Investigating the survivability of drone swarms with flocking and swarming flight patterns using Virtual Reality

Daniel Biediger[1], Arun Mahadev[2], Aaron T. Becker[2].

Abstract— **It is now possible to deploy swarms of drones with populations in the thousands. There is growing interest in using such swarms for defense, and it has been natural to program them with bio-mimetic motion models such as flocking or swarming. However, these motion models evolved to survive against predators, not enemies with modern firearms. This paper presents experimental data that compares the survivability of several motion models for large numbers of drones. This project tests drone swarms in Virtual Reality (VR), because it is prohibitively expensive, technically complex, and potentially dangerous to fly a large swarm of drones in a testing environment. We model the behavior of drone swarms flying along parametric paths in both tight and scattered formations. We add random motion to the general motion plan to confound path prediction and targeting. We describe an implementation of these flight paths as game levels in a VR environment. We then allow players to shoot at the drones and evaluate the difference between flocking and swarming behavior on drone survivability.**

I. Introduction

Defense departments desire to use unmanned drones for detection and deterrence of hostile units. Enemy forces desire to destroy these drones. This inspires us to develop strategies that make use of the speed and agility of drones to survive in hostile environments. While single drones are useful for some applications [1], [2], swarms can be more survivable and pose a larger threat in a way that is difficult to counter [3]–[5]. As drones become smaller, lighter, and more cost effective, they become preferred in situations where sending human scouts and interdiction teams would be dangerous [6]–[8]. Because they are small, fast, and agile, they can avoid or mitigate dangerous situations. Because individual drones have limited endurance, carry weight, and computational power, swarms can provide increased redundancy, coverage, and mission capacity [9].

Robots are widely used in areas that are too dangerous for humans. Remotely piloted aircraft are deployed to watch over friends and search for enemies. There is growing interest in using drones to locate and deter hostile forces [10]. Where a single drone could be defeated, a swarm of drones offers strength in numbers. A person might swat a single bee, but would be reticent to disturb a whole swarm of bees.

In the same way that birds and bees group together differently, there is a difference between flocks of drones and swarms of drones. Both represent collective behaviors of a group, but the idea of a flock or school reflects a tight order and a similar general direction of motion for all of its members. In contrast, a swarm suggests a more distributed and chaotic motion, where members appear to circulate randomly. Against some types of adversaries, tight grouping and common motion direction can be catastrophic. Consider the way that fish school together to confuse predators. Because it is difficult for predators to identify and isolate a single target, the individual fish are safer as part of a tight group. Unfortunately, this defensive strategy fails completely against a carefully placed net – the entire school of fish can be trapped. In the same way, a tight formation of drones could be countered by a net, or defeated by modern air-burst projectiles.

II. Previous Work

While drones are becoming popular for a number of applications, the use of swarms or flocks of drones are still relatively new. In part, this is due to the complexity of managing large groups of drones together. In [11], [12], there are approaches to coordinating flocks of drones and managing their flight paths. This was especially significant as we consider the task of maneuvering groups of drones. The survey article, [13], proved invaluable as a recent compilation of the state of the art on drone swarms. It covers a wide range of topics from the the lowest level flight dynamics (important for realistic simulation) to the macro-level control of swarms and insights on typical mission packages.

One of the most impressive uses of large numbers of drones acting together was the lighting display at the opening ceremony of the 2018 Olympic games [14]. Swarms of drones also provide interesting applications beyond those of a single drone. They can spread to cover larger areas or provide different perspectives on the same area of interest. This is reflected in recent work on outdoor cinematography [15], providing previously unavailable shots and unlocking new creative options. This idea has reconnaissance and surveillance applications. Swarms also offer potential in helping to localize and track targets [10]. They offer promise in collecting real-time data or in monitoring large areas for changes over time [2], [16]. Swarms of drones can be deployed to deny an airspace [3] or to attack fixed targets [17].

There is active research into methods for defeating drones and swarms of drones. A number of systems focus on jamming the transmissions to unmanned vehicles, both directed and automatic, [18]–[20]. These systems focus on disabling drones that are under remote control and forcing them to

This work was supported in part by the NSF under Grant Nos. [IIS-1553063] and [IIS-1619278]

[1]D. Biediger is with the Department of Computer Science, University of Houston, Houston, TX 77204 USA debiediger@uh.edu

[2]A. Mahadev and A. T. Becker are with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77204 USA {aviswanathanmahadev,atbecker}@uh.edu

land or return to their launch point. This indicates the need for a drone swarm with minimal communications, especially when near an objective. Another set of solutions is directed at capturing errant drones with nets [18]. Some systems use friendly drones as counters [21], including some drones with their own weapon systems [22]. A larger swarm could overwhelm the ability of such counter to defend an area.

III. Approach

We describe our current approach, which addresses the development and subsequent testing of motion plans for swarms of drones loitering in close proximity to an actively hostile area of interest. The drone swarm must move through an area of interest to accomplish some goal or mission while ensuring that the swarm remains intact. Drones must avoid collisions with the rest of the swarm and survive hostile action, we devise a multistage approach to construct and test potential solutions.

There is a need for a swarm motion plan to move the drones in and through the required area. This plan must enable completing the mission objectives and support the survival of the group. This motion plan should be parametric to minimize computational cost while offering some assurance on non-overlapping motion paths. The path should be of sufficient complexity to make enemy motion estimation difficult. By changing the parameters between missions, we can prevent adversaries from building up experience that could be used against the swarm. The bulk motion plan that the drones follow must minimize need for communication, computation, and sensing, as these increase the cost and complexity of a plan. These can also provide a surface for electronic attack, which could confuse, disable, or destroy the entire swarm.

Beyond the basic plan, there are further refinements that may be added to improve the survivability of drones without altering the general motion plan. These changes can make use of the unique flight characteristics of drones, as long as the additions preserve flight paths that are non-intersecting. By making sudden changes in speed and direction, under some constraint, an element of unpredictability can be added to the motion of individual drones.

To measure the efficacy of this approach, there must be a way to execute and test different scenarios. A simulation, in Virtual Reality (VR), offers an immersive environment for observing and testing different flight paths. The simulation must faithfully recreate the drones, their performance, and their flight characteristics. It should allow for repeated testing of scenarios and for the collection of relevant data in order to measure the way in which an adversary would attack a swarm. In this way, it should reflect the viability of the different motion plans.

A. Bulk motion

Developing a bulk motion plan for a swarm of drones presents a number of significant challenges. The primary challenge is to move a large number of drones through some limited air-space. Sometimes, the drones must fly in close proximity without collision. The natural analogue to this desired behavior is the flocking of birds, the schooling of fish, or swarming of insects. An example of the difference between flocking and swarming in nature is visible in Fig. 1. Unfortunately, birds, fish, and insects are decentralized autonomous agents and reflecting this behavior in drones requires complex communications or expensive sensing to avoid collisions, a concept that has been explored [11].

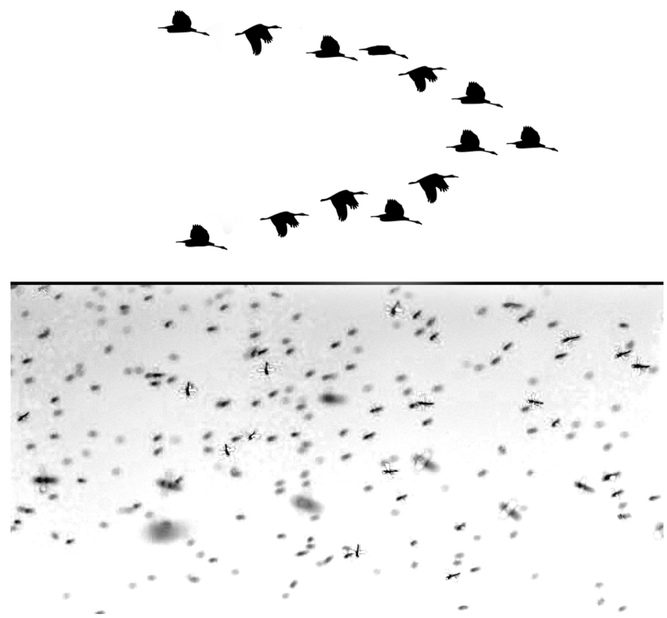

Fig. 1. Flocks of birds and insect swarms serve as the biological models for a swarm of drones.

By selecting parametric paths for the drones to follow, it is possible to prevent collisions between drones in the swarm. Each drone can be assigned a relative position along a path, and allowed to fly in a certain segment of the air-space. If these segments move together and do not overlap, the drones can be reliably prevented from colliding. We selected a set of simple knots on the surface of a torus, positioned around an area of interest. The advantage of knots as flight paths is that they are non-intersecting and can be developed to an arbitrary level of complexity, like the way physical ropes tend to knot themselves in complex ways. The first scenario investigated provides a flight of drones in close formation, all executing the same flight path together. We refer to this flight pattern as a *flock* (L_f), because it resembles a flock of birds. This formation is visible in Fig. 2.

The airspace around a given location is limited, and to minimize the chance for collisions, it is necessary to limit the number of drones flying on a given manifold. Fortunately, it is possible to stack and nest manifolds to provide space for many sub-groups within the swarm. To increase the visual confusion, we alternate the direction of motion of these subgroups, clockwise or counter-clockwise, for each additional manifold. To further increase visual confusion, we moved from a tight formation to a shuffled and widely distributed swarm. The second scenario, *swarm* (L_s), represents a set of

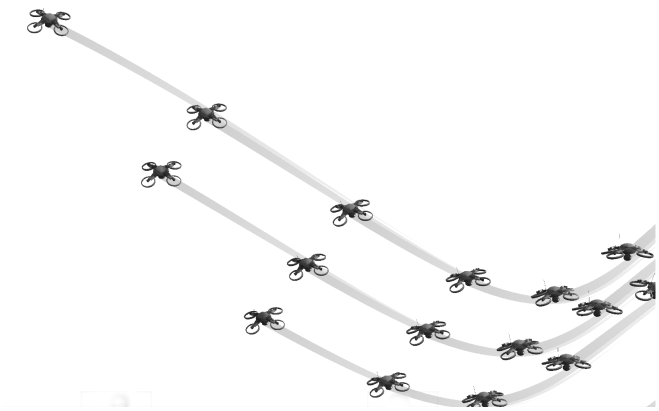

Fig. 2. The flight path of drones in tight formation on the flock level (L_f) including trails to show the flight path.

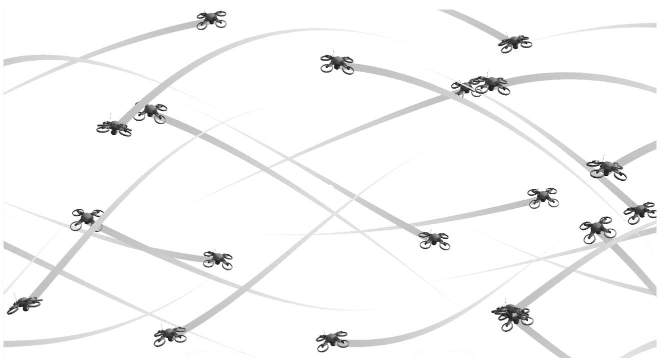

Fig. 3. The seemingly chaotic flight path of drones on the swarm level (L_s) including trails to show the flight path.

drones moving along the same path, but shuffled in position and reordered in time. This results in large gaps and a more chaotic apparent motion. The same overall motion patterns are present, but they are somewhat obfuscated. Drones appear to move in many different directions due to the shuffling of the flight patterns. As a result, we refer to this scenario as a *swarm pattern*, distinct from the flocking of the first scenario. The swarming motion is visible in Fig. 3.

B. Defensive confounding motion

As drones move along parametric paths, patterns of motion may emerge which can be exploited. An adversary could take advantage of these patterns to defeat a swarm. It is easy to imagine a simple path, where drones would circle around an area at a fixed altitude and radius and moving at a constant speed. This simple parametric path would be easy to detect and the parameters (altitude, radius, and speed) would be easy to determine. In addition, drones would pass through the same points in space on successive trips around the area of interest. This presents a dangerous opportunity for an opponent to counter the drone swarm [4], [23].

To confound parameter estimation, we provide a third scenario, *juking (L_j)*, that includes small random motions. The drones still fly the same bulk path, but can wander randomly a few drone lengths from their prescribed trajectory. New offsets are selected randomly and updated regularly. Though the drones in this scenario have the same average speed, they move slightly faster or slower and do not pass through the same points on successive cycles. By carefully selecting the amount of allowable drift, the drones remain inside non-intersecting corridors. The added juking motion makes use of the unique flight characteristics of quadcopters to make abrupt and unpredictable changes in motion.

C. Virtual Reality simulation

The VR simulation described was implemented in the Unity game engine. For the simulation, we used a mixed-reality headset. The simulation provides scenarios in fixed or random order and of fixed duration. It allows for customization of the scenarios to include specific tasks or missions. The simulations allows for a wide range of customization of the levels, including the number of drones, flight paths, type of drones, and weapon characteristics. In the VR simulation, participants are given a virtual weapon, are asked to accurately shoot down drones in a limited time, and placed in a virtual world. The simulation is created to gather data on how humans attack swarms of drones. Much of this information recorded would be difficult or impossible to capture using physical drones. The simulation provides a controlled and repeatable environment for testing.

IV. Analysis

For testing the VR system, we requested consenting volunteers to play through all three levels of the game[1]. In this section, we will refer to the *Flocking Level* as L_f, *Swarming Level* as L_s and *Juking Level* as L_j. In each of the three levels the individual drones followed the same basic flight pattern of a knot on a torus. The drones across all trials flew at the same average speed. The drones with juking enabled would vary in speed around the average, moving slightly faster or slower over time depending on their random motions. Recognizing the need to reflect physical drone behavior in our virtual drone simulations, we limited the velocity and acceleration to values below a conservative estimate of the performance of commercially available quadcopters.

A group of 60 participants played the game which lasts for a total of 3 minutes 45 seconds. Each level has 100 drones and lasts for 60 seconds, followed by a rest period of 15 seconds. The order of the levels is randomized to reduce biasing due to learning or user fatigue between the levels when performing statistical analysis. Participants are instructed that they have infinite projectiles, but that their score will be calculated on the basis of hit ratio $H_{\text{ratio}} = (N_h/N_s)$ and total hits N_h in each level where N_s is number of shots. We normalize the number of hits by N_d, the number of drones in the level and apply a coefficient of 100 as a convenience to scale up the score range.

$$\text{Score} = 100(H_{\text{ratio}})^2(N_h/N_d) \tag{1}$$

[1]This study was conducted under IRB #STUDY00002110 through the University of Houston.

A. Data gathering

The system logs at 60 Hz the orientation and pose data from the VR headset and the controller. This data and the locations of the drone hits are written to text files. After processing the data, we generate three dimensional hit clouds of each participant as seen in Fig. 6. The participant is represented with their eye-level at (0,0,0) (Z, X, Y). Each hit location is corrected by translation with respect to the head of the participant, where Z is depth, X is horizontal position, and Y is vertical position. The values are measured in meters. We also derive the horizontal angle, θ, and vertical angle, ϕ, for each hit.

In addition to the logged data, we collect survey information from participants. After each trial, the participants complete a short 13 question survey. Among the questions, the survey covers gaming and VR experience as well as perceived difficulty of the levels.

Performance	L_f	L_s	L_j	Total score
Good	$90/115$	$96/108$	$82/100$	200.45
Bad	$62/127$	$40/75$	$31/115$	29.89

TABLE I
GOOD VS. BAD PERFORMANCE

B. Data analysis

Our analysis compares the spatiotemporal distribution of hits and drone survivability as a function of game level.

Spatiotemporal analysis compares the distance between successive hits, where $\Delta\theta$ and $\Delta\phi$ are the difference in θ and ϕ between hits and the time between hits is Δt_h. As seen in Fig. 4, flocking behavior results in $\Delta\theta$ and $\Delta\phi$ that are closer to 0 than with other behaviors. This may be because drones all move in the same pattern and are clustered together.

The hit ratios are also higher for L_f and the time between hits Δt_h is lower as a consequence. This is also reflected in the probability distribution of $\Delta\theta$, $\Delta\phi$, and Δt_h, shown in Fig. 5. For each measurement, L_f has a higher peak than the other two levels.

From the dataset gathered, the representative scores for a good (90th percentile) and bad (5th percentile) performance are tabulated in Table I. The 3D hit clouds for these performances are shown in Fig. 6.

Although the performance between participants varies, the plot of scores across percentiles for all levels (Fig. 7) shows that participants consistently perform best in L_f, followed by L_s, with L_j a distant third.

Specific excerpts of the percentile values tabulated in Table II indicate likewise.

These tests support the hypothesis that increased variation in behavior of drones makes them significantly harder to shoot down. Another testable hypothesis is that predictable patterns, even if in different behaviors, results in similar performance and so the scores for L_f and L_s will be close to each other as seen in Fig. 7, even though L_s and L_j use the same base movement. The participants also indicated this in the post game survey.

Percentile	L_f	L_s	L_j	Total
5	15.83	11.11	2.68	29.89
25	30.21	22.77	17.72	85.28
50	42.54	37.90	26.88	108.51
75	59.26	49.12	36.95	143.48
95	81.64	82.98	73.40	205.46

TABLE II
PERCENTILE SCORE

Since the scores are based on the accuracy and ease of shooting down drones in each level, validation by analyzing statistical significance can be causally linked to single parameter changes between the levels. The null hypothesis H_0 is that all scores are drawn from a distribution with the same mean: $\mu_{L_f} = \mu_{L_s} = \mu_{L_j}$. The corresponding alternate hypothesis H_1 is that not all μ's are the same. Good players tend to be good at all levels and poor players perform poorly on all levels. Since the three levels played by the same participant will have dependent scores, we perform *repeated measures ANOVA* (analysis of variance) on the data set. We first identify F_{critical}, the statistical significance value for our analysis. The number of participants $K = 60$ and number of groups (levels) $n = 3$. We then identify the degrees of freedom (*df*), between the groups $df_{\text{between}} = n - 1 = 2$, within groups $df_{\text{within}} = n(K - 1) = 177$, within subjects $df_{\text{subjects}} = K - 1 = 59$ and the error $df_{\text{error}} = df_{\text{within}} - df_{\text{subjects}} = 118$. F_{critical} is then found from the look-up table for F_{critical} where the row is df_{error} and column is df_{between}. We then calculate $F_{\text{value}} = MS_{\text{between}}/MS_{\text{error}}$ where MS stands for mean square.

With 60 participants and 3 levels, for $\alpha = 0.05$, if $F_{\text{value}} > F_{\text{critical}}(2, 118) = 3.07$, then the null hypothesis can be rejected. The values of the analysis are tabulated in table III and $F_{\text{value}} = 17.42$. In the table, SS refers to the sum of squares and p is the probability that our null hypothesis is correct by chance. The three levels vary significantly on scores with a p-value of $2.36 \times 10^{-7} \ll 0.05$. This is the probability that any of our observations are by chance given the dfs and the calculated F_{value}.

	SS	*df*	MS	F_{value}	*p-value*
Between	6.35×10^3	2	3180	17.4	2.36×10^{-7}
Subjects	4.90×10^4	59	830.9		
Error	2.15×10^4	118	182.4		
Total	7.69×10^4	179			

TABLE III
REPEATED MEASURES ANOVA

Therefore, the null hypothesis is rejected. We can say with certainty that changing the behavior of drones to have more variance in velocity vectors makes the level harder. We must

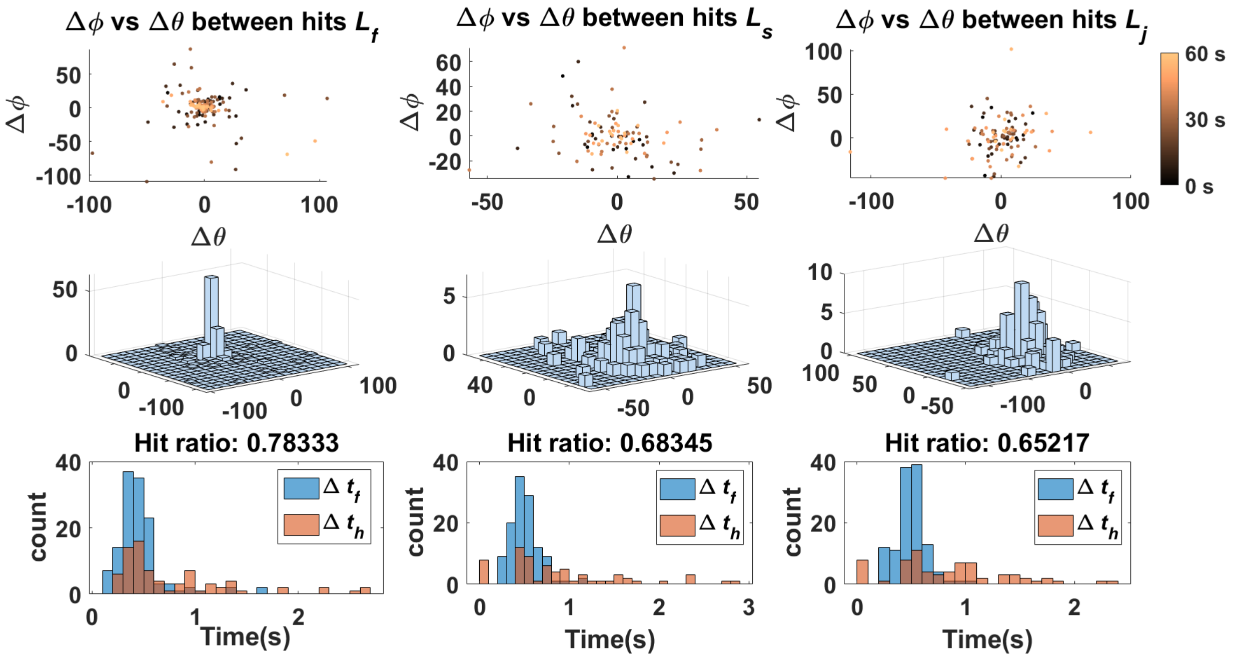

Fig. 4. The first row of plots depict the $\Delta\theta$, $\Delta\phi$ spread in degrees. The color bar is from 0 (dark) to 60 (light) where 0 s is the beginning of the level and 60 s is the end of the level. The bivariant histogram in the second row also shows high clustering near 0. The third row is a histogram plot of the Δt_f, the time elapsed between successive firing of projectiles and Δt_h the elapsed time between successive hits.

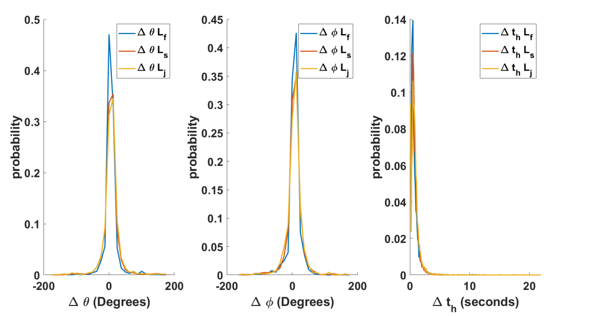

Fig. 5. Probability distribution of $\Delta\theta$, $\Delta\phi$ and Δt_h all having higher peaks for L_f indicative of the relative ease of hitting drones when following flocking behavior

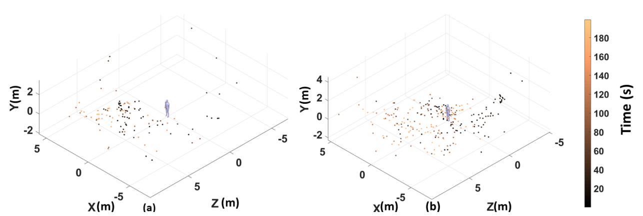

Fig. 6. Three dimensional spread of hit cloud. The color of the hit corresponds to the time in seconds when a drone was shot down. (a) represents a poor performance with 143 hits out of 300 and (b) represents a good performance with 268 hits out of 300

still identify which change is more significant, for which we can perform the *two sample T-test* among the levels. For this test, the null hypothesis H_0 is that the scores in any pair of levels are independent random samples from normal distributions with equal means and equal variances.

The alternate hypothesis is H_1; The scores in any pairs of

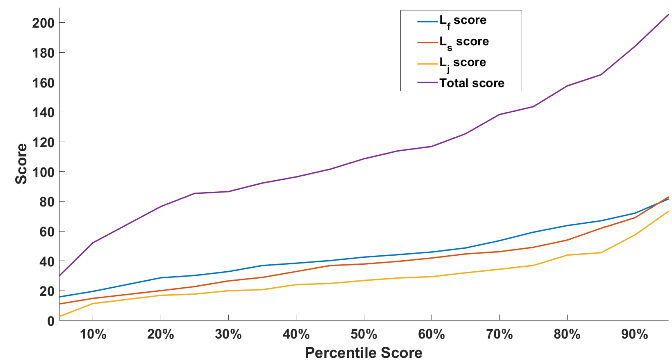

Fig. 7. The plot in purple is the total of the scores as a function of percentile. The L_f score (blue) is close to the L_s score (red), but the score difference is highest between L_f score (blue) and L_j score (yellow).

levels comes from populations with different distributions. On calculation we get $pL_{f_s} = 0.14$, meaning the probability that $\mu_{L_f} \neq \mu_{L_s} \approx 86\%$. $pL_{s_j} = 0.017$, meaning that the probability that $\mu_{L_s} \neq \mu_{L_j} \approx 98\%$, higher than the difference between performance in L_f and L_s. $p_{L_{13}} = 9.14 \times 10^{-5}$, meaning that the probability that $\mu_{L_f} \neq \mu_{L_j} \approx 100\%$, significantly higher than the difference between the scores between L_f and L_s or L_s and L_j. This result supports our hypothesis that the performance difference between the predictable levels L_f and L_s is not as significant as the difference between predictable (L_f and L_s) and unpredictable pattern L_j as shown by the box-plot in Fig. 8.

V. Conclusions and Future Work

The VR system developed has the capacity to handle simulation of large swarms of drones in an immersive expe-

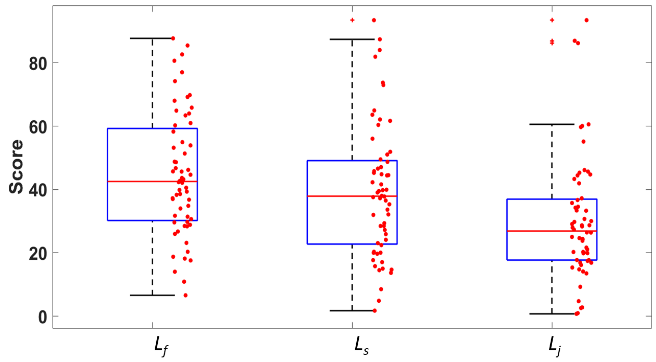

Fig. 8. Box-plot comparison of the scores in the three levels. The distribution of scores achieved in the parametric level are significantly lower than the distribution of other two levels. The scatter in red are the actual scores of all the participants.

rience. We are able to use our scoring and analysis method to investigate the survivability of flocking and swarming patterns in a quantifiable way. Our scoring system helps quantitatively study the differences in levels and the analysis shows that survivability is improved as predictability drops. L_f is predictable since all the velocity vectors are the same. The small spacing between drones in L_f also reduces cognitive strain and increases the confidence of shooters. Introduction of variability in velocity in L_s improves survivablity, however, as mentioned by participants and as observed in the scores, even with the seemingly chaotic flight path in L_s once participants pick up on the pattern, the drones are easy to shoot. The random motion in L_j correlates with a decrease in the trigger pull rate. This is likely due to the added difficultly in tracking the drones and leads to an improvement in drone survivability.

With the VR simulation system in place, we can evaluate the effectiveness of different motion plans and test additional scenarios. We plan to include different mission scenarios for the drones and evaluate swarm motion strategies to ensure the success of these missions. We can then test and optimize the swarming behavior across a number of potential missions.

With the rich set of data available from participants, we could construct a shooter-model that reflects the attributes of the top scoring participants. By modelling the pose of the shooter as a unknown state, we can apply state estimation techniques to develop controllers that work on individual drones hence allowing for decentralized control. Using this model, we could quickly evaluate potential swarming strategies before testing them against human adversaries. With a vetted shooter-model, we plan to employ deep learning to generate swarm strategies and test them against virtual shooters.

References

[1] J Senthilnath, Manasa Kandukuri, Akanksha Dokania, and Kn Ramesh. Application of uav imaging platform for vegetation analysis based on spectral-spatial methods. *Computers and Electronics in Agriculture*, 140, 2017.

[2] A. Clark. Small unmanned aerial systems comparative analysis for the application to coastal erosion monitoring. *GeoResJ*, 13:175–185, 2017.

[3] Leslie F. Hauck and II Geis, John P. Air mines: countering the drone threat to aircraft.(views). *Air & Space Power Journal*, 31(1), 2017.

[4] A new dogfight; combating drones. *The Economist*, 430(9127), 2019.

[5] Irving Lachow. The upside and downside of swarming drones. *Bulletin of the Atomic Scientists*, 73(2):96–101, 2017.

[6] Patrick Tucker. U.S. deploys unmanned vehicles. *The Futurist*, 43(6):12–13, 2009.

[7] Johnson Dave. Where humans dare to go: drones will carry out dangerous inspections.(cover story). *Industrial Safety & Hygiene News*, 51(8), 2017.

[8] Helen Knight. Firefighting robot to replace humans: Qinetiq develops remote-controlled vehicle to send into hazardous situations too dangerous for emergency services. (robotics). *The Engineer*, 291(7604), 2002.

[9] A Alfeo, N De Francesco, A Lazzeri, M Lega, and G Vaglini. Swarm coordination of mini-UAVs for target search using imperfect sensors. *arXiv.org*, 12(2), 2019.

[10] Antonio L. Alfeo, Mario G.C.A. Cimino, Nicoletta De Francesco, Massimiliano Lega, and Gigliola Vaglini. Design and simulation of the emergent behavior of small drones swarming for distributed target localization. *Journal of Computational Science*, 29:19–33, 2018.

[11] R. Olfati-Saber. Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, March 2006.

[12] Gábor Vásárhelyi, Csaba Virágh, Gergő Somorjai, Tamás Nepusz, Agoston E. Eiben, and Tamás Vicsek. Optimized flocking of autonomous drones in confined environments. *Science Robotics*, 3(20), 2018.

[13] S. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar. A survey on aerial swarm robotics. *IEEE Transactions on Robotics*, 34(4):837–855, Aug 2018.

[14] Chinese company claims to have broken world record with 1,374 dancing drones. *ABC Premium News*, 2018.

[15] I. Mademlis, V. Mygdalis, N. Nikolaidis, M. Montagnuolo, F. Negro, A. Messina, and I. Pitas. High-level multiple-uav cinematography tools for covering outdoor events. *IEEE Transactions on Broadcasting*, pages 1–9, 2019.

[16] Pablo Garcia-Aunon, Juan Jess Roldn, and Antonio Barrientos. Monitoring traffic in future cities with aerial swarms: Developing and optimizing a behavior-based surveillance algorithm. *Cognitive Systems Research*, 54:273–286, 2019.

[17] David Hambling. Swarm of drones attacks airbase. (Russian forces in Syria) (News & Technology). *New Scientist*, 237(3161), 2018.

[18] J. R. Wilson. The new world of counter-drone technology, November 1 2018.

[19] Yasmin Tadjdeh. Industry eyes opportunities in counter-drone market. *National Defense*, 102(768):37–38, 11 2017.

[20] Staff Writers. MyDefence demonstrates drone swarm counter UAS jammer. *UPI Space Daily*, Oct 30 2018.

[21] Allyson Versprille. Marine corps developing low cost robot swarms to counter enemy drones. *National Defense*, 100(741):32 – 33, 2015.

[22] Anonymous. UAVs learn how to defend themselves. *Flight International*, 193(5635):13, 2018.

[23] David Cenciotti. Army air force video explains how to evade flak (anti-aircraft fire) in WWII. *The Aviationist*, 2014.