

Designing an Autonomous Aerial Robot for Successful Operations in a Non-Deterministic Indoor Environment

Abdullah Almosalami

Department of Computer Science and Mathematics, North Dakota State University

Tyler Blanchard and Brady Goenner

Department of Mechanical Engineering, North Dakota State University

Andrew Jones and Jeremy Straub

Department of Computer Science, North Dakota State University

ABSTRACT

This paper provides an overview of the design of an aerial robot (also commonly known as an unmanned aerial vehicle or UAV) to meet the goals identified for Mission 7 of the International Aerial Robotics Competition. This mission requires the robot to engage in a herding application to influence ground robots to exit the playing field via a designated side. To meet this challenge, a hardware-software system was developed which implemented all elements required for flight, as well as software for target identification, decision making, and path planning. In this paper, particular focus is placed on the solutions related to the guidance, navigation, control, and autonomy requirements. An overview of the systems of the aerial robot is presented and the solutions developed for the mission challenges are discussed.

1. INTRODUCTION

Despite the many applications where teleoperation (or remote control) has been successfully utilized a myriad of limitations exist when having a human operator directly in control of an aerial robot or unmanned aerial vehicle (UAV). These human limitations translate into operational limitations for the controlled robot. More problematically, a UAV that is designed for teleoperation must still have a robust autonomous control system to take over in the event of communications link failure. Thus, with teleoperation, the robot is relying on the awareness, capabilities and competence of its operator, which may underperform an onboard decision making system which does not have link delay, human reaction time delays and other issues present with the operator.

The development and operations of UAVs and, in particular, small UAVs (sUAVs) are demonstrably increasing in autonomy, despite operating limitations imposed by current FAA regulations in the United States and similar restrictions imposed in other areas of the world. On many UAVs, most low-level tasks are already handled without operator involvement. Even with teleoperation-based systems, operators are assisted in flight control by craft stabilization and other automated systems. UAV systems can, and in many cases are, teleoperated like the pilot is onboard an aircraft. However, many other UAVs are commanded using goal identification where only the highest level commands are provided by the user and these commands are autonomously translated into lower-level commands that control the aircraft by an autonomous command and control system. For example, several commercial UAVs allow flight plans to be defined using waypoints and translate this into craft flight between them.

The current level autonomous control systems have limits. Many of these systems are heavily reliant on Global Positioning System (GPS) technologies, limiting their use inside buildings or in GPS-denied environments. Other systems are heavily dependent on a priori-known maps, which may not be available for many applications. The development of autonomous command and control systems that do not have these limitations is an open problem and one of the core challenges posed by the International Aerial Robotics Competition's (IARC's) Mission 7.

This paper presents an overview of the North Dakota State University team's aerial robot design and a discussion of its operations. In this section, the technical problems presented by the competition are, next, discussed. Then an overview of the solution is presented and goals for the team's participation in this year's competition are briefly discussed.

a. Problem Statement

Mission 7 presents three primary challenges. First, the UAV must interact with and avoid numerous moving targets. Second, it must navigate in a changing environment without use of GPS, simultaneous localization and mapping (SLAM), or external references. Third, the potential interaction between drones must be managed. All of the foregoing must be performed autonomously.

Those behaviors are demonstrated by having the UAV herd ten ground robots towards one side of the competition arena within a ten minute time constraint. These ground robots, while starting from a central location, disperse and move in fairly random trajectories. Four moving ground obstacle robots are also placed in the ring that the UAV must avoid. Collisions between robots on the ground further randomize the robot trajectories.

A limited method for controlling the movement of the ground robots is provided. The UAV can either land on the robot, causing a 45-degree turn, or position itself in front of the ground robot to cause a 90-degree turn.

Finally, in addition to these challenges, in Mission 7b these tasks must be performed while avoiding collision and competing with another drone in the arena.

b. Conceptual Solution

This section provides an overview of the hardware and software solution developed in response to the Mission 7 requirements. Each of the hardware and software elements discussed is elaborated on in subsequent sections of the paper. An overview of the system architecture is provided in Figure 1.

In developing the hardware system, it is important to note that a significant portion of the mechanical design accounted for mass and risk reduction and flight stability. The mechanical design includes a lightweight aluminum primary frame, a propeller system, shock absorbing landing gear, propeller guards and a barrier system designed for interaction with the ground vehicles. The design of the electronic system focused on providing sufficient power to allow for a ten minute flight time while, through this time, providing adequate power to allow sufficient thrust to be provided by the motors for lift and aerial maneuvers.

All of the computing is done onboard with an Erle Brain 3 flight computer (FC), which is based on the Raspberry Pi 3 single-board computer. Basic flight control is handled through the communication between the Erle Brain and the QBrain electronic speed controller (ESC).

Decision making for controlling flight operations is performed by software running on the Erle Brain 3 computer. Using data from an inertial measurement unit (IMU), visible light cameras and an optical flow sensor, through odometric techniques and applying an extended Kalman filter (EKF) - which is able to account for uncertainties and biases - the UAV is able to position itself relative to an initial position, without the use of GPS or external reference points. Through that positioning, an internal kinematic model of the UAV is used to project position and state information.

Ground robot detection and tracking is accomplished through computer vision algorithms applied on input from an onboard camera. Obstacle avoidance is performed using data from the cameras and distance sensors. Using all of the foregoing information, an intelligent agent within the software system performs strategic decision making and path planning to implement the defined competition strategy. Further details are provided in subsequent sections.

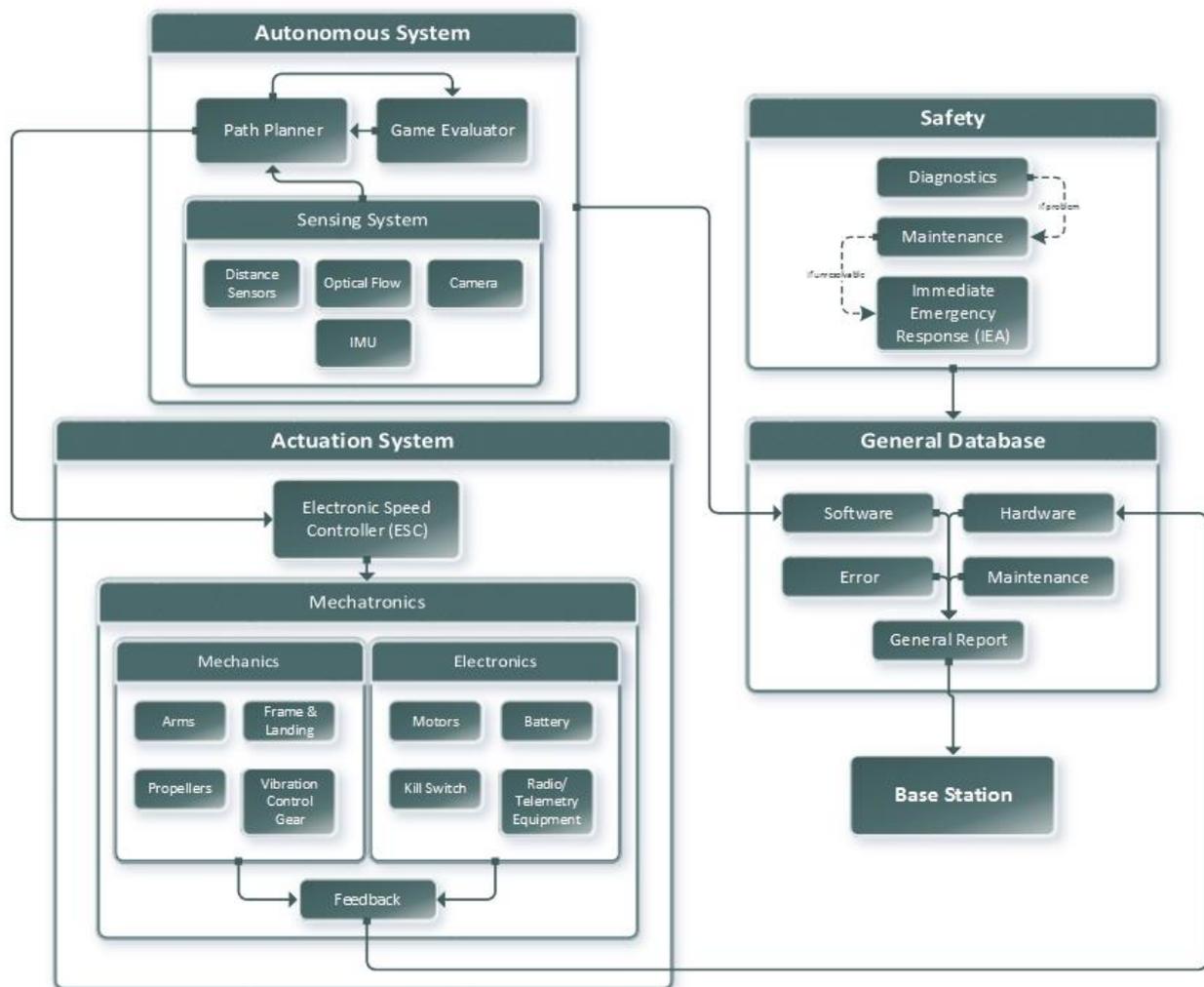


Figure 1. System Architecture.

c. Yearly Milestones

As this is North Dakota State University's first time participating in the IARC, the goals for this year's participation included the development of all hardware and software elements required to meet competition requirements. A suitable testing environment and hardware and software systems for UAV testing also had to be developed.

2. AERIAL VEHICLE

The aerial vehicle system is comprised of both hardware and software components. Hardware components include the primary chassis, which is comprised of shock-absorbing landing gear, lightweight aluminum links that support the rotor units and a housing for the electronic systems. A picture of the UAV, with the propeller guard, ground robot interaction unit and rotors removed is shown in Figure 2.



Figure 2. Chassis of UAV with key electromechanical components labeled.

The key electronic systems include the Erle Brain 3 FC, the QBrain electronic speed controller, the radio system, the rotor motors, the battery and the sensors. On the software side, key software components include the autonomous decision making system, the actuation control system, the safety system, the general database and software on the base station. The autonomous decision making system (ADMS) handles all decision making and control, and acts as the brain of the system. It is depicted in Figure 3.

Based on mission goals identified by the ADMS (using sensor system input), the path planner system (PPS) determines a path. This is provided to the actuation control system (ACS) which translates this into high-level commands and then low-level signals that are sent to the motor controllers. Both systems provide information to the general database (GD) that stores information on the state of the systems of the UAV. The safety assurance system (SAS) constantly scans the GD looking for readings that may suggest an anomaly and, if one is found, it runs diagnostics and maintenance routines. Should the need arise, the SAS is also responsible to perform immediate emergency action response (IEAR) activities. The SAS updates the GD, as needed. The GD outputs a systems status report to the base station (BS) for human review. Note that the BS does not have the capability to send commands to the UAV, as all needed resources are on the UAV itself. The BS is used solely for testing and informational purposes.

A key role of the ADMS is to determine how to achieve mission goals. As part of this process, the ADMS selects targets (i.e., the ground robots to be herded) to interact with and defines the type of interaction (land on or block). Once a target has been selected, its position is determined, relative to the UAV. Drone and target kinematic models are made and, combined

form a system kinematic model (SKM). The SKM is used as part of the path planning process.

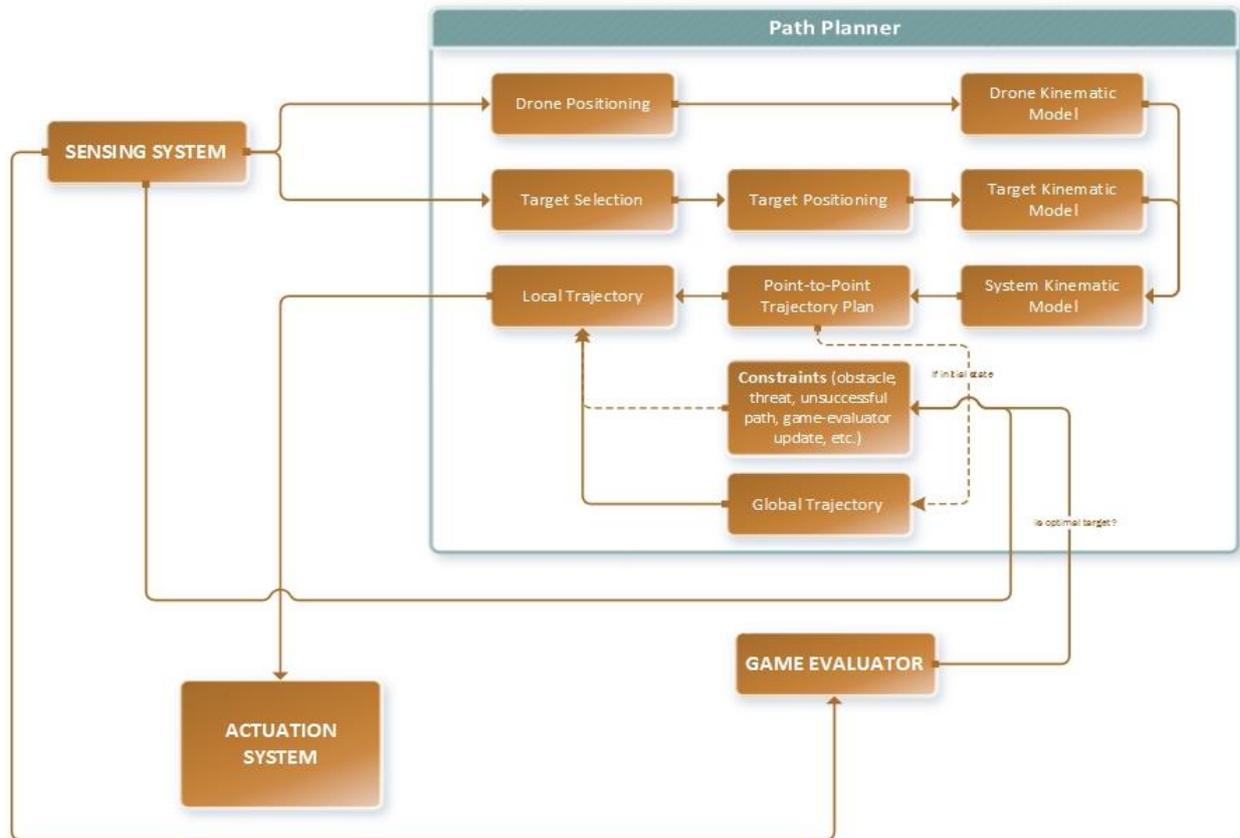


Figure 3. Autonomous decision making system overview.

For planning purposes, two mission scopes are used. A long-term mission guides the overarching activities of the system. The long-term mission goal is herding as many ground robots as possible to the designated area, within the time limitation. This long-term mission is accomplished by defining numerous several short-term missions. In this case, each short-term mission is typically herding a selected ground robot to a desired location. The game evaluator (GE), a component of the ADMS, is responsible for defining these short-term missions.

With the short-term mission defined, a trajectory for completing this mission is determined. This global trajectory (GT) defines the general path the UAV must follow to complete the mission. It is updated with numerous local trajectories (LTs) which define the immediate steps to achieve the GT, for a given time interval. The LT path may differ from the GT due to unexpected movement constraints such as obstacles, target movement and unsuccessful path-following performance. The GT may also be changed based on an update from the GE. Once the LT has been determined, it is outputted to the ACS and performed. Sensors are used to determine how close the path taken matches the desired path (and update state information in the GD). Unless the long-term mission is complete, the whole decision making loop begins again.

Propulsion and Lift System

As shown in Figure 2, an X-frame type quadcopter has been developed. The selection of this X-frame designed provides both agility and balanced mobility. The design consists of two sets of counter rotating rotors and utilizes four 2216-800KV motors to provide lift along with pitch

(longitude) and roll (lateral) control.

To achieve the ten-minute target flight time, a 5500 mAh lithium polymer battery has been used (this is discussed further, subsequently). Four 11 in carbon fiber propellers are utilized to provide thrust. Carbon fiber propellers were selected over the more common plastic variants as they, unlike plastic, don't flex and lose energy when in motion.

Guidance, Navigation, and Control

The guidance, navigation and control (GNC) functionality is provided by the stability augmentation system and navigation system. Each is now discussed. Figure 4 provides an overview of the GNC system.

Stability Augmentation System

Several design decisions have been made to provide aerodynamic stability. The UAV includes a triple-axis gyroscope on-board the Erle Brain FC for flight stabilization. The FC coprocessor provides real time response to external disturbances. Through repeated testing, calibration, and machine learning training, the UAV's stability has been attained.

Mechanically, the quadrotor has an x-copter configuration, which provides flexibility in control. The weight distribution of the UAV is such that the center of mass is at the geometric center of the UAV in the x-y plane and below the propellers in the z-axis. The SAS (described in more detail, subsequently) detects and handles stability and other disturbances. The risk reduction section discusses the vibration reduction measures which further aid in achieving the aerodynamic stability of the UAV.

Navigation

The proposed design uses the Erle Brain flight computer for motor control and navigation. The software utilizes the Robot Operating System (ROS) GNC program interface framework. The guidance system, based on [1], uses the state space model defined by the equations:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k(t)) + \mathbf{G}(t)\mathbf{u}(t) \quad (1)$$

$$\mathbf{y}_k = \mathbf{H}(\mathbf{x}_k(t)) \quad (2)$$

with the state vector:

$$\mathbf{x}(t) = [x \ y \ z \ v_x \ v_y \ v_z \ \theta \ \phi \ \psi \ \omega_x \ \omega_y \ \omega_z]^T \quad (3)$$

for the quadrotor position, heading, and velocities. The position describes the distance to the desired waypoint, determined from the camera sensor. The control input of the quadrotor is described as angular velocities and linear thrust. The input from the sensors help the robot maintain stability, and can be described as:

$$\mathbf{y}_k = [z \ v_x \ v_y \ \alpha_x \ \alpha_y \ \alpha_z \ a_x \ a_y \ a_z]^T \quad (4)$$

Within the EKF, the covariance matrix is updated using the discrete Lyapunov equation:

$$\mathbf{P}_k^- = \mathbf{F}\mathbf{P}_{k-1}\mathbf{F}^T + \mathbf{Q} \quad (5)$$

The state and covariance matrix are then used to update the Kalman gain:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R})^{-1} \quad (6)$$

The state update then uses the nonlinear function, $h(\mathbf{x}_k)$, of the current state to reduce the uncertainty issues with a linearized Kalman filter:

$$\mathbf{x}_k^+ = \mathbf{x}_k^- + \mathbf{K}_k (\mathbf{z}_k - \mathbf{y}_k) \quad (7)$$

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \quad (8)$$

The EKF receives waypoint data from the camera system to track the ground robot as it navigates along the ground, providing a control system input (as shown in Figure 4). The ground robot's estimated distance from the quadrotor is continuously updated, as it is a dynamic waypoint. A separate proportional integral derivative (PID) control loop is used for controlling the pitch of the robot to maintain view of the ground robot.

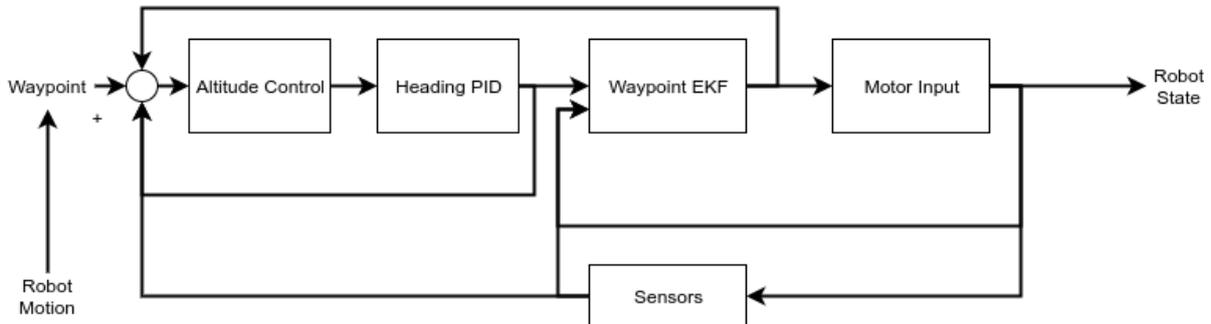


Figure 4. Diagram of control system architecture.

Flight Termination System

The flight termination system is a key component of the SAS. This section further elaborates on its functionality. The goal of the SAS is to allow the UAV to be able to solve its own problems, while providing the option for a human user to take control, if needed.

The SAS runs recurring analysis and diagnostics using the data stored in the software and hardware databases. It detects anomalies based on a set of preloaded anomaly signatures, as well as noting deviations from expected behavior. An anomaly is defined to be any state of the system that deviates from what is expected. An anomaly signature is a dataset of values that signifies an anomaly is present. For example, if the UAV were restrained and its flight obstructed, this would produce kinematic and motor current draw values indicative of this problem. After performing experimentation, recurrently, to obtain statistically significant results, an anomaly signature was established for multiple potential issues and preloaded into the diagnostics system for use.

Once the diagnostics system detects an anomaly, it passes this information to the maintenance system, which is responsible for attempting to resolve anomalies. If the anomaly is resolvable (e.g., quick self-calibration can be performed by the UAV), the maintenance system performs the appropriate resolution. If not (e.g., there is irreparable mechanical damage), the maintenance system triggers IEAR system.

The IEAR system's protocol is, first, to transfer control to the BS and await an adequate response while constantly evaluating if the control was enough to solve the anomaly. If

unsuccessful (e.g., lack of a timely response, failure to resolve the problem even via remote control), power to the motors is automatically cut and the drone shuts down and lands. Given the capabilities of the landing gear, from operating heights this shouldn't result in any harm to the UAV or its surrounding environment.

The UAV also has a 'remote kill switch' that may be activated from the BS. The 'kill switch' cuts power to the motors immediately upon the UAV receiving the command. The 'kill switch' was implemented to follow the specifications provided by the IARC. There is also a command that can be set from the BS that immediately shifts control of the UAV from its autonomous control to the BS, providing remote control of the UAV (i.e., a remote override).

PAYLOAD

Sensor Suite

Guidance, Navigation and Control

The control system maintains stability using a 3-axis accelerometer, and a 3-axis gyroscope the data from which is processed by the FC coprocessor. The inertial data, along with data from the PX4FLOW (flow-sensing) unit, is sent to the Raspberry Pi for waypoint control of the quadrotor. The EKF runs in a separate node within ROS providing control input for waypoint acquisition. The PX4FLOW provides local sensor input along with maintaining relative position when required.

Mission Sensors

The main mission-performance sensor is the mounted camera on the quadrotor for ground robot detection and tracking. The camera is connected to the Raspberry Pi, and processed using custom software and the OpenCV framework in order to monitor and predict the movement/location of each target ground robot.

Target Identification

In order to accurately land on (or in front of) a moving ground robot, the location of the target relative to the aerial vehicle must be determined. For this mission, the target robots are circular and have a color that is distinct from the floor. With this prior knowledge, a simple circle Hough transform [2] is feasible to detect the outline of each robot in view of a camera. This detected circle yields relative coordinates of the target as well as the radius.

$$\begin{aligned}x &= a + \cos(\theta) \\y &= b + r\sin(\theta)\end{aligned}$$

This step is implemented using the built-in-to-OpenCV Hough circle transform function, which uses the Hough gradient method to reduce computational complexity. To reduce noise, the input into the function is an image filtered by the color of a target ground robot. The smaller inner circle feature of the ground robots is pruned through defining a minimum radius, adjusted based on perceived altitude. Subsequently, the distance to the target is (partially) determined by comparing its detected radius to the measured radius of the robot, known a-priori.

The motion of detected ground targets is calculated based on reference to identified features in the environment, such as the grid markings, and multiple observations of the same target over time (see Figure 5a). The grid markings and other features, are used as a temporary point of reference (as shown in Figure 5b) to facilitate the comparison of to determine the changes in ground robot position and orientation that occur over multiple observations.

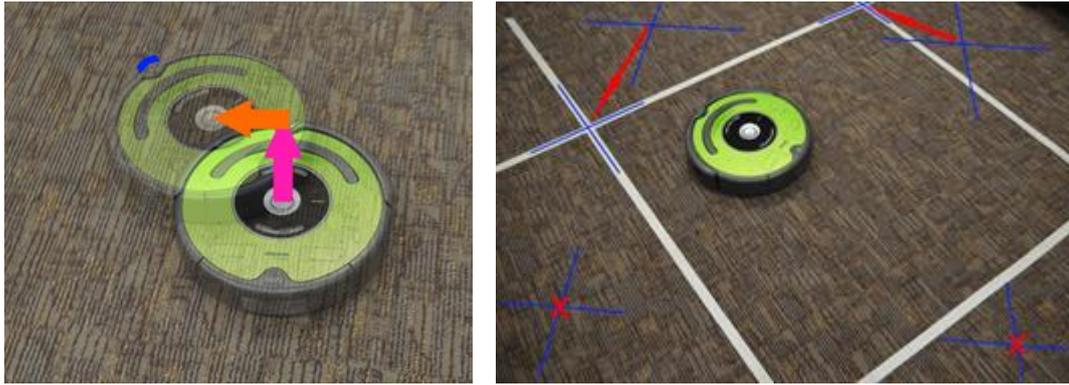


Figure 5. (a) Showing detection of movement and orientation change of the ground robot, left, and (b) showing movement of the imaging position, right.

Threat Avoidance

The system uses ultrasonic sensors mounted to the quadcopter for threat detection. Once the sensors detect a threat, appropriate constraints are applied in path planning to avoid the obstacle or other threat.

Communications

The FC communicates with the Raspberry Pi single board computer through the GPIO pins located on the board, allowing for other sensor devices to access the UART and I2C pins. The PX4FLOW connects to the Raspberry Pi through the UART port on the FCU connecting the Raspberry Pi. The Raspberry Pi is interfaced to the GS through a Wi-Fi network that uses the Secure Shell (SSH) protocol to secure communications.

Power Management System

In determining the power requirements for the UAV, certain initial considerations were established: the UAV will weigh approximately 1.5 kg, the voltage needs to be around 12V for the average motor, the UAV needs to be able to fly for at least 10 minutes, and the battery will be a Lithium-Polymer rechargeable battery. Given that, additional constraints on the current draw of the system were determined and the needed thrust per motor was determined using the the following equation:

$$I_{MAX\ Av.}(C, T) \leq [(C*60)/T]*0.9$$

Where $I_{MAX\ Av.}$ is the maximum total average current draw of the system, and T is the amount of time the system needs to last. The 0.9 is because the available capacity of the of Li-Po should not drop below 10% for safety and quality assurance. The 60 is to convert the capacity to Amp-Minutes instead of the usually Amp-Hours unit, as T is in minutes. The \leq is to show that the total average current draw of the system can go up to but must not surpass the right side of the inequality. Assuming motors are the primary current consumers of the system, the equation:

$$I_{M-MAX\ Av.}(C, T, N) < I_{MAX\ Av.}(C, T)/N$$

Where $I_{M-MAX\ Av.}$ is the maximum average current draw of each motor and N is the number of motors in the system. Here, it is stated that the $I_{M-MAX\ Av.}$ must be less than the $I_{MAX\ Av.}$, as there are other components in the system that will draw current, albeit much less so.

To give more meaning to how the maximum average current draw of each motor relates to the maximum possible current draw of each motor (and accordingly have an idea of what that average looks like), the following equation is used:

$$R_{Av.-to-MAX}(C, T, N, I_{M-MAX}) = I_{M-MAX Av.}(C, T, N)/I_{M-MAX}$$

where I_{M-MAX} is the maximum possible current draw of the motor. Furthermore, the minimum thrust required per motor is:

$$T_M(m) \geq (m * g) / N$$

and given $T_M(I)$ (the thrust of a motor at a given current draw) is proportional to $\omega_M(I)$ (the speed at which the motor rotates at a given current draw):

$$T_M(I_{M-MAX Av.}) \geq T_M(m)$$

where T_M is thrust per motor, m is the mass of the system, and g is the acceleration due to gravity. If T_M is equivalent to the right side of the first inequality, then the system hovers, assuming all else has been accounted for. The second inequality indicates that when the current draw of the motor is at or below the $I_{M-MAX Av.}$, the thrust of that motor must be greater than or equal to $T_M(m)$. If at $I_{M-MAX Av.}$ the thrust is not at $T_M(m)$, then the system will fail to achieve the required flight time as the motors would have to draw more than $I_{M-MAX Av.}$ which will drain the battery's capacity faster.

It is clear from the above equations that a balance must be struck between T_M and $I_{M-MAX Av.}$. The eCalc tool was used to obtain data on motor-propeller pair and their likely thrust output. Using that data along with the current draw of the given motor, a visualization of the data (not included due to space constraints) showed all possible pairs. Then given the highest capacity 3C Li-Po that could be fit on the UAV is 5.5Ah in capacity ($C=5.5A * h$), and given the required time is 10 minutes ($T=10min$), the appropriate propulsion system outlined above in the corresponding section was selected.

OPERATIONS

This section will define how the drone is operated.

Flight Preparations

Before any flight, the following checklist is performed:

1. Affirm quality of components (e.g., balance of propellers, wiring, connections, adapters)
2. Check battery power
3. Power on quadcopter
4. Allow FCU to initialize and broadcast Wi-Fi
5. Login and initialize autonomous mode

Man/Machine Interface

The quadrotor interfaces through a Wi-Fi network to the BS and can be connected to through a SSH terminal. Through this, diagnostic information can be displayed (such sensor information) as well as the camera feed. The kill switch and manual override transceiver are located at the GS.

RISK REDUCTION

Significant consideration was given to risk reduction and crashworthiness in designing the UAV. Figure 6 presents the UAV's frame design. This frame not only protects the UAV and those around it, it provides the surface for the ground robots to interact with that is required for blocking-based turning of them.

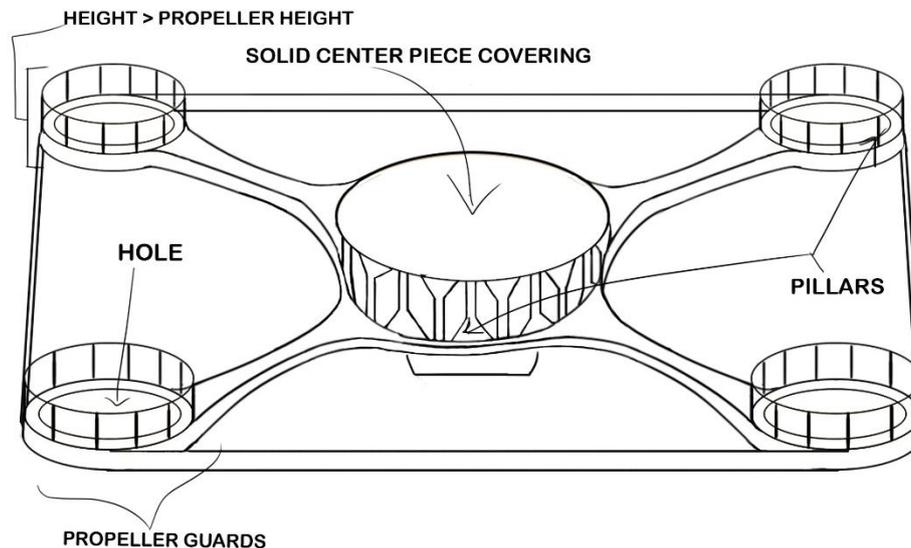


Figure 6. Diagram of aerial robot frame.

All components are guarded in casings and guards that protect them from the mechanical damage of potential collisions and also protect the environment from any potential harm the UAV may present. The UAV was initially 3D printed using PLA filament, but it was upgraded after collision testing. Now, the arms are made of aluminum bars, the surrounding guard is printed using polycarbonate filament, the central component casings are made from 3D-printed Carbon Fiber filament, and the landing gear are made of durable and flexible plastic. The materials were chosen due to the desired strength vs flexibility ratio of the region the material was protecting. The surrounding guards, for example, needed to be relatively strong but flexible to allow for shock absorption, hence the use of Polycarbonate. The inner casings needed to be strong and inflexible, hence the use of Carbon Fiber. The landing gear was chosen to be more flexible than rigid (but still durable) in order to absorb the impact of imperfect landings, hence the durable plastic. Furthermore, all components that needed to be out in the open (e.g., the sensors) were placed as internally as possible (surrounded by all the other components' protective casings) while still maintaining a functional position.

Vehicle Status

Shock/Vibration Isolation

Several Moongel dampeners have been placed throughout the frame design, as appropriate, (such as between the boards and the frame) to provide vibration absorption. All screw connections have thick rubber washers that add additional vibration absorption.

Furthermore, an important measure taken is ensuring the propellers are perfectly balanced before each flight (i.e., the center of mass of the propeller should be at the geometric center of the propeller). Unbalanced rotating propellers produce significant vibrations that propagate throughout the UAV. The propeller balancing is done by using a propeller balancer tool that allows visual testing as to where the center of mass lies. Any imbalances were fixed by sanding the appropriate side off until both sides were well balanced.

EMI/RFI Solutions

EMI/RFI refer to Electromagnetic Interference/Radio-Frequency Interference and involve signals that may disrupt the functionality of the electrical system. The electronics used have adequate EMI immunity and as such, EMI has not been a primary concern. An EMI filter has been incorporated in the electronics for redundancy. The operating radio-frequency will be tested beforehand to ensure no accidental signal interference and kept secret.

Safety

Ensuring quality obstacle detection and avoidance and anomaly resolution of the UAV is a fundamental priority, and so software-wise, a system has been implemented to protect the environment. Should the system fail for any reason, since the UAV is surrounded by a frame guard, and the propellers surrounded by prop-guards in the x-y-z planes, there is little risk that something from the environment may be harmed by the UAV.

Modeling and Simulation

The nonlinear control, described in [3], uses PID control for a nonlinear system quadrotor. The quadcopter is an under-actuated system. The states of linear components limit the maneuvers of the quadcopter but simplify the design. The use of a EKF filter will stabilize localization. To test the control loop design, the system was simulated in Gazebo with mock sensors. The simulation tests were performed to evaluate the stability of the EKF design as well as the state transitions. The use of simulated static walls was placed for threat detection and avoidance maneuvers.

Testing

The quadrotor control systems were simulated before being run on the real system. The EKF localization and control component were tested for their stability. The state transition for avoiding obstacles was simulated in Gazebo. The robot camera system was tested to track the ground robots and estimate the distance from the camera. Significant real-world testing has also been performed.

CONCLUSION

In conclusion, the UAV presented herein has been designed to accomplish the challenges set forth by IARC 2017 Mission 7a. Sensor data is used for positioning, obstacle detection, and target tracking, and that information is provided to an autonomous command and control system for path planning and game evaluation. In addition to that, the approach to safety and self-reliance of the UAV mechanically and software-wise is innovative and should perform well under competition conditions.

REFERENCES

- [1] Groves, P. Principles of GNSS, Inertial and Multisensor Integrated Navigation Systems. Norwood, MA: Artec House, 2013.
- [2] Pedersen, S. Circular hough transform. Aalborg Univ. Vision, Graph. 2007, 1–6.
- [3] Heba talla Mohamed Nabil ElKholy. Dynamic Modeling and Control of a Quadrotor Using Linear and Nonlinear Approaches, 2014, URL: http://dar.aucegypt.edu/bitstream/handle/10526/3965/Heba_ElKholy_Thesis_S2014.pdf?sequence=1
- [4] Tsourdos White Shunmugavel. Cooperative Path Planning of Unmanned Aerial Vehicles. New York, NY: Wiley, 2011