AUVSI SUAS Competition 2022

# Technical Design Paper

**UBC Unmanned Aircraft Systems**
University of British Columbia
2329 West Mall
Vancouver, BC Canada V6T 1Z4
info@ubcuas.com

## 1. Abstract

This paper provides support and technical overview of the unmanned aerial system (UAS) solution developed by the University of British Columbia Unmanned Aircraft Systems (UBC UAS) engineering design team. The system is designed for the missions outlined in the Association for Unmanned Vehicle Systems International Student UAS Competition, including strategies for autonomous flight, obstacle avoidance, object detection, classification, localization, mapping, and air drop. UBC UAS has created a safe, cohesive, and robust system that is optimized for the presented missions through a combination of aircraft, software, communications, and robotics design.

## 2. Requirements & Acceptance Criteria

### 2.1 Imaging System

The imaging system allows for the detection of objects for the Object Detection, Classification, Localization (ODCL) part of the mission. The imaging system must be able to provide images where the objects are at least 18 pixels wide from a minimum flight altitude of 30.48m. This requirement comes from the minimum resolution required to do letter recognition reliably. A successful imaging system would meet the minimum resolution while being under 1kg.

### 2.2 Object Detection, Classification, Localization

The ODCL mission can be split into two parts: finding the objects and identifying the objects. For finding the objects, an area search algorithm is required, with preference for the algorithms that generate the fastest paths. For identifying the objects, most points are awarded to accuracy rather than autonomy. So a software interface where a user can select, crop, and classify images is set as a minimum requirement. An autonomous system might yield more points but its consistency is likely to be not satisfactory, thus a manual system should be developed even if it is a back-up. A satisfactory ODCL system would get 80% of the available points for ODCL.

### 2.3 Mapping

The mapping task requires an aircraft that can traverse the whole mission area. This means a fast aircraft is required for this part of the mission. Alternatively, a higher resolution imaging is needed. With a high resolution camera, the Unmanned Aerial Vehicle (UAV) can fly at a high altitude without losing the required details. To accomplish mapping, the system would require a combination of high speed aircraft and a high resolution imaging system.

### 2.4 Air Drop

The air drop system needs to be able to deliver a standard 8oz water bottle via Unmanned Ground Vehicle (UGV) to a designated target location. The deployment system needs to ensure that the UGV is lowered from the air drop altitude to a maximum of 5 feet off the ground. The UGV must also be able to fall from 5 feet without damage. Additionally, the entire UGV and deployment system must weigh less than 2.2 kg. The deployment process should be done in under 1 minute. The UGV must be deployed to an accuracy of less than 40ft and the UGV must drive to its location with an accuracy of under 10ft.

### 2.5 Aircraft

To maximize our mission time, the aircraft needs to have a flight time of at least 30 minutes. To complete the mission within our flight time window, our UAV should have a maximum speed of at least 17 m/s and have a flight range of at least 16 km. Furthermore to accurately complete the waypoint pathing, the aircraft's minimum turn radius should be less than 50 meters. The minimum takeoff distance would have to be less than 80 meters to takeoff at competition. To complete the airdrop mission described above, our aircraft will have a minimum payload capacity of 1.2 kilograms and 150 x 150 x 120 mm. Furthermore, the minimum speed of the aircraft must be less than 10m/s for accurate drop. To ensure safety of our system, a kill switch and the aircraft's ability to return-to-launch (RTL) are required.

### 2.6 Communication

It is very important to ensure robust communication between the ground station and aircraft for flight routing and control, telemetry, emergency manual radio control (RC), and sending images for ODCL over a 4km range. There must be no loss in the communication links for the flight routing and emergency RC. The image communication link requires a bandwidth of at least 15 Mbps.

### 2.7 Autopilot & Obstacle Avoidance

For autonomous flight, an autopilot that is capable of executing autonomous commands throughout the mission is required. This autopilot needs to be able to plot a course for the aircraft, avoid stationary and moving obstacles, signal components on the aircraft to act (such as the airdrop system) while accounting for the aerodynamics of the aircraft it is tasked to control. It also needs to account for the aircraft's flight time, battery voltage, and mission time. A successful autopilot system would be able to plan the aircraft's course, avoid all obstacles, and hit all waypoints with an accuracy of 85% while sending and receiving commands from other components of the system. Additionally, the system should upload its telemetry to the Interoperability System to reduce the chance of a collision. In case of a detected failure, it should begin the RTL procedure.

## 3. System Design

### 3.1 Imaging System

The UAS described in this paper uses a 20MP camera - the FLIR BFS-U3-200S6C-C[1]. This camera is accompanied by a

[1] Flir.ca. 2022. Blackfly S USB3 — Teledyne FLIR. [online] Available at: ¡https://www.flir.ca/products/blackfly-s-usb3/?model=BFS-U3-200S6C-C¿ [Accessed 14 February 2022].

12mm F2.6 lens, the COMPUTAR V1226-MPZ[2]. The microprocessor on the UAV sends image triggers continuously via USB to the camera in order to capture images programmatically. The microprocessor also sets the camera's exposure settings automatically depending on light conditions (see Figure 1).

**Figure 1.** Microprocessor auto-exposure at work

The imaging system is also capable of providing high resolution ODCL imagery at distances exceeding the minimum height requirements. Figure 2 shows a result of image capture during simulated vertical range testing performed using the previously described system. At a simulated altitude of 50m above ground level (AGL), the objects are of adequate resolution and clearly visible. Thus, the objects may be decomposed effectively into their features.

**Figure 2.** Image of scaled objects taken from 50m AGL

## 3.2 Object Detection, Classification, Localization

The solution created for ODCL is a network of software across distinct platforms. The system processes objects in 4 distinct steps: metadata insertion, data transport, image projection and manual classification. During flight, images are taken by the UAV and geotagged with relevant flight data—latitude, longitude, altitude, heading, roll—using the DroneKit-Python SDK (Software Development Kit). Trigonometry is used to ensure that the location of the ODCL is as accurate as possible. As seen in Figure 3, the image is mapped in 3D space based on inertial measurement unit (IMU) measurements, altitude and the camera's field of view. An object's location inside the image is then projected from the camera, through the mapped image and onto the ground plane to determine the exact location in real space. In order to ensure that all images can be processed in the allotted time This data is sent to the IMP server, where Global Positioning System (GPS) adjustments are applied to calculate the real GPS location and crop the image.

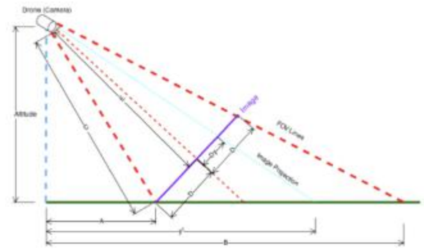

**Figure 3.** Image projection trigonometry

## 3.3 Mapping

Since the UAS can perform survey missions autonomously and is equipped with a gimballed camera, it has the capacity to perform mapping tasks. However, with respect to this task, the UAS is currently limited by its software capabilities. Instead of dedicating team members to writing map seam blending software, it was decided that the team would focus on the other tasks this year.

## 3.4 Air Drop

UBC UAS designed an air drop system consisting of a winch deployment mechanism and a differentially steered UGV. Once our aircraft is positioned above the drop location, the winch will deploy the UGV. The descent is controlled through a Proportional, Integral, Derivative (PID)-controlled brake system to adjust the drop speed. The winch uses a magnetic encoder to determine the velocity of the drop. In order to optimize the drop time, the drop speed is gradually slowed from the initial release speed to the target UGV release speed at the end of the spool.

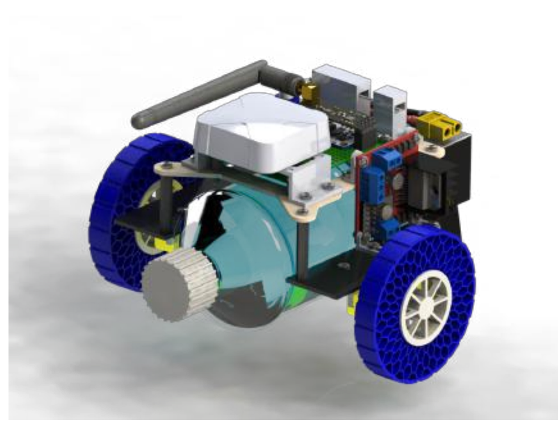

**Figure 4.** Render of the UGV

[2] BACK-BONE. 2022. Computar V1226-MPZ 1" 12mm 20MP f2.6 C-Mount Lens - BACK-BONE. [online] Available at: ¡https://www.back-bone.ca/product/v1226-mpz/?v=3e8d115eb4b3¿ [Accessed 14 February 2022].

The UGV consists of two differentially driven wheels with a third wheel as support. The 8oz water bottle is secured to the UGV frame as shown in Figure 4 via zip ties. The UGV's driving wheels are 3D printed hybrid thermoplastic polyurethane/polylactic acid honeycomb wheels, designed to deform and absorb shock loads. It is powered by a 3S LiPo battery with 15 minutes of battery life. Once the UGV detects landing with the onboard IMU, it releases itself from the winch. After disconnecting, the winch will retract. The UGV then uses a PID controller with GPS feedback to navigate to the destination. The vehicle will also check if the distance between the target and itself is less than 10ft. and stop moving.

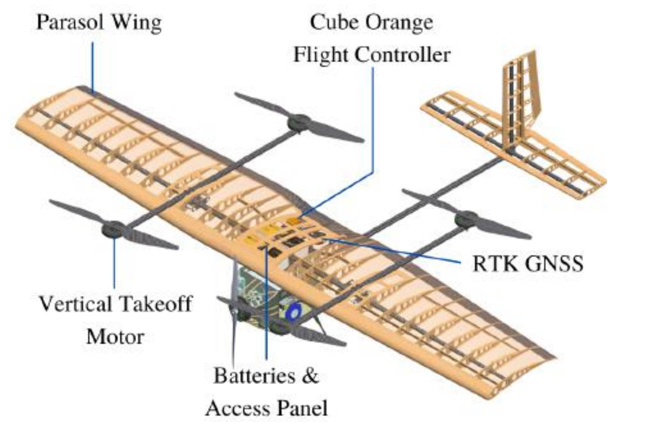


**Figure 6.** Labelled full aircraft diagram

### 3.5 Aircraft

UBC UAS designed Beetle, an electrical vertical takeoff and landing (e-VTOL) aircraft with a QuadPlane setup. Beetle can hover for accurate airdrop and can transition to forward flight for high speed, endurance flight to accomplish all missions within the 30-minute flight window (shown in Figure 5). Beetle's parasol wing is tapered for drag optimization. The wing also serves as a structural component and holds flight controllers, radio devices, batteries and vertical propulsion motors that act as wing bending relief loads (shown in Figure 6). This wing design allocates space for the UGV inside the fuselage while minimizing the weight and form drag as seen in Figure 7.

| Requirement | Metric |
|---|---|
| Flight Time (min) | 32.0 |
| Hover Time (min) | 5.0 |
| Min. Speed (m/s) | 13.9 |
| Max. Speed (m/s) | 21.3 |
| Turning Radius (m) | 27.2 |
| Flight Range (km) | 22.3 |
| Aircraft Weight (kg) | 10.8 |
| Payload Capacity (kg) | 2.2 |

**Figure 5.** Flight profile requirements

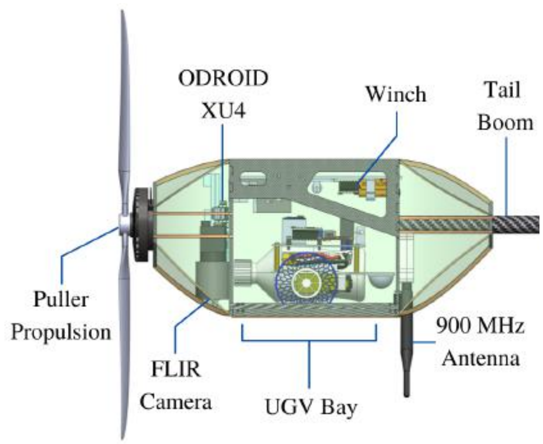


**Figure 7.** Labelled fuselage diagram

Beetle has a parasol wing with 2 different airfoils and a total wingspan of 2.4m. The root uses the airfoil N-24 and transitions into S-1223 at 0.215m from the wing root, as shown in Figure 8 along with the lift and drag distributions[3]. The twist angle, and taper ratio were determined by iterating the analyses to optimize for efficiency when compared to an elliptical lift distribution.

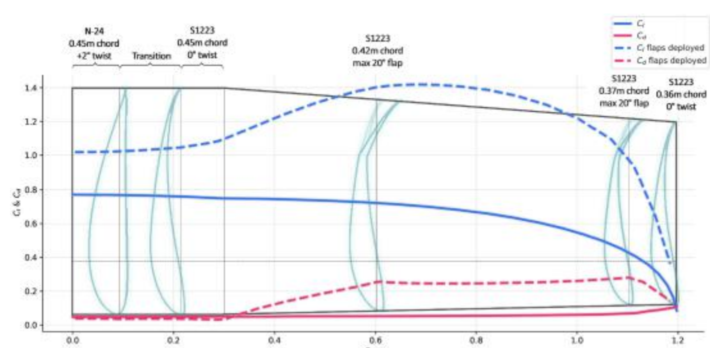


**Figure 8.** Wing profile with lift/drag distributions

[3] Analysis for performance was conducted on XFLR5 using linear line theory and verified using vortex lattice method

Beetle uses a NACA 0015 symmetric airfoil for its vertical and horizontal tail. The horizontal tail has a 0.8m span, with a 0.2m root chord and 0.16m tip chord. The vertical tail has a 0.25m span with a 0.2m root chord and 0.16m tip chord. Stability analysis from XFLR5 was utilized to adjust the center of gravity (CG) of the aircraft, determine the distance between the wing and tail, and size the tail. Furthermore a flow analysis was conducted to ensure the tail was not affected by the down wash of the wing. The QuadPlane setup also allows additional stability and control in forward flight from the vertical propulsion system.

Beetle's wing and tail use a balsa rib structure with tapered carbon tubes and an UltraCote skin. This allows for a lightweight yet high strength build. Structural components such as wing-fuselage attachment, fuselage bulkheads, and landing gear mounting plates are built using a balsa-carbon fiber sandwich for increased strength.

The propulsion systems for forward and vertical flight were selected using a custom python script that minimized the weight and optimized thrust to power ratio using a variety of motor and propeller data. The target thrust values were determined iteratively depending on the weight of the propulsion system. Figure 9 shows the propulsion system selected.

| | Vertical Propulsion | Forward Flight Propulsion |
|---|---|---|
| Motor | Tmotor U8 Lite 100 KV | Tmotor U8 Lite 85 KV |
| Propeller | Tmotor P22x6.6" | APC 18x10E |
| Maximum Thrust per Propeller (N) | 45.73 | 30.74 |
| Maximum Power per Motor (W) | 808.44 | 519.00 |
| Thrust to Power Ratio (N/W) | 0.0566 | 0.0592 |
| Total Weight (kg) | 1.13 | 0.32 |

**Figure 9.** Selected propulsion system

The total battery capacity required for 19 minutes of cruise, 5 minutes of hover, takeoff and landing was calculated to be 516.67Wh and Beetle has a total capacity of 547.2Wh (12S, 12Ah Lipo Batteries).

Due to Beetle's small fuselage, it uses a tail dragger landing gear made using aluminum. The landing gear was designed to support vertical landing and conventional horizontal takeoff. During the design, the landing gear was analyzed using FEA to support up to $5g_0$ of impact force.

## 3.6 Communications

The system pictured in Figure 10 is connected through the custom Ground Control Station (GCS), responsible for proxying, processing, and managing: telemetry, autonomous missions, imagery, and control inputs. The GCS gets images from Sunflower, an Antenna Tracking Station, which maintains a high-bandwidth WiFi connection to the aircraft. It fulfills this by aiming a 2.4 GHz directional Yagi antenna in the direction of the aircraft at all times. UAS uses a 928MHz link to get telemetry from the aircraft and send commands from the Ground Station during the mission. For emergency RC takeover, UAS uses a Dragon Link 433MHz RC system, to ensure a robust long distance connection is maintained.

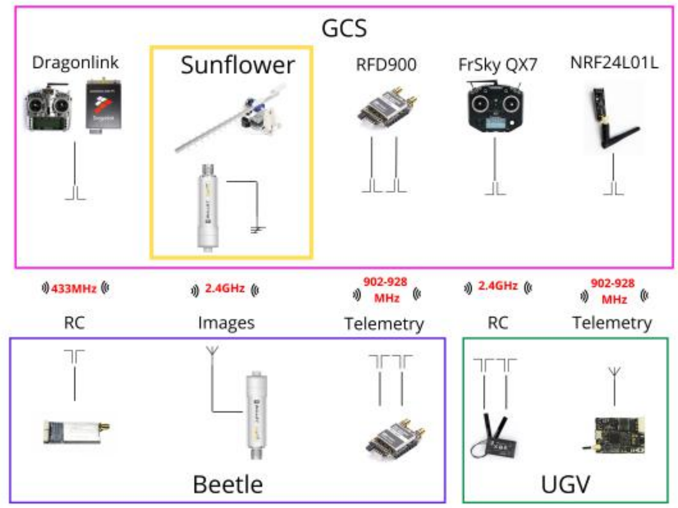


**Figure 10.** Communications block diagram

## 3.7 Autopilot

### 3.7.1 Autopilot

The team is using ArduPilot as its flight controller software. ArduPilot provides precise waypoint navigation, autonomous takeoff, and autonomous landing. Missions are optimized prior to upload, such that total flight distance is minimized. Additionally, the included waypoint navigation feature will aid in object avoidance, task rescheduling and mission route alteration. The team will also be able to avoid manual takeover penalties by using ArduPilot's autonomous landing/takeoff features. ArduPilot is run on the CubePilot Orange, with a triple redundant IMU system, an abundance of industry-standard connection points and internal vibration isolation.

### 3.7.2 Ground Control Station

In order to monitor and control the actions of this autopilot board, a GCS named GCOM-X was developed, shown in Figure 11. The GCS provides a user-friendly interface for operators to connect to the interoperability server and displays obstacles and waypoints in real time. The software also enables operators to upload, start, pause, and abort fully autonomous missions. In addition, it displays real-time telemetry and system-wide logs from UBC UAS' suite of software.

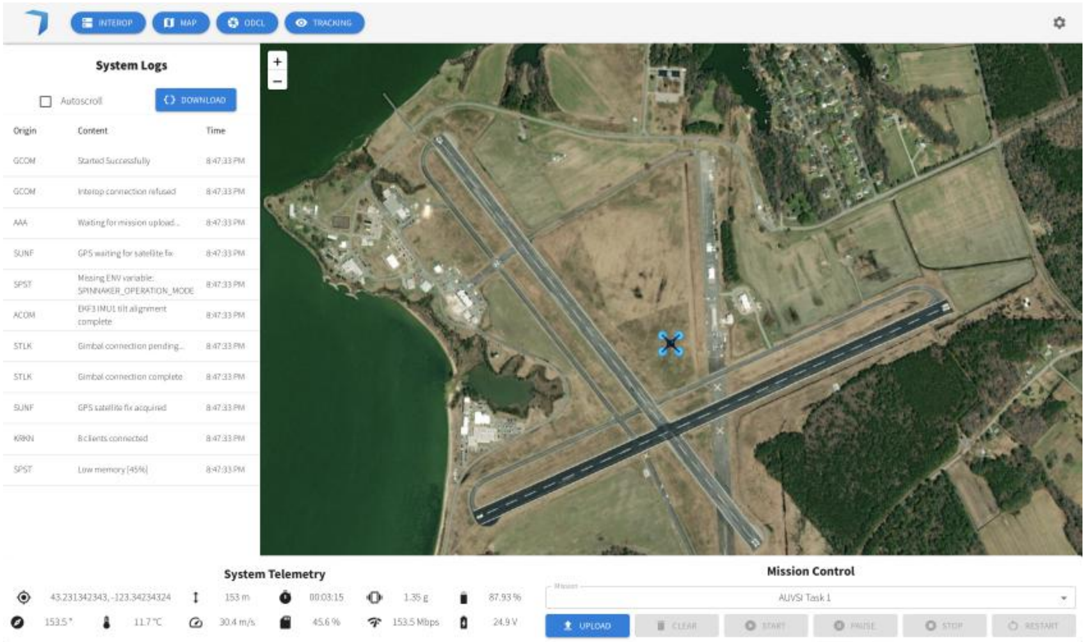


**Figure 11.** Screenshot of GCOM-X GCS user interface

## 3.8 Obstacle Avoidance

The core algorithm used for avoidance and routing in our system is based on a modified version of the A* algorithm called L1. The A* algorithm[4] treats the flight space as a 2D grid, and is computed as a graph. This grid was pre-processed with specific nodes (the waypoints given, points close to the obstacles and points around the inside edge of the flyzone) to reduce the number of potential paths subject to algorithmic consideration. To further speed up the routing, the edges of the graphs between the points are pre-computed immediately after the mission information is gathered from the interop server.

The system is also capable of active aircraft avoidance. A new microservice was developed for converting the telemetry of other aircraft obtained via the interoperability server into dynamic obstacles. If an aircraft enters the buffer zone within a calculated radius of our UAV, a reroute is performed autonomously using previously described methods to prevent a mid-air collision. The buffer zone is determined autonomously using the difference in velocity between the UAVs and a desired timeframe for execution of evasive maneuvers.

[4] "A* search algorithm," GeeksforGeeks, 06-Feb-2022. [Online]. Available: https://www.geeksforgeeks.org/a-search-algorithm/. [Accessed: 14-Feb-2022].

# 4. Alternatives Considered

## 4.1 Aircraft

Alternate solutions considered for the Aircraft included a Quadcopter, a conventional fixed wing aircraft, and a Tiltrotor e-VTOL. A quadcopter could achieve much lower flying speeds compared to a fixed wing aircraft and VTOL options, which would not allow us to attempt the ODCL and mapping parts of the mission. Although a conventional aircraft would achieve better flight performance than a VTOL, it would make the airdrop a lot more difficult and the aircraft would have lower flight stability compared to a QuadPlane. A tiltrotor, like a QuadPlane, would allow the aircraft to utilize forward flight for efficient flight and can transition to hover for accurate payload drop. Tiltrotors can be lighter due to fewer motors, however it adds a lot of complexity to the aircraft and the propulsion systems for forward flight and vertical flight. The selected QuadPlane has independent propulsion systems and provides additional dynamic stability from vertical motors during forward flight.

## 4.2 Autonomous ODCL

The team considered doing ODCL fully autonomously. This option was assessed by doing preliminary classification testing. Sample images were gathered and a letter detection algorithm was written with Python. The algorithm was able to identify 50% of the letters correctly (sample size of 14) but had trouble identifying colors and shapes and only 20% of the characteristics. Using these results, the team decided to not pursue autonomous ODCL.

### 4.3 Onboard Microprocessor

The two contenders for the microprocessor to be used on the aircraft were Odroid XU4 and Raspberry Pi 4. The main considerations for this selection were sufficient performance and I/O with low power draw. Performance was measured by the amount of 20MP images the microprocessor can process and save from the camera. The payload system runs on a Teensy 4.0, so the main microprocessor must have GPIO.

From these alternatives, ODROID-XU4 was selected. Raspberry Pi 4's lower power draw was not enough to offset the performance difference. Alternatives such as NVIDIA Jetson and UDOO BOLT V8 consume too much power.

### 4.4 Air Drop

The air drop system consists of a deployment mechanism and bottle delivery vehicle. Three primary methods for UGV deployment were considered. The simplest option was to directly drop the UGV, but a UGV capable of withstanding the impact would exceed the weight budget. Testing revealed that parachutes would cause difficulty in reaching the accuracy goal. The selected winch design allows us greater control in deploying the UGV within the weight constraints.

### 4.5 Communications

A solution considered for improving line-of-sight communications was a fixed wing aircraft communications relay. This design originated from concern over multipath interference caused by the signals bouncing off of the ground. To minimize multipath interference, an additional aerial vehicle can be used as a relay. This approach is valid, however it adds complexity and points of failure to the system. Furthermore, testing done by UAS showed that path loss due to multipath interference was below the significant signal disturbance threshold.

## 5. Testing  Evaluation Plan

### 5.1 Developmental Testing

Tests for all systems are crucial in development. Figure 12 defines tests performed on the physical system during development. For software development, testing the autopilot, pathing, obstacle avoidance, and ODCL is mainly done by simulations, unit, and functional tests. These software testing methodologies allow for testing code without the need of a physical device other than a computer. For each change to the software, the code has to pass unit and functional tests to be merged in. Every week a mission simulation involving simulated aircraft is conducted to ensure that the changes work together and do not cause any failures.

### 5.2 Mission Testing

The approach to mission testing involves breaking down the full mission into smaller, repeatable mission tests that can verify that the subsystems meet the requirements before doing a full mission demonstration. These smaller task-specific tests can be used to optimize the order of operations in the flight window and verify flight line checklists to ensure smooth operations at the time of demonstration. The smaller mission tests are as follows: setup and teardown, autonomous waypoint flight, obstacle avoidance, object detection, classification, and localization, mapping, and airdrop. Figure 13 describes the subsystem mission tests in greater detail.

## 6. Safety, Risks,  Mitigations

### 6.1 Developmental Risks & Mitigations

Please refer to Figure 14 to read about the safety risks in development and their associated risk mitigation procedures.

### 6.2 Mission Risks & Mitigations

Please refer to Figure 15 to read about the safety risks in mission operations and their associated risk mitigation procedures.

## 7. Conclusion

After performing extensive technical analysis, developing an innovative unmanned aerial vehicle and mission-specific systems, and testing the system for all operation-critical conditions, UBC UAS believes that the solutions presented will be able to perform at a very high level at the 2022 AUVSI competition.

| Subsystem | Purpose/ Requirement | Test Methods | Metric |
|---|---|---|---|
| Aircraft | Verify propulsion system thrust and power curve | Use a Motor Test Stand designed by UAS to verify thrust vs power | Thrust vs power curve chi-square within 95% confidence interval |
| Aircraft | Endurance flight time > 30 min | Simulate the aircraft mission flight (including takeoff, hover, flight and landing), with a fully charged battery. | Total Flight time (min) |
| | Hover time > 3 min | | Hover Time (min) |
| Aircraft | Maximum flight speed > 17 m/s | Test flying aircraft at maximum throttle with RC control over a 10 minute window and determine the average of peak speeds achieved from flight logs | Speed (m/s) |
| Aircraft | Autonomous flight test | Controlled flight in hover mode | Checklist Pass/Fail |
| | | Controlled flight to test transition with ability to switch to hover | Checklist Pass/Fail |
| | | Controlled flight to test forward flight stability with ability to switch to hover | Checklist Pass/Fail |
| | | Controlled flight to test conventional takeoff | Checklist Pass/Fail |
| Aircraft | Minimum turn radius | Fly the aircraft in forward mode with RC control and determine the minimum turning radius from flight logs. | Turning Radius (m) |
| Aircraft | RTL safety trigger | In a controlled environment, initiate an RTL from the ground station. | Checklist Pass/Fail |
| Communi-cations | Communications range test | Test image transfer, telemetry, and RC connections using a bench test on ground with radio line of sight | Distance communication links are maintained until (km) |
| | | | Bandwidth for the 2.4 GHz link at 4km distance (Mbs) |
| Airdrop | Deployment Time | Deploy the rover a distance of 30m while timing to the deployment through a physical timer and through onboard microcontroller timer logs | Time(s) |
| Airdrop | Autonomous Deployment | Deploy the rover and use logs to ensure that it detected a landing. This should also be done with false landings and differing conditions | Checklist Pass/Fail |
| Autopilot | Autonomous waypoint flight | Execute a fully autonomous mission and record telemetry for the entirety of it. Compare telemetry and planned route to verify that the autopilot is functioning as expected. | Within 2m of >80% of the points |
| ODCL | GCS IMP test | Use IMP to manually classify and submit an object. Check locally hosted interoperability server for object submission confirmation. | Pass/Fail |

**Figure 12.** Developmental Tests

<table>
<tr><th>Mission Task</th><th>Purpose/Requirement</th><th>Test Methods</th><th>Metric</th></tr>
<tr><td rowspan="4">Setup/ Teardown</td><td rowspan="4">To ensure safe, timely, and methodical setup and teardown of all system components necessary for a successful mission</td><td rowspan="4">Time the setup and teardown of the aircraft, payload system, communications, and software systems with the use of checklists. Time and practice pre-flight briefing.</td><td>Checklist Pass/Fail</td></tr>
<tr><td>Setup < 10 mins</td></tr>
<tr><td>Debrief < 5 mins</td></tr>
<tr><td>Teardown < 10 mins</td></tr>
<tr><td>Autonomous Waypoint Flight</td><td>For validating precision of waypoint navigation and reliability of autonomous flight to maximize accuracy points</td><td>Connect GCS to aircraft, feed waypoints to GCS, compare overflight path GPS coordinates to the set waypoint coordinates.</td><td>$\max(0, \sqrt{\frac{(100ft - distance)}{100ft}})$</td></tr>
<tr><td rowspan="2">Obstacle Avoidance</td><td rowspan="2">To ensure UAV is capable of avoiding stationary obstacles and downloaded telemetry of other vehicles</td><td rowspan="2">Connect GCS to aircraft, feed flight path coordinates with set stationary obstacles of varying cylinder sizes. Compare overflight path GPS coordinates to ensure obstacles are avoided.</td><td>Pass/Fail</td></tr>
<tr><td># Objects successfully ID’d in 15 mins</td></tr>
<tr><td>Object Detection, Classification, and Localization</td><td>To ensure objects can be identified, cropped, described, and located at the overflight altitude and path</td><td>Connect GCS to aircraft, feed flight path coordinates. Compare detected objects and their characteristics with the actual objects and their characteristics.</td><td>Number of correct characteristics defined will be scored with a scoring equation*</td></tr>
<tr><td>Mapping</td><td>To decide if the quality of the map and the time taken to produce the stitched map can be completed in the flight window</td><td>Connect GCS to aircraft, outline map boundaries and record time taken to generate path, collect images, and generate map.</td><td>Time taken to complete mapping task < 5 mins</td></tr>
<tr><td rowspan="3">Air Drop</td><td rowspan="3">To validate the lateral drop accuracy and drive to location to achieve full points</td><td rowspan="3">Takeoff, feed target GPS coordinate to GCS, winch release UGV. Record touchdown location and UGV stop location.</td><td>Drop distance < 5ft</td></tr>
<tr><td>Delivery < 5 mins</td></tr>
<tr><td>UGV stop < 10ft GPS</td></tr>
<tr><td>Full Mission Test</td><td>To ensure a full mission demonstration (including all the tests above) can be completed reliably without penalty</td><td>Perform a full mission test including all of the tasks above in a simulated timed competition flight window.</td><td>Mission time < 40 mins</td></tr>
</table>

$$*score = \frac{number\ of\ correct\ characteristics}{total\ number\ of\ characteristics} * \frac{2}{recall\ of\ detection + precision\ of\ detection}$$

**Figure 13.** Mission Tests

| Safety Risk | Risk Mitigation Procedure |
|---|---|
| Misused manufacturing tools | Persons carrying out manufacturing have adequate training. Manufacturing is not carried out alone. Appropriate Personal Protection Equipment (PPE) is provided. All members are certified in university-provided workplace safety courses. |
| Poorly maintained manufacturing tools | Equipment is regularly checked for disrepair. Potential hazards are identified and appropriate parties are notified. Equipment is repaired promptly or replaced when necessary. |
| Improper handling of electronics with high energy capacity | Relevant persons are adequately trained and are aware of the hazards and what actions can cause an incident. Work with electronics is not carried out alone and is cross-checked with other trained persons. |
| Lithium batteries | Before carrying out any test, all hazards are identified and mitigation is applied where possible. Lithium batteries are stored in fire-safe containers and bags. |
| Miscalculations | UBC UAS members carry out design reviews and calculation appraisals to validate designs. Members maintain extensive documentation where assumptions and test findings can be factored into the validation. |

**Figure 14.** Developmental Risks

| Safety Hazard | Risk Mitigation Procedure |
|---|---|
| Other aircraft in proximity | Pilots and observers practice vigilance in detecting other aircraft in the sky, Radio Detection and Ranging (RADAR) is used to identify possible aircraft when UAV is BVLOS. Observers monitor and communicate with any other pilots on the ground. Active Aircraft Avoidance (AAA) is enabled when in autonomous flight. |
| Aircraft test flights in proximity of critical infrastructure and bystanders | Test flights for vertical flight are to be carried out at enclosed fields and horizontal flights are to be conducted in a designated airfield outside of the control zones. Appropriate signage is used to alert the public of aircraft activity and to maintain a safe distance. The aircraft is not approached by UBC UAS members until the motors have fully stopped and are disarmed. The crew is appropriately briefed before flights. |
| Improper test setups | Persons who carry out testing are appropriately trained for all apparatus involved or are under the guidance of someone who is. Apparatus is used as intended. |
| Obstacles encountered in flight | Pilot practices vigilance in identifying obstacles in the air. Observers perform continuous site surveys to identify obstacles. RADAR is used to identify obstacles when UAV is BVLOS. |
| Objects falling out of aircraft | Before a flight, UBC UAS members carry out thorough installation evaluation, including vehicle shake tests for loose objects and pre-flight checklists. |
| Loss of communication with aircraft | There is a failsafe function on the aircraft to activate safe and swift landing in case of communication loss. |
| Aircraft crashes | Operation is carried out in open areas where possible where potential fires can be contained, and damage to the surroundings is minimal. Crew are trained with putting out a fire and securing surroundings in case of a crash. |
| Winch mechanism releases rover early | To prevent damage from early release of the rover, tests will be conducted in an open area with special attention to overflown areas being free of bystanders/infrastructure. |

**Figure 15.** Mission Risks