Cognitive and Collaborative sUAS Swarms in Urban Environments

Maarten Uijt de Haag and **Svenja Huschbeck**
Technische Universität Berlin (TU Berlin)
Marchstrasse 12, 10587 Berlin
GERMANY

maarten.uijtdehaag@tu-berlin.de

ABSTRACT

This paper focusses on advanced absolute and relative navigation methods for swarms and swarm members for both open-sky and urban operations and discusses the development of the theory and algorithms of a novel absolute and relative PVA estimation approach based on the principles of cognition (i.e., appropriate and adapted actions based on perception and knowledge) and collaboration (i.e., improved ability to reason and interact based on information exchange and spatial distribution) between the swarm members. This paper outlines the underlying methodology, some simulation results, and the platform hardware components (three multi-copters and one ground station). It furthermore analyses and discusses the performance using sUAS flight test data collected in an open-sky and and simulations in an urban environment.

1.0 INTRODUCTION

Small Unmanned Aircraft Systems (sUAS) operations are increasing in demand and complexity. Using multiple cooperative sUAS (i.e., a swarm or group) can be beneficial and is sometimes necessary to perform certain tasks (e.g., infrastructure inspection, mapping, law enforcement, traffic monitoring) either independent or collaboratively. Using multiple dissimilarly equipped sUAS to perform a task cannot only significantly reduce the time required to complete a task but also reduce the risk of threats to population and properties due to the reduced complexity and weight, increased reliability, longer endurance of the smaller platforms [1], and, most importantly, the ability of the swarm to increase safety by exploiting the increase and reliability of knowledge in distributed cognition and swarm-wide collaboration.

Since many swarm applications take place even at very low altitude levels (VLL) and over urban, often populated, areas, it is important to address and limit the air and ground risks associated with their operation [2]. To lower the probability of an accident, and, thus, make the operation safer and more acceptable, the operation of the swarm members can be carefully planned, limited to pre-defined areas that are free of obstacles such as routes or geofenced zones, or achieved by equipping the swarm members with a conflict detection and resolution function that ensures that possible conflicts of a swarm member with other members or its environment are detected in time and alternative routes can be planned and executed. These strategies require reliable estimates of position, velocity, and attitude (PVA) of the swarm and its members both in the absolute sense (i.e., with respect to the geographic coordinate frame in which the routes and geofences are defined) and in the relative sense (i.e., with respect to other swarm members, other traffic, and objects in the environment).

In the example shown in Figure 1-1, each individual member must calculate its absolute position vector $\mathbf{r}_i^n$ while also assuring that the relative position or separation vectors $\mathbf{s}_{ik}^n$ between it and every other member are as accurate as possible. Here 'n' refers to the navigation frame in which all absolute coordinates and vectors are expressed (e.g. lat-lon-height, Earth Centered Earth Fixed or East North Up). Note that this coordinate frame must be the same for all sUAS. In many scenarios these separation vectors are even more important than the global positions of the members, to avoid collisions or perform complex cooperative tasks.

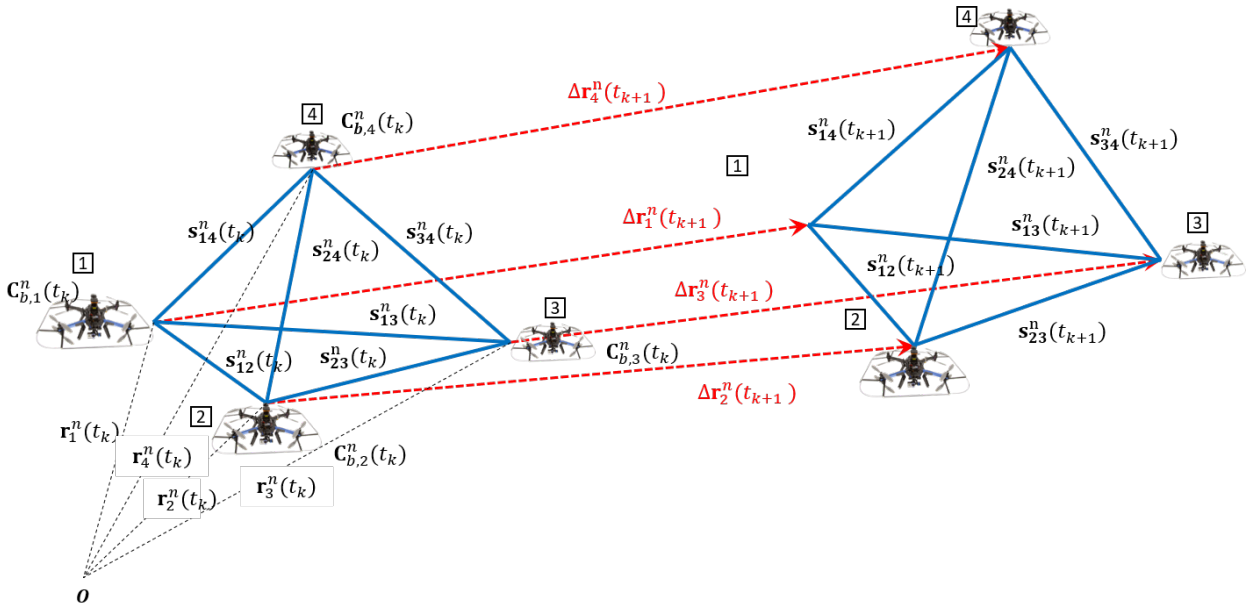

Figure 1-1: Swarm geometry of four sUAS as a function of time.

For many commercial (s)UAS, Global Navigation Satellite Systems (GNSS) have become one of the most dependable sources for absolute on-board position and velocity information enabling very stable position control of the UAS operating in autonomous or semi-autonomous modes. GNSS has been used as the positioning source for swarm operations in various research flight tests, for example, to evaluate methods to mimic the behaviour of a flock of birds with UAS [3], and the development of centralized [4] or ad-hoc communication networks [5][6] to relay information between the various members of the swarm. Even the large, impressive lightshows that use hundreds of UAS by companies such as Intel [7] and Ehang [8] are heavily choreographed and mainly use GNSS or real-time kinematic GNSS. These systems are typically operated in open areas where GNSS reception is optimal and, can, support the navigation performance (i.e. accuracy, integrity, availability and continuity) required for the operation.

Relative GNSS-based navigation methods, such as the one presented in previous work by the authors [9] work well for this task, given that all members have a GNSS capability. Because most errors are the same for receivers in proximity, the separation vectors can be easily calculated and remain accurate at the decimetre level. However, in the event of an GNSS outage or a compromised or malfunctioning receiver, the ability to calculate accurate separation vectors significantly degrades.

The reliance on GNSS forms a big challenge for operation of cooperative swarms in GNSS-denied or semi-denied environments such as, urban canyon environments for bridge or construction inspection, under-the-canopy operation during environmental monitoring or inventory tasks, city traffic surveillance, etc. If a swarm seeks to function in such environments, members must be able to perform relative (and absolute) navigation when some or all members are GNSS-denied. To enable operation of the swarm the PVA must be robust and not solely dependent on GNSS. To improve availability and guarantee continuity of service in GNSS-challenged environments, GNSS can be integrated with an IMU [10] or improved by increasing its sensitivity by using external data sources (i.e., assisted GPS). An alternative strategy is the integration or fusion of multiple sources of data may not only improve the accuracy of the position and attitude estimate, but also add integrity, continuity, and availability to the solution. Alternative navigation technologies may include (a) the integration of inertial sensors with imagery and laser scanners [11], (b) beacon-based navigation (i.e., pseudolites, UWB), (c) or navigation using signals of opportunity [12].

2.0 COOPERATIVE AND COGNITIVE SWARMING

In nature, swarms of animals have proven to be capable of solving a variety of different and complex tasks using their cognitive and collaborative abilities (e.g., honeybees, ants, birds [13][14]). Following the success of these swarms, we identify ***cognition*** (i.e., appropriate and adapted action based on perception and knowledge) and ***collaboration*** (i.e., improved ability to reason and interact based on information exchange and spatial distribution of a swarm) as the central elements of a swarm navigation architecture and use existing knowledge from neuroscience, biology and robotics to design methods to achieve the swarm's mission while meeting the stringent navigation requirements for urban operation. As an example of how cognition and collaboration is helping navigation in nature, [15] describes five mechanisms a swarm of animals may use to improve their navigation abilities and accuracy during migratory behaviour:

A *many wrongs*: improvement of the navigation performance by averaging or filtering the navigation estimates of the individual members. This principle is illustrated in Figure 2-1(a). The 5 sUAS are shown there with their uncertainty ellipse. When combining the noisy position and velocity estimates to obtain a global estimate of the swarm's motion, a noise reduction can be expected due to the averaging effect.

B *leadership*: some swarm members have better knowledge of their navigation solution and use that knowledge to help the remaining members meet their navigation performance requirements. An example of this mechanism is illustrated in Figure 2-1(b). Here two high flying sUAS are equipped with high-accuracy and resilient GNSS/Inertial systems where the other three swarm members (3,4, and 5) are not and are operating in an environment where GNSS performance is significantly deteriorated resulting in an inability of sUAS 3,4, and 5 to determine its absolute position accurately or at all. Now, sUAS 3,4, and 5 may be able to use the advanced knowledge of the leaders to improve their absolute and global position estimate. An example of this mechanism is discussed in [16].

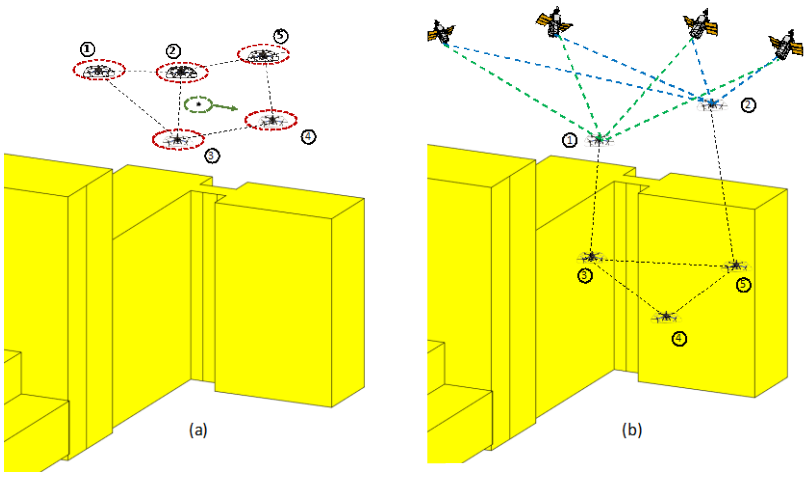

Figure 2-1: (a) many wrongs principle; (b) leadership.

C *emergent sensing*: the whole swarm comes up with a set of measurements that can be used to build a model of the environment (e.g., situational awareness) and help in the current or future navigation efforts. This mechanism is illustrated in Figure 2-2. In this example, swarm members with different sensor equipment perform an infrastructure inspection mission for the

first time and use the different sensor modalities to generate a map of the environment, that then can be used for navigation purposes in a manner similar to simultaneous localization and mapping (SLAM). Over time, this model will not only help the swarm perform its infrastructure inspection mission more efficiently, but also come up with a mission path that guarantees that the required navigation performance criteria are met and the mission is performed safely.

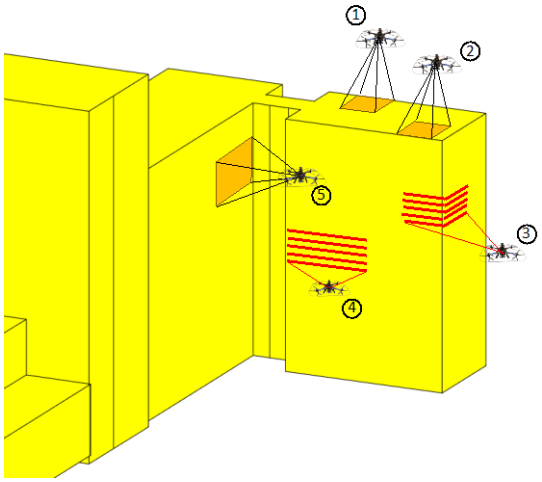

Figure 2-2: Emergent sensing.

D *social learning:* information existing with the individual swarm members is exchanged so the whole group can benefit. This mechanism allows new swarm sUAS to learn from the members that have been used for missions for a while. Furthermore, equally experienced sUAS can exchange information from past missions when their experiences have been different due to, for example, different routes/trajectories that they have flown. Information could include (partial) maps of the environment or expected navigation performance at certain locations as e.g. under bridges or urban canyons.

E *collective learning:* where interactions within the group lead to better and more detailed knowledge of the environment and, therefore, so better collision avoidance decisions can be made and routes can be identified that support the required navigation performance. An example would be to fly a configuration that optimizes map-building while at the same time using that map to maintain a good estimate of the relative location of the swarm members (in addition to GNSS).

As mentioned in the introduction, a swarm of UAS may significantly reduce the time required to complete the application task. If one distributes the sensors necessary to perform a certain task across the UAS within the swarm, it will be possible to use smaller and lighter UAS that operate longer and cost less, thus, offsetting the costs associated with the increase in platforms required. More importantly, it is our thesis that, due to the possibility to spatially distribute the swarm members, the swarm sensors combined can observe a larger part of the environment when working together, while at the same time use different perspectives to estimate for e.g. building features. In nature, this mechanism is referred to as emergent sensing and collective learning and may lead to increased accuracy in animal swarms. The result will be a better model of the environment that can be used to assess the collision risk by detecting possible loss-of-separation events of

the swarm members for the near future. If so, appropriate action can be taken to avoid this collision. Furthermore, the navigation capability may also deteriorate for one of the swarm members if its sensors does not provide sufficient information to estimate both the absolute and relative PVA state. Collaboration through the exchange of information, may alleviate this problem.

3.0 METHODOLOGY

3.1 Perception, Comprehension and Projection and Decision Making

The basic block diagram of the proposed method for cognitive and collaborative navigation of swarms is shown in Figure 3-1. This block diagrams resembles the model of situational awareness in dynamic decision system as introduced by Endsley [17]. Even though Endsley's work focusses on Human Factors, the concept of situational awareness of a human can be easily translated to a situation awareness model for the swarm members and the swarm as a whole. The decision-making loop shown in Figure 3-1 also shows a lot of similarities with the Observe, Orient, Decide and Act (OODA) loop by Boyd and the perception-action cycle [18]. A more detailed comparison of these approaches is outside the scope of this paper.

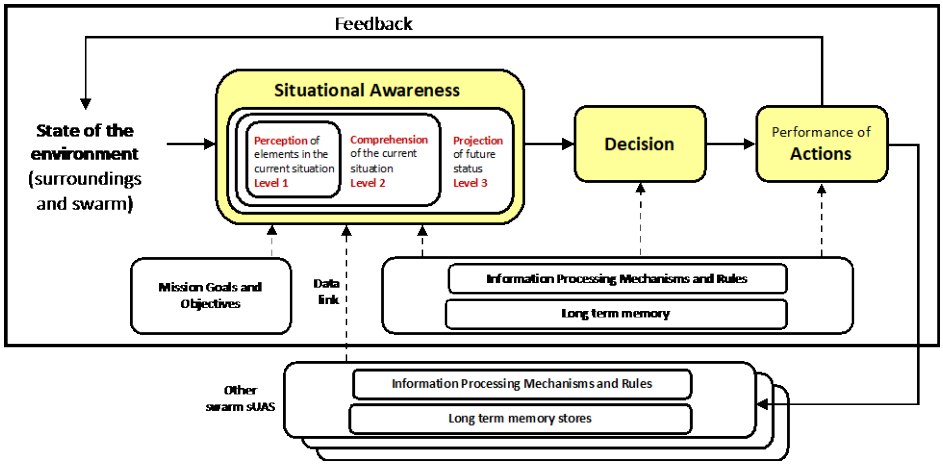

Figure 3-1: Cognitive and Collaborative Processes.

In the decision-making loop, the swarm and its members are improving their awareness of their surroundings not only to achieve their mission goals and objectives, but also to do so safely by meeting the required (stringent) absolute and relative navigation performance requirements. Like in human factors, three levels of situational awareness exist. In level 1, perception, the swarm member assesses what relevant information is available from the onboard navigation sensors and stored in the onboard long-term memory (e.g., maps, etc.). Furthermore, the swarm member also assesses what kind of information may be available in the swarm (memories and sensors) and can be requested if not already broadcast by the other members. The assessment results are then used by the comprehension module to get a clear understanding of the current situation, and, finally, to predict what the future situation would be. In terms of navigation performance this could refer to the current absolute and relative navigation performance and the future navigation performance, respectively. Based on this knowledge and the mission goals and objectives, the swarm members will make decisions regarding their planned actions (e.g., motion changes, new trajectories, new swarm configurations). Table 3-1 shows some of the navigation-related sensors that may be available on the swarm members. Note that the swarm may not necessarily have to consist of identical sUAS. This may lower the costs and weight of the platforms.

Table 3-1: Sensor information: measurement examples.

	Raw	Processed
GNSS	Pseudorange (ρ_i), carrier-phase (φ_i)	Position ($\mathbf{r}_i$), position change ($\Delta\mathbf{r}_i$)
Laser range scanners	Range (ρ_i), scan angle (α_i), point cloud	Position change ($\Delta\mathbf{r}_i$), orientation change ($\Delta\mathbf{C}_i$)
3D Imagers	Range (ρ_i), azimuth (α_i), elevation (β_i), point cloud	Position change ($\Delta\mathbf{r}_i$), orientation change ($\Delta\mathbf{C}_i$)
Camera (mono)	Unit vector pointing to feature/pixel ($\mathbf{e}_i$) + intensity	Scaled position change ($\lambda\Delta\mathbf{r}_i$), orientation change ($\Delta\mathbf{C}_i$)
Camera (stereo)	Pairs of unit vector pointing to feature/pixel ($\mathbf{e}_i, \mathbf{e}_j$)	Position change ($\Delta\mathbf{r}_i$), orientation change ($\Delta\mathbf{C}_i$)
Beacons	Range (ρ_i)	Position ($\mathbf{r}_i$)
Range radios (e.g., UWB)	Relative range between '*i*' and '*j*' (ρ_{ij})	-
IMU	Acceleration/specific force ($\mathbf{f}_i$), angular rate ($\boldsymbol{\omega}_{nb}^{b}$)	Position ($\mathbf{r}_i$), velocity ($\mathbf{v}_i$), attitude ($\boldsymbol{\psi}_i$)
Optical flow	-	Scaled velocity ($\lambda\mathbf{v}_i$),
Radio/laser altimeter	Height above ground (h_{AGL})	-
Baro altimeter	Height w.r.t. pressure reference height (h_{baro})	-
Magnetometer	Orientation w.r.t. magnetic field ($\mathbf{m}_i$)	-

An example of this approach is illustrated in the simple two-dimensional 3-member swarm example given in Figure 3-2. Based on the information available in each of the UAS in Figure 3-2 left only the 2D position of UAS 1 can be calculated, and a position estimate of UAS 2 and 3 is not possible. However, by exchanging information ($\mathcal{I}$) in the form of measurements between the swarm members, enough information will be available in each UAS to determine all three positions as shown in Figure 3-2 on the right. In this example, ρ_{ij} is the range between UAS '*i*' and '*j*' from a range radio sensor, and d_{ij} is the shortest distance between UAS '*i*' and surface '*j*' from a laser range scanner.

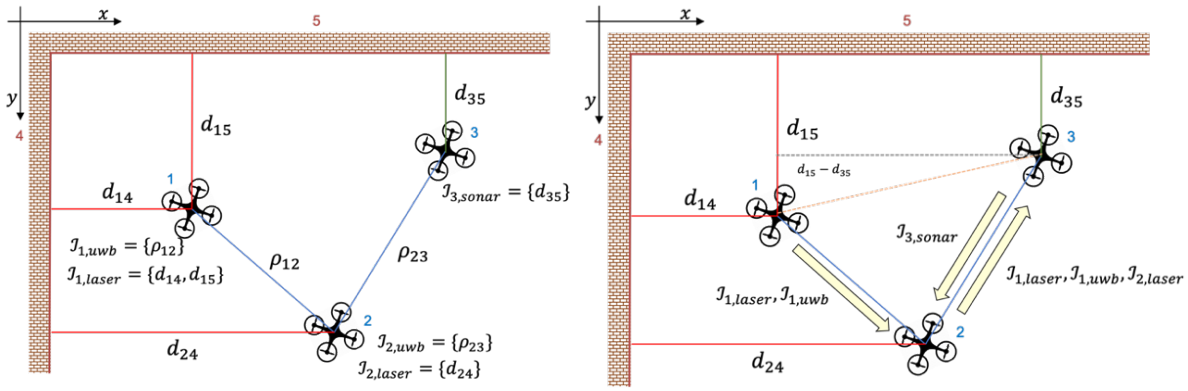

Figure 3-2: 2D swarm navigation without (left) and with (right) information exchange.

If, in addition to the information exchange, we also control the actions of the UAS to improve the position estimate, the movement of UAS 3 in Figure 3-3 within radio range of the ultra-wideband (UWB) range radio of UAS 1 results in two ways to estimate the UAS position, allowing the user to perform a consistency check and possibly detect sensor faults that may exist and lead to hazardous situations.

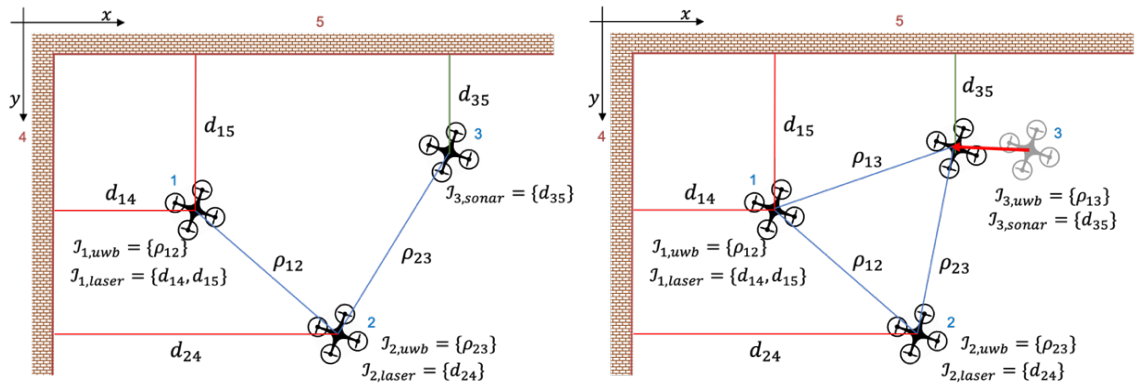

Figure 3-3: 2D swarm action reasoning without (left) and with (right) information exchange.

3.2 Swarm Constraints

When selecting the (minimum) set of sensor measurements required to estimate the position of each swarm member, it is important to identify the constraints introduced by the available measurements. The measurements must be selected such that the defined state vector determined by PVA, can be observed [19]. In the following examples, it is assumed that the whole swarm configuration is computed by each swarm member's processing unit. Note that the choice between central and decentralized swarm architecture is not the focus of this paper.

Figure 3-4 shows a 2D example of a small swarm with 3 sUAS where the ranges between all members are known due to the availability of range radio measurements. The range constraints introduced by these measurements fix the relative position of the sUAS within the swarm but do not constraint the swarm's orientation and location in 2D space. When one member of the swarm, has an absolute position capability, the swarm is anchored to this location with associated position uncertainty, but can still rotate freely about this point in 2D (Figure 3-4 left).

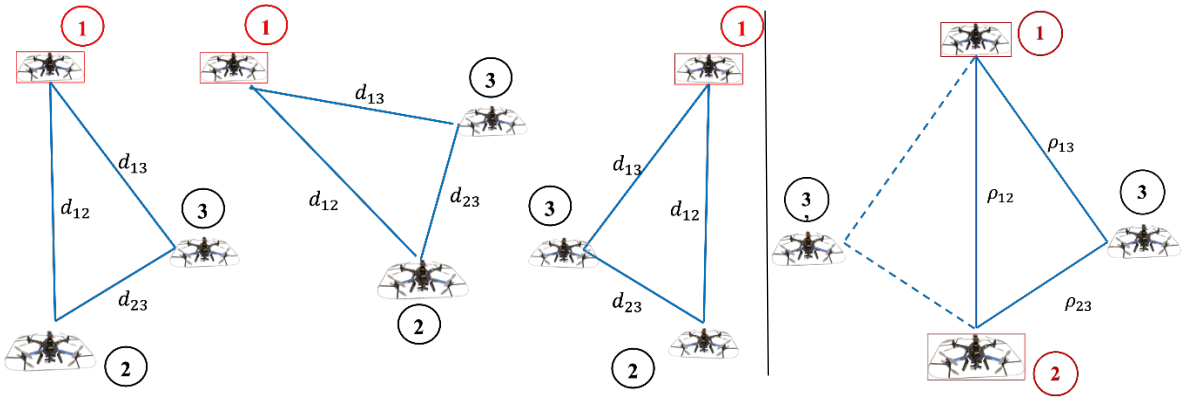

Figure 3-4: 2D absolute position constraints when all ranges between the swarm members are known; (left) single global constraint; (right) dual global constraints.

When two of the members (1 and 2) have knowledge of their global position estimate, but member 3 does not, the ambiguity is reduced to 1-degree of freedom as shown in Figure 3-4 right. With known ranges ρ_{12}, ρ_{13}, and ρ_{23}, member 3's position is ambiguous: it could either be in at position 3 or position 3' in 2D. In three dimensions (3D) the location of sUAS 3 could be on a circle. In 3D the ambiguity could again be

reduced to two degrees (horizontal translation, rotation around a vertical axis) when the altitude is known from baro-altimeters, for example, as illustrated in Figure 3-5.

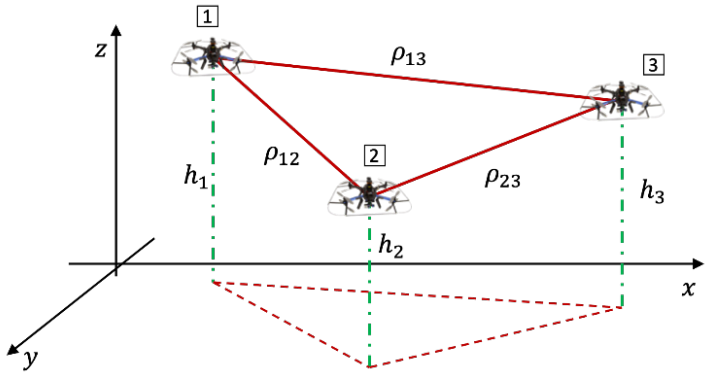

Figure 3-5: 3D swarm configuration with range and altitude constraints.

3.3 Filter Mechanizations

To integrate the information from a swarm member's onboard sensors and data received from other swarm members, various integration approaches may be selected including snapshot methods such as ordinary least squares (OLS) and weighted least squares (WLS) estimators, or sequential estimators such as Kalman Filters (KF), Extended Kalman filters (EKF), or Particle Filters (PF). For details on all these estimators many good reference texts are available including [19] and [20]. Alternatively, one can also utilize a batch of data (across sensors and across time) and obtain a maximum likelihood estimate using non-linear least squares solver tools such as g2o [21], Ceres [22] and the various tools available in Matlab. These non-linear least squares solvers (NLS-Solver) have been used extensively in Simultaneous Localization and Mapping (SLAM) methods such as GraphSLAM [23].

The filter mechanizations must be selected based on available sensor information and care must be taken that potential transitions from one mechanization to another do not affect the navigation performance adversely (e.g., jumps). For each of the sensors available within the swarm, a basic measurement model, $\mathbf{z} = \mathbf{h}(\mathbf{x}) + \boldsymbol{v}$, must be derived that relates the sensor measurements to the state vector, $\mathbf{x}$ (i.e., navigation state or error state) through function $\mathbf{h}$. Component $\boldsymbol{v}$ represents the error introduced by the sensor. In many cases, this error is assumed to be a zero-mean normally distributed noise with a covariance matrix equal to $\boldsymbol{\Sigma}_{\boldsymbol{v}}$. However, this assumption must always be tested before designing the filter mechanization. In the following sections we will shortly describe the mechanizations that were used in the results section. Note that, for integrity purposes, fault models should be considered as well. Due to the extensive nature of these models, that discussed will be the topic of a future publication.

3.4 Case I: Inertial/Range-radio/Baro Integration

Case I is based on an actual flight test with multiple sUAS whose results are shown in Section 4.0. In Case I the swarm consists of 6 sUAS equipped with range radios, low cost IMUs, and altimeters. Furthermore, a subset of sUAS has been equipped with a GNSS receiver. This case was first introduced in [24]. The Case I scenario shows an example of the leadership and social learning principles mentioned in Section 2.0. It is assumed here that all information is shared among the sUAS. Of course, in a real scenario this would be too much of a burden on the data links of the swarm members, but for the results are shown for illustration purposes. To estimate the position of all sUAS, a centralized complementary extended Kalman filter (CEKF) has been used. The state vector consists of the errors states of all swarm members involved in the position calculation (here: 6). Error states include 3D position and velocity errors, the tilt error, and the accelerometer and gyro bias vectors. Expressions for these as well as their corresponding state transition

matrices can be found in reference texts such as [10] and [25]. For a swarm with 'N' INS systems, the state vector would be size $15N \times 1$.

The range between any two sUAS can be computed using their respective INS position estimates $\tilde{\mathbf{r}}_{k,ins}$, and expressed in terms of the true relative position and an additional error term that combines the INS errors from '*i*' and '*k*', or:

$$\rho_{ik_{ins}} = \left\| \tilde{\mathbf{r}}_{i,ins} - \tilde{\mathbf{r}}_{k,ins} \right\| = \left\| \mathbf{s}_{ik} - \delta\mathbf{r}_{ik,ins} \right\| \tag{1}$$

Equation (1) can be linearized with respect to the involved sUAS INS position estimates.

$$\tilde{\rho}_{ik_{ins}} \approx \rho_{ik_{true}} + \frac{\mathbf{s}_{ik}^T}{\rho_{ik}} \delta\mathbf{r}_{ik,ins} = \rho_{ik_{true}} + \underbrace{\mathbf{u}_{ik}^T \delta\mathbf{r}_{ik,ins}}_{\delta\rho_{ik_{ins}}} \tag{2}$$

The row elements of the measurement vector, $\mathbf{z}$, are given by the various available range differences in the swarm:

$$z_{ik} = \tilde{\rho}_{ik_{ins}} - \tilde{\rho}_{ik_{rr}} = \delta\rho_{ik_{ins}} + v_{rr} \tag{3}$$

where $\delta\rho_{ik_{ins}}$ is the error due to the contributions of two INSs involved, and v_{rr} is the range noise error of the range-radio. Equation (3) can be expressed in the inertial position error terms as follows:

$$z_{ik} = \delta\rho_{ik_{ins}} + v_{rr} \approx -\mathbf{u}_{ik}^T \delta\mathbf{r}_{ik,ins} + v_{rr} \tag{4}$$

The baro-altimeter measurement is much simpler, as it is already linear:

$$z_{alt,k} = h_{k,ins} - h_{k,alt} = \delta r_{k_{z,ins}} + v_{baro} \tag{5}$$

where $\delta r_{k_{z,ins}}$ is the z-component of the position vector computed by the INS of swarm member k'. The results of equations (4) and (5) can be used to setup the $\mathbf{H}$-matrix. A block diagram that summarizes this method is shown in Figure 3-6.

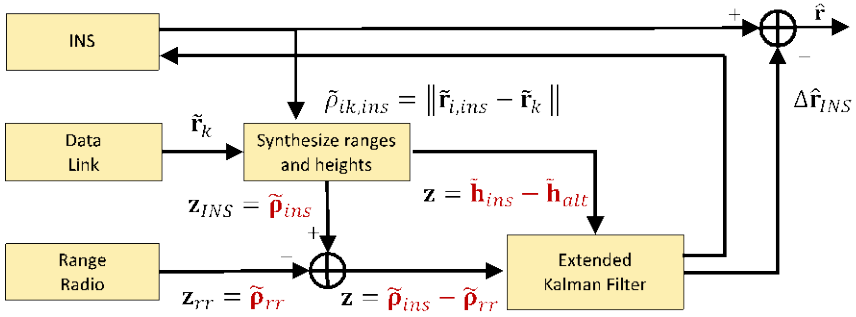

Figure 3-6: Integration of range radio, INS, altimeter, and leader sUAS GNSS positions.

For the sUAS that have valid and reliable GNSS available, the measurement vector $\mathbf{z}$ will be augmented with the difference between its INS and GNSS position and velocity.

3.5 Case II and III: Non-linear solvers and Inertial/Range-radio/Beacon Integration

Cases II and III simulate a swarm of 6 and 8 sUAS, respectively. Both the Case II and II swarms operate in an urban environment at low altitudes. The equipment list for all swarm members is provided in Table 3-2. In Case II, a network of local beacons is installed and two of the six sUAS (i.e., leaders) are equipped with receivers that are capable of making range measurements, ρ_{i,b_k}, to these beacons $\{b_k | k = 1, \dots, M\}$. In Case III, this beacon system is not available but two high flying and moving sUAS are equipped with high-end GNSS/INS and range-radios. Regarding communications, the complete swarm is fully connected.

Table 3-2: sUAS Equipment List

	Equipment	1	2	3	4	5	6	7	8
Case II	IMU	√	√	√	√	√	√	n/a	n/a
	Baro	√	√	√	√	√	√	n/a	n/a
	Beacon	√	-	-	√	-	-	n/a	n/a
	Range-radio	√	√	√	√	√	√	n/a	n/a
	GNSS	-	-	-	-	-	-	n/a	n/a
Case III	IMU	√	√	√	√	√	√	√	√
	Baro	√	√	√	√	√	√	√	√
	Beacon	-	-	-	-	-	-	-	-
	Range-radio	√	√	√	√	√	√	√	√
	GNSS	-	-	-	-	-	-	√	√

In terms of filters, the swarm of fleet has multiple variants available. For the sUAS equipped with the beacon system, these filters include a standard OLS uses the range measurements to the beacons and baro-altitudes and a loosely coupled linear Complementary Kalman Filter (CKF) that integrates the INS output with the position output of the beacon/baro OLS. In addition, the filter sets also included a tightly integrated beacons/baro/inertial mechanization but that one is not included in the Case II result section.

The other Case II sUAS must rely on their INS, baro, range-radio and information received from the beacon-system equipped leaders. For their position solution, they depend on a non-linear least squares solver like the ones mentioned earlier. In this case, the position is obtained by minimizing a cost function based on the sum of log-likelihoods of the various measurements. The likelihood and the log-likelihood functions are related to the measurement equation and can be expressed as follows:

$$p(\mathbf{z}|\mathbf{x}) = \eta \, exp\{[\mathbf{z} - \mathbf{h}(\mathbf{x})]^T \boldsymbol{\Sigma}_v^{-1} [\mathbf{z} - \mathbf{h}(\mathbf{x})]\} \tag{6}$$

$$\begin{aligned} &\log\{p(\mathbf{z}|\mathbf{x})\} = \log(\eta) + [\mathbf{z} - \mathbf{h}(\mathbf{x})]^T \boldsymbol{\Sigma}_v^{-1} [\mathbf{z} - \mathbf{h}(\mathbf{x})] \Rightarrow \\ &\log\{p(\mathbf{z}|\mathbf{x})\} \propto [\mathbf{z} - \mathbf{h}(\mathbf{x})]^T \boldsymbol{\Sigma}_v^{-1} [\mathbf{z} - \mathbf{h}(\mathbf{x})] \end{aligned} \tag{7}$$

where η is the probability density function's normalizing constant and $\mathbf{z} - \mathbf{h}(\mathbf{x})$ is the residual vector. For measurement set $\mathcal{S} = \{\mathbf{z}_i | i = 1, \dots, N\}$ of independent measurements, one can obtain an estimate for state vector $\mathbf{x}$ by minimizing the log likelihood sum of all observations:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \mathbf{F}(\mathbf{x}) \tag{8}$$

where:

$$\mathbf{F}(\mathbf{x}) = \sum_{\mathbf{z}_i \in \mathcal{S}} [\mathbf{z}_i - \mathbf{h}_i(\mathbf{x})]^T \boldsymbol{\Sigma}_{v_i}^{-1} [\mathbf{z}_i - \mathbf{h}_i(\mathbf{x})] \tag{9}$$

In our case the various residuals are:

$$F_{rr,1}(\mathbf{x}) = \frac{1}{\sigma_v}\left[\|\mathbf{x} - \mathbf{r}_{lead,1}\| - \tilde{\rho}_{i,1}\right]$$
$$F_{rr,4}(\mathbf{x}) = \frac{1}{\sigma_v}\left[\|\mathbf{x} - \mathbf{r}_{lead,4}\| - \tilde{\rho}_{i,4}\right] \quad (10)$$
$$F_{h,1}(\mathbf{x}) = \frac{1}{\sigma_{alt}}\left[z - h_{alt}\right]$$

Note that the residuals are weighted by the inverse of the standard deviation to obtain residuals that are unit-variance. Furthermore, σ_v must include both the effects of range radio errors and the errors due to uncertainty in the leader positions. Now, this assumes a Normal error probability distribution which is an incorrect assumption and should be corrected in future implementations. Another improvement would be the inclusion of the inertial measurements using, for example, IMU factor such as proposed for GraphSLAM in [26][27].

In this simple case, this optimization method could be replaced by a carefully formulated OLS, WLS or EKF as well. The NLS-solver was just used here so it can be easily extended to more measurements, like is the case in Graph-based SLAM methods.

In Case III, the beacon system is omitted, but two GNSS/INS equipped higher-flying sUAS are included. In terms of algorithms, all sUAS except for the leaders can use (i) an OLS (or WLS) using the range radio ranges to the leaders and the leader's position estimates, (ii) or the LNS-solver method. Again, the leader's position uncertainty must be considered.

4.0 CASE I: FLIGHT TEST RESULTS

To evaluate the swarm's absolute and relative positioning capability using the method explained in Section 3.3, data from real sUAS platforms was used (see Figure 4-1). The platforms were equipped with an Odroid XU4 processor (running Ubuntu and the Robotics Operating System) to perform the data collection, with multiple inertial units of varying costs and qualities and different GNSS receivers: Platform A with a Sensonor STIM300 and a Novatel OEM-615; Platform B with a VectorNav VN-100 and Xsens Mti-1 inertial and a Novatel OEM-615 GNSS receiver, and Platform C with an Xsens Mti-1 and a U-blox M8T. In addition, the platforms were equipped with a point-to-multipoint data radio for platform-to-platform and platform-to-ground station communication. The latter was just present for monitoring purposes.

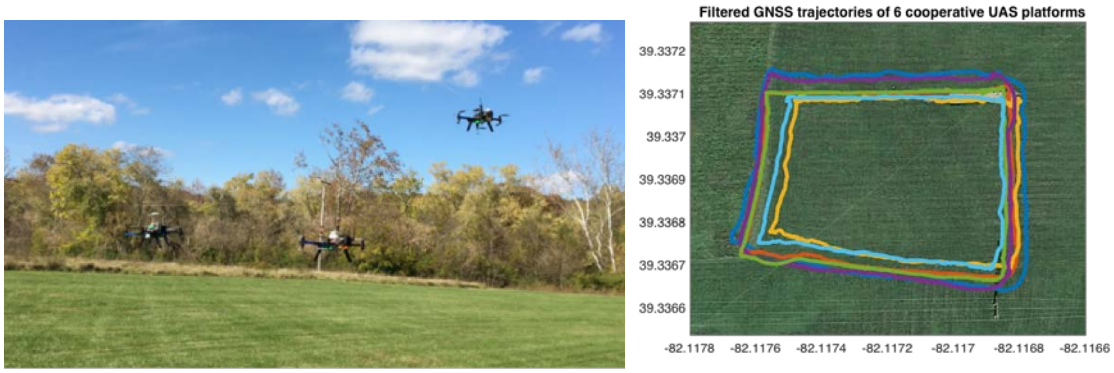

Figure 4-1: Formation flight at test field at Ohio University and overlaid flight trajectories.

During two flights, sensor data was collected and time-tagged. The two flight-data set times were adjusted

and then overlaid in post-processing to generate a single data set equivalent to a six-member flight. The 6 overlaid flight trajectories are shown in Figure 4-1 on the right side. Range radio measurements were simulated from the GNSS position solutions and noise was added based on the ranging performance of the range radios developed by Ohio University and presented earlier in [24].

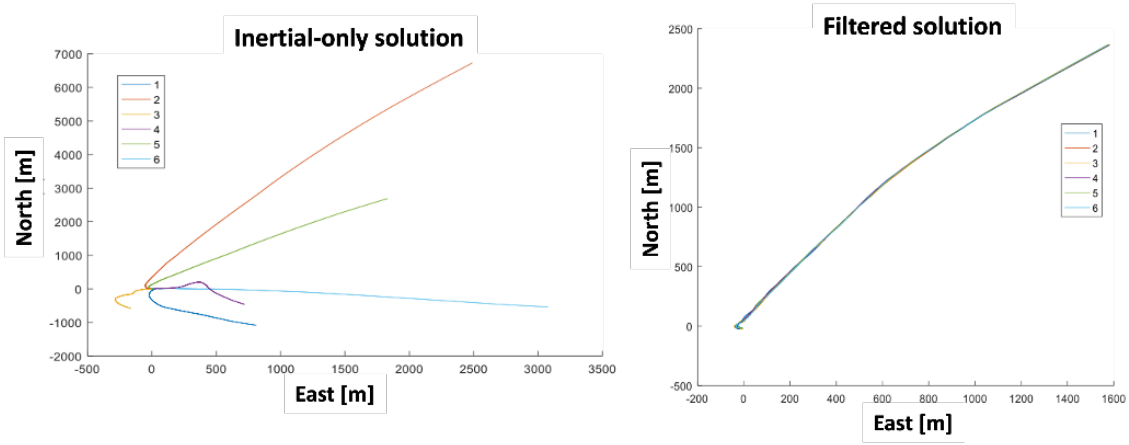

Figure 4-2: (left) positioning estimates based on inertial only; (right) inertial/range radio integration.

The results for the inertial-only and inertial-range-radio integration mechanizations are show in Figure 4-2. Whereas the scenario in which each sUAS determines its position based on its own inertial and baro-measurements shows positions estimates that drift apart over time, the filter results show that the position estimate still drifts but that now the swarm drifts due to a significant performance improvement relative navigation performance. This is even more evident from the error plots shown in Figure 4-3 where the relative navigation error plots are shown using log(y) axes to better capture both the inertial-only and the filtered results.

In case, one or more members are equipped with GNSS equipment and in an area where GNSS would provide fault free performance (as established by integrity-monitoring functions) or, in case of interference or denial-of-service, have resilient GNSS receivers, their knowledge of the global position can be used by the swarm as a whole. Figures 4-4 and 4-5 show the effect on the estimated trajectories when one, two or three of the six swarm members can determine their “global” position.

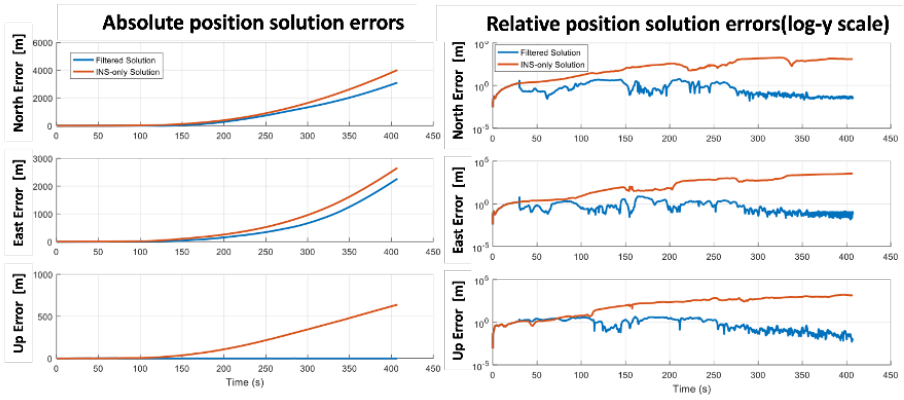

Figure 4-3: Formation flight at test field at Ohio University and overlaid flight trajectories.

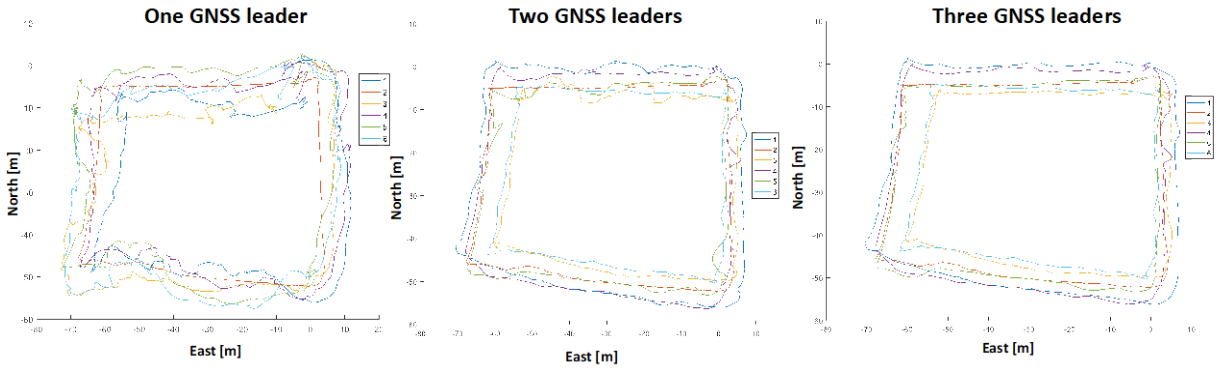

Figure 4-4: Filter results (trajectories) when one or more leaders have knowledge of the global (absolute) position due to their advanced equipment or advantageous location.

With a single GNSS-aided member in the swarm (Octocopter 2), every member's solution becomes more accurate, both in an absolute and relative sense. The swarm is globally tied down by the GNSS-aided member and will rotate around it as the members' INSs drift while keeping relative positions intact. When the filter settles after about 30 seconds, the solution with two GNSS-aided members (Octocopters 2 and 6) becomes even more accurate than with one GNSS-enabled member – the absolute error drops to near-zero from a ±20m fluctuation in the one-GNSS-enabled configuration. At this point, all three dimensions are tied down – x and y from the two GNSS-aided members, and z from the baro-altimeter.

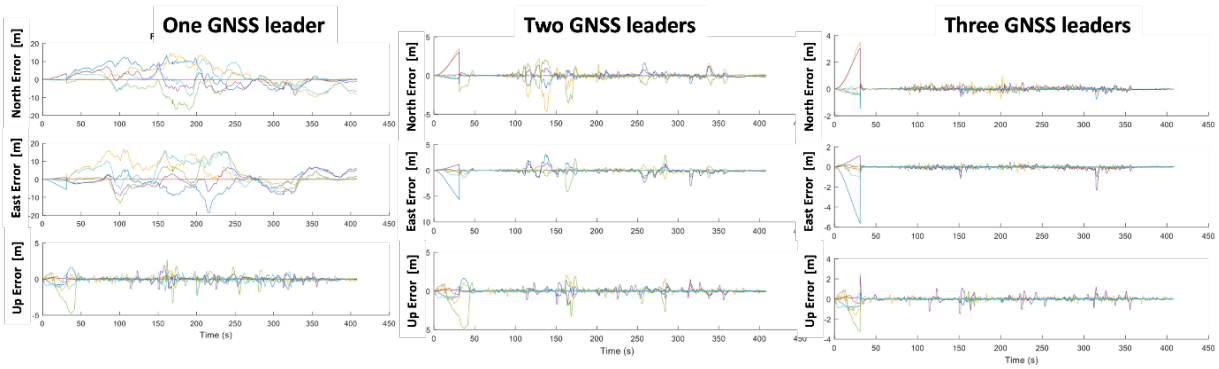

Figure 4-5: Filter results (errors) when one or more leaders have knowledge of the global (absolute) position due to their advanced equipment or advantageous location.

With three points of constraint, the solution should improve, but diminishing returns are expected as the solution becomes is over-constrained

5.0 CASE II: SIMULATION RESULTS

To evaluate the various Cognition and Collaboration methods, a simulation environment was developed in Matlab. This simulation includes 3D models of the urban environment based on CityGML LOD2 files for Berlin. In past work, this model has been used to perform extensive urban Dilution of Precision analyses for urban GNSS performance assessments. The left side of Figure 5-1 shows one of the output screens of this environment. This figure is a snapshot as, typically, this figure will show an animation of the trajectories followed by the swarm. For the example provided here, the 6-sUAS trajectories from Case I were geographically moved from their benign environment in Ohio to the urban environment in Berlin. The

advantage is that it allows for partial reuse of the actual sensor data with their actual errors. For example, in shown case the real inertial data (accelerometers and gyro outputs), baro-altitudes and GPS data was "translated" to Berlin. For the GPS data, this means expressing all reference data (i.e., satellite positions) in a local frame. Of course, this will then no longer represent an actual GPS satellite path.

Figure 5-1 also shows the filter performance on the right side for UAS 1, 3 and 4, were sUAS 1 and 4 are the leaders equipped with beacon system receivers. The beacon system measurements were given a noise error of 0.7 1-σ. One would expect some multipath errors as well, but these were not modelled in this simulation. As can be seen in Figure 5-1, both the OLS (UAS #4) and the integrated beacon/inertial, show good results with an expected noise reduction in the latter.

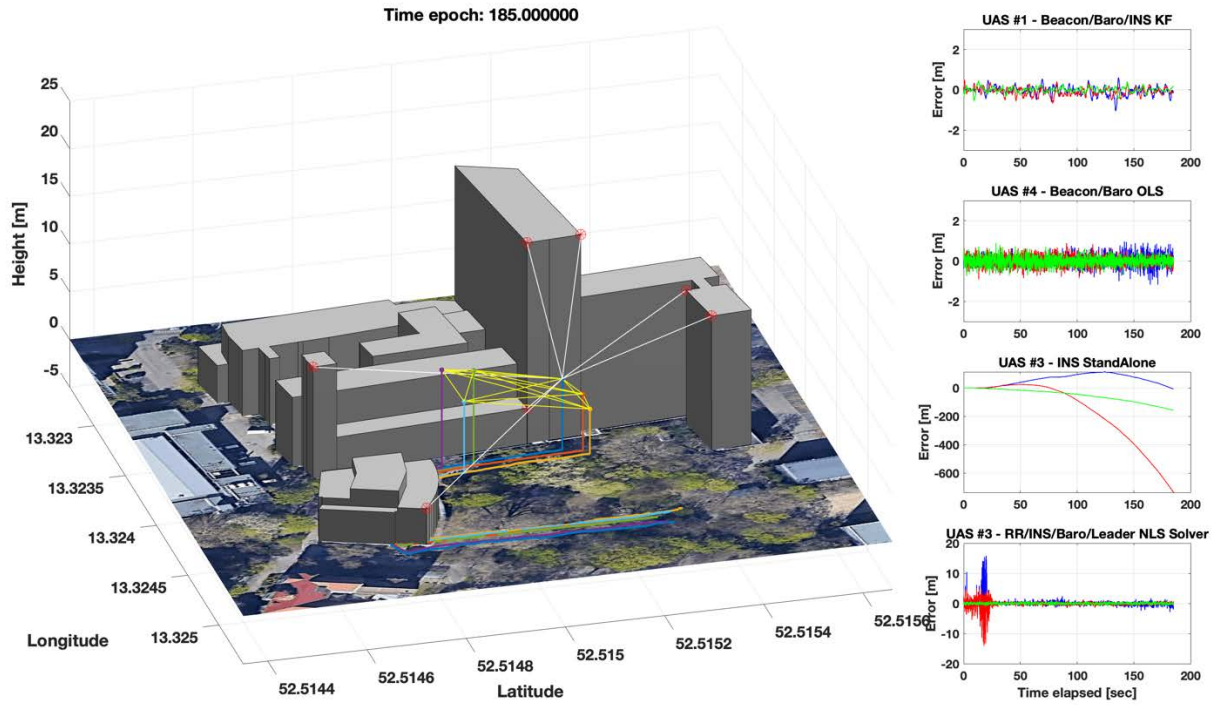

Figure 5-1: Case II: Swarm of 6 sUAS performing a collaborative mission in an urban environment using a locally installed beacon system.

The NLS-Solver plot for UAS #3 shows good results as well. The large uncertainty at the beginning can be explained by a bad geometry formed by the two leaders and UAS #3. With some of the others that is not the case. It would therefore have to be part of the *comprehension* and *projection* module to predict that this will be the case, and to define appropriate actions (trajectory changes) to avoid such a bad geometry. To evaluate that, the simulation tool will have to be adjusted as the trajectories can no longer be based on playback data.

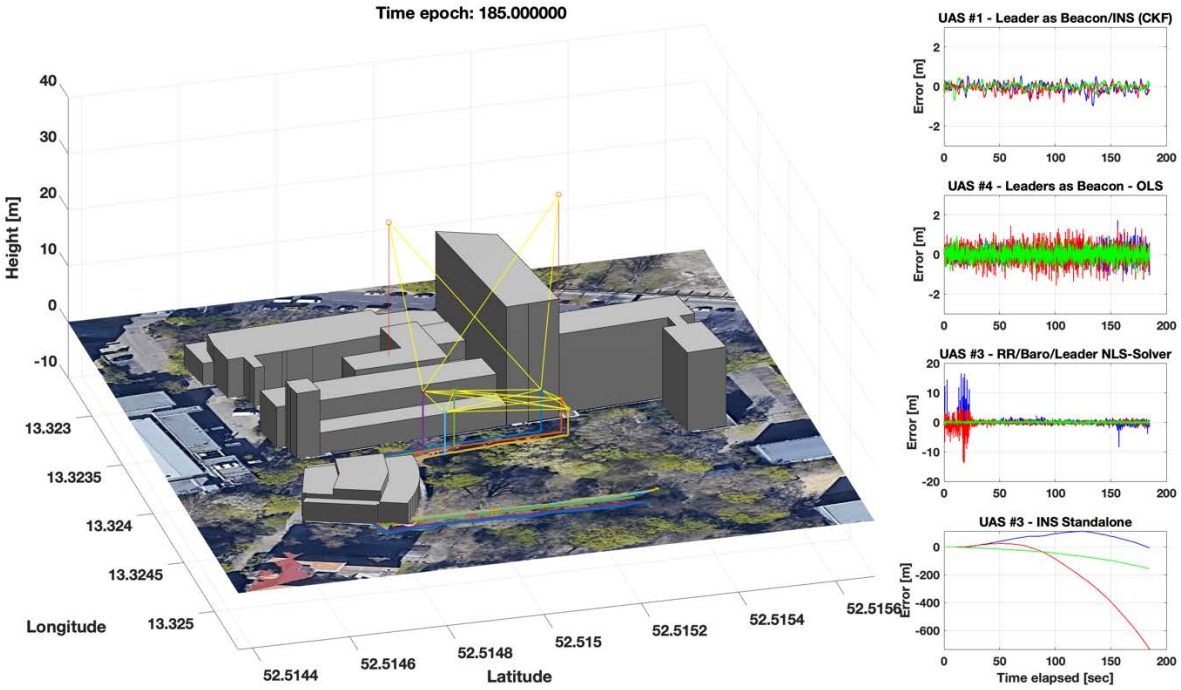

Figure 5-2: Case III: Swarm of 8 sUAS performing a collaborative mission in an urban environment using two leaders with high-performance navigation equipment at higher altitudes.

The results for the Case III scenario are shown in Figure 5-2. As expected, the OLS results are noisier due to the noise contributions of the leaders' position estimates. This is less visible in the KF results. Again, the NLS-Solver results of UAS#3 are quite similar including the larger error during the time that UAS#3 forms a bad geometry with UAS #1 and #4. The NLS-Solver results are still noisy, and it is expected that the inclusion of IMU measurements as well as optimizing using a time-window of data (like in GraphSLAM) will smooth the results. Alternatively, a EKF implementation may smooth the trajectories. Of course, no accuracy (and other navigation performance) targets were defined for this paper, so it is hard to say if the result meet the targets levels of safety.

6.0 SUMMARY AND CONCLUSIONS

This paper introduced the initial foundations of a cognition and collaboration approach to absolute and relative navigation of swarms of sUAS based, in part, one some basic principles of swarm navigation in nature. Results from a flight test and a simulation demonstrated that the leadership and social learning principles do apply nicely and that by evaluating the necessary constraints filters can be defined that allow some of the swarm members to operate in GNSS-challenging environments.

So far, the methods have mainly addressed the accuracy performance of the navigation solution. The next steps will address the inclusion of other navigation performance parameters (i.e., integrity, continuity, and availability). Once those have been established, the situational awareness module can be completely implemented by including methods to predict what actions (i.e., short- and mid-term trajectory changes) of the swarm and its members will lead to a continued safe operation. To evaluate these new methods the simulation tool must be updated and improved flight test in a relevant operation prepared.

Finally, the proposed method must be aligned with the large volume of communication literature to minimize the utilized communication bandwidth, while maintaining safe operation.

7.0 REFERENCES

[1] Y. Tan, Z. Zheng, "Research Advance in Swarm Robotics," in *Defence Technology*, ISSN: 2214-9147, Vol: 9, Issue: 1, 2013, pp. 18-39.

[2] JARUS, "JARUS Guidelines on Specific Operations Risk Assessment (SORA)," JAR-DEL-WG6-D.04, January 2019.

[3] G. Vásárhelyi, Cs. Virágh, G. Somorjai, T. Szörényi, T. Nepusz, A. E. Eiben, T. Vicsek, "Optimized flocking of autonomous drones in confined environments," *Sci. Robot.* 3, 2018.

[4] T. Indriyanto, et al., "Centralized swarming UAV using ROS for collaborative missions," *AIP Conference Proceedings* 2226, 030012 (2020); https://doi.org/10.1063/5.0002616, June 2020.

[5] A. Pospischil, "Mapping ad hoc communications network of a large number fixed-wing uav swarm," Thesis, Naval Postgraduate School, Monterey, California, 2017.

[6] M. Campion, P. Ranganathan, and S. Faruque "UAV swarm communication and control architectures: a review," *J. Unmanned Veh. Syst.*, Vol. 7, 2019.

[7] Intel, https://www.intel.com/content/www/us/en/technology-innovation/aerial-technology-light-show.html, Accessed on 12 August 2019.

[8] Ehang, https://www.ehang.com/news/249.html, Accessed on 12 August 2019.

[9] J. E. Huff and M. Uijt de Haag, "Assured relative and absolute navigation of a swarm of small UAS," in *Proceedings of the IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, St. Petersburg, FL, USA, 2017.

[10] Farrell, J. L., *GNSS Aided Navigation & Tracking – Inertially Augmented or Autonomous*, American Literary Press, 2007.

[11] M. M. Miller, M. Uijt de Haag, A. Soloviev, M. Veth, "Navigating in Difficult Environments: Alternatives to GPS – 1," Proceedings of the NATO RTO Lecture Series on "Low Cost Navigation Sensors and Integration Technology," SET-116, November 2008.

[12] M. M. Miller, J. Raquet, M. Uijt de Haag, "Navigating in Difficult Environments: Alternatives to GPS – 2," Proceedings of the NATO RTO Lecture Series on "Low Cost Navigation Sensors and Integration Technology," SET-116, November 2008.

[13] C. A. Freas, P. Schultheiss, "How to Navigate in Different Environments and Situations: Lessons from Ants," *Front. Psychol.*, 29 May 2018.

[14] R. Menzel, et al., "Honeybees navigate according to a map-like spatial memory," *Proc. Natl. Acad. Sci.* U.S.A., February 2005.

[15] A. M. Berdahl, et al., "Collective Animal Navigation and Migratory Culture: from Theoretical Models to Empirical Evidence," Phil. Trans. R. Soc. B37320170009, March 2018.

[16] F. Causa, "Planning Guidance and Navigation for Autonomous Distributed Aerospace Platforms," Ph.D. Dissertation, Università degli Studi di Napoli "Federico II", March 2020.

[17] M. R. Endsley, "Toward a Theory of Situation Awareness in Dynamic Systems," *Human Factors*, 1995, 37(1).

[18] S. Haykin, J. M. Fuster, "On Cognitive Dynamic Systems: Cognitive Neuroscience and Engineering Learning From Each Other," *Proceedings of the IEEE*, Vol. 102, No. 4, April 2014.

[19] P. S. Maybeck, *Stochastic Models, Estimation and Control – Vol. I*, Navtech Books and Software Store, 1994.

[20] B. Ristic, et al., *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House, 2004.

[21] R. Kuemmerle, et al., "g2o: A General Framework for Graph Optimization," Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2011.

[22] S. Agarwal, K. Mierle, et al., "Ceres Solver", http://ceres-solver.org.

[23] S. Thrun and M. Montemerlo, "The graph SLAM algorithm with applications to large-scale mapping of urban structures," *Int. Journal of Robotics Research,* 25(5-6):403, 2006.

[24] J. Huff, Absolute and Relative Navigation of an sUAS Swarm Using Integrated GNSS, Inertial and Range Radios, *M.S.E.E. Thesis*, Ohio University, 2018.

[25] P. D. Groves, Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Artech House, 2008.

[26] T. Lupton and S. Sukkarieh, "Visual-Inertial-Aided Navigation for High-Dynamic Motion in Built Environments Without Initial Conditions", TRO, 28(1):61-76, 2012.

[27] C. Forster, et al., "IMU Preintegration on Manifold for Efficient Visual-Inertial Maximum-a-Posteriori Estimation", *Robotics: Science and Systems (*RSS), 2015.