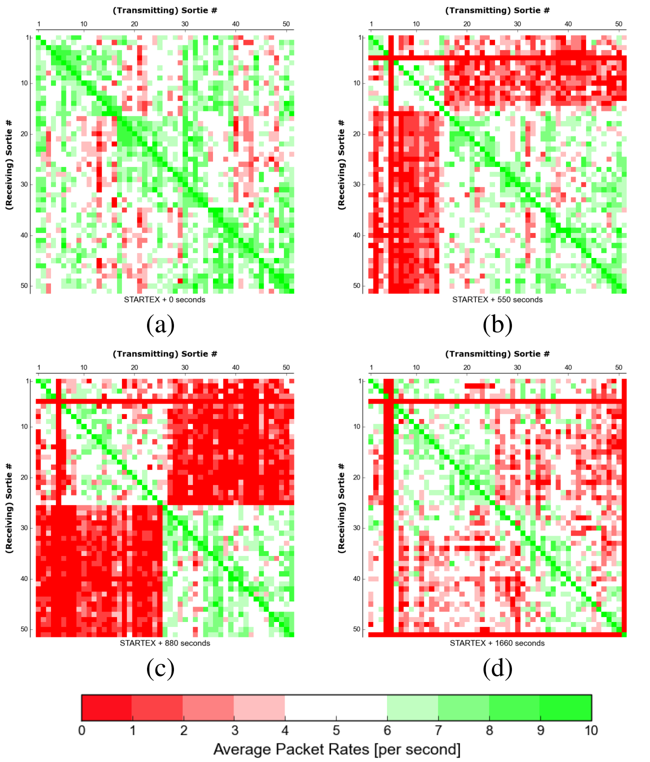


Fig. 10. Mean Position Error, $MPE(t)$, as a function of time for a segment during the 50-UAV mission where the leader-follower formation behavior was active with $N = 20$ UAVs. The lightly shaded band represents 1σ error.

C. Performance of a Flying Ad Hoc Swarm Network

A critical aspect of any collaborative autonomous capability is the ability to communicate effectively and efficiently among teammates. While there has been substantial efforts in mobile ad hoc networks for terrestrial and vehicle-borne systems, substantive and rigorous characterization of communications performance for large-scale, highly mobile, and dynamic airborne networks has largely been limited to simulation models or flight tests with limited numbers of nodes [13]. The presented multi-UAV capability provides further benefit as a testbed for investigation of such flying ad hoc networks (FANETs) [14].

While data throughput may be paramount in many conventional networks, the challenges of latency (i.e., delay) and packet (delivery) rate across the wireless medium are critical for fast-moving nodes, highly dynamic topologies, and time-sensitive coordination behaviors requiring timely updates from sub-swarm team members. In order to better characterize such systems, we can collect relevant data from our live-fly systems and measure, e.g., packet delivery rates.



(a) (b) (c) (d)

Fig. 11. Network performance as measured by packet rate between aircraft, i.e., packet rate observed at UAV i (row) received from UAV j (column), over the course of the 50-UAV flight test. Packet rates shown: (a) at start of the experiment ($t = 0$), where all 50 aircraft are still on-deck; (b) after the first 15 UAVs have been launched ($t = 550$ seconds); (c) after the first 25 UAVs are airborne into Stack 1 ($t = 880$ seconds); and (d) once all 50 UAVs are aloft ($t = 1660$ seconds).

Such characterizations may be used to inform other system design decisions, such as spatial configurations of swarm elements and/or types of viable collaborative autonomy algorithms.

Figure 11 illustrates empirical results from network traffic for the 50-UAV field test, for which receipt of packets (containing state data of the source UAV) are recorded at each UAV and aggregated over a one-second window to construct the average packet rate (i.e., measured in packets received per second). Each colored element in the illustrated matrix corresponds to the average packet rate received by the i^{th} UAV (row) from the j^{th} UAV (column), ordered by sortie number, 1 through 51.

Figure 11(a) illustrates the network quality immediately prior to the first sortie's launch (defined as $t = 0$), i.e., all aircraft still on the ground. In general, aircraft are largely able to communicate with one another, enjoying relatively high packet arrival rates of 6-10 packets per second. (Recall that state messages are broadcast at 10 Hz by all UAVs.)

As deployment gets underway, Figure 11(b) illustrates the network performance after launching the first 15 UAVs (i.e., $t = 550$ seconds after first UAV launch). It is evident that communication is reasonable amongst robots that are aloft and, separately, among those that are still on the ground. However, communications between air-to-ground or ground-to-air via the wireless link are rather degraded. Such insight is especially useful when designing swarm behaviors that might require coordination between those groups, e.g., for self-deconfliction algorithms during takeoff.[1]

Figure 11(c) showcases the state of the network at $t = 880$ seconds, at which point the first 25 UAVs have been launched into their flight operations areas. The partitioning between aloft and on-deck aircraft is even more dramatic than for the 15-UAV case, with nearly no communications extended between air and ground. Another interesting note can be observed, in that upon detection of a UAV's launch, each aircraft automatically increases the transmit power of its onboard WiFi radios (from 10dBm to 20dBm). However, these aloft aircraft are now highly mobile and spatially dispersed, and as such, appear to have a slight degradation in average packet rates, in comparison to their grounded counterparts. Such points merit further investigation into the impact of dispersion, relative attitude (e.g., antenna polarization), transmit power, and other factors governing RF and network performance for these flying ad hoc networks.

Finally, Figure 11(d) represents the case once all 50 aircraft have been successfully deployed and are airborne at $t = 1660$ seconds. Here, the fact that each group of 25 UAVs has their flight operations areas separated laterally is visible in the continued partitioning of network performance. However, unlike the prior case where some aircraft were aloft and others were on the ground, as all aircraft are aloft, we observe links with lower (but not zero) packet delivery rates between the two groups of 25 UAVs. This fact may likely be attributed to distance-dependent fading, and future analysis may yield empirical validation of relevant RF propagation models, with parameter estimates of relevant coefficients to inform future employment concepts of these aerial networks.

IV. Summary of Contributions and Future Work

This paper presented a live-fly field test capability for large numbers of aerial robots that serves as a testbed for advances in collaborative autonomy while facilitating a holistic systems approach to integration and experimentation. Video highlights of this successful 50-UAV mission can be found at `https://youtu.be/2T_tt5j1rpA`. In addition to detailing the open architecture design and development methodology, which leverages open-source and low-cost commercial resources as well as custom enabling technologies developed in-house, we highlighted recent field experiments where we successfully launched, flew, and landed fifty autonomous, fixed-wing aerial robots in live-fly operations. The lessons learned captured in this paper also help to inform the types of performance metrics and benchmark studies relevant to large-scale, field experimentation research of interest to the broader robotics community.

Efforts described in this paper readily identify numerous avenues for further research and development. Of significant interest is in the investigation and implementation of various classes of collective behaviors and coordination algorithms of interest to large numbers of aerial autonomous systems. Ongoing and future research is focused on developing analytic and applied methods, such as consensus- or market-based algorithms, for distributed multi-robot capabilities including swarm search strategies, split/join maneuvering [15], pursuit and evasion trajectory planning [16], multi-target assignment and tracking, as well as core behaviors for collision avoidance and distributed formation control. Additional topics for deeper exploration include advances in human-swarm interaction for managing and composing swarm behaviors, as well as in networking for efficient and robust routing of message traffic.

[1] Note that Sortie 5, that is, UAV09, is the aircraft that was manually landed after exhibiting anomalous flight behaviors (c.f. Figure 7), and no longer transmitting state information after being powered down. Hence, the column and row corresponding to Sortie 5 shows zero packet rates.

References

[1] R. Beard, D. Kingston, M. Quigley, D. Snyder, R. Christiansen, W. Johnson, T. McLain, and M. Goodrich, "Autonomous vehicle technologies for small fixed wing UAVs," *AIAA J. Aerospace, Comput., Information, Commun.*, vol. 2, no. 1, pp. 92–108, Jan. 2005.

[2] D. T. Cole, S. Sukkarieh, A. H. Goktogan, H. Stone, and R. Hardwick-Jones, "The development of a real-time modular architecture for the control of UAV teams," in *In Proceedings of the 5th International Conference on Field and Service Robotics*, Port Douglas, Queensland, Australia, 2005.

[3] P. Almeida, R. Bencatel, G. M. Gonçalves, and J. B. Sousa, "Multi-UAV Integration for Coordinated Missions," in *Encontro Científico de Robótica, Guimarães*, Apr. 2006.

[4] J. How, E. King, and Y. Kuwata, "Flight demonstrations of cooperative control for uav teams," in *In Proc. of AIAA 3rd Unmanned Unlimited Technical Conference, Workshop and Exhibit*, Chicago, IL, Sep. 2004.

[5] S. Bayraktar, G. Fainekos, and and G. Pappas, "Experimental cooperative control of fixed-wing unmanned aerial vehicles," in *in Proceedings of the 43rd IEEE Conference on Decision and Control*, Atlantis, Bahamas, Dec. 2004.

[6] V. Dobrokhodov, O. Yakimenko, K. Jones, I. Kaminer, E. Bourakov, I. Kitsios, and M. Lizarraga, "New Generation of Rapid Flight Test Prototyping System for Small Unmanned Air Vehicles," in *AIAA Modeling and Simulation Technologies Conference Proceedings*, 2007.

[7] S. Hauert, S. Leven, M. Varga, F. Ruini, and A. C. Et al., "Reynolds flocking in reality with fixed-wing robots: communication range vs. maximum turning rate," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, California, USA, 2011.

[8] A. Bhatia, M. Graziano, S. Karaman, R. Naldi, and E. Frazzoli, "Dubins trajectory tracking using commercial off-the-shelf autopilots," in *AIAA Guidance, Navigation, and Control Conference*, 2008.

[9] S. Park, J. Deyst, and J. P. How, "A New Nonlinear Guidance Logic for Trajectory Tracking," in *Proceedings of the AIAA Guidance, Navigation and Control Conference*, 2004, pp. 1–16.

[10] S. A. P. Quintero, G. E. Collins, and J. P. Hespanha, "Flocking with fixed-wing UAVs for distributed sensing: A stochastic optimal control approach," in *2013 American Control Conference*. IEEE, Jun. 2013, pp. 2025–2031.

[11] C. K. Peterson and J. Barton, "Virtual structure formations of cooperating UAVs using wind-compensation command generation and generalized velocity obstacles," in *2015 IEEE Aerospace Conference*. IEEE, Mar. 2015, pp. 1–7.

[12] I. Navarro and F. Matía, "A proposal of a set of metrics for collective movement of robots," *Proc. Workshop on Good Experimental Methodology in Robotics*, 2009.

[13] E. W. Frew and T. X. Brown, "Airborne communication networks for small unmanned aircraft systems," *Proceedings of the IEEE*, vol. 96, no. 12, 2008.

[14] O. Sahingoz, "Networking Models in Flying Ad-Hoc Networks (FANETs): Concepts and Challenges," *Journal of Intelligent & Robotic Systems*, vol. 74, no. 1-2, pp. 513–527, 2014.

[15] I. n. Navarro, "Exploring the Split and Join Capabilities of a Robotic Collective Movement Framework," *Int J Adv Robotic Sy*, vol. 10, no. 224, 2013.

[16] D.-I. You and D. H. Shim, "Design of an aerial combat guidance law using virtual pursuit point concept," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 229, no. 5, pp. 792–813, 2015.