

Georgia Tech Team Entry for the 2012 AUVSI International Aerial Robotics Competition

Daniel Magree*, Dmitry Bershadsky†, Xo Wang‡, Pierre Valdez§, Jason Antico¶,
Ryan Coder||, Timothy Dyer,**Eohan George,††Eric N. Johnson‡‡
Georgia Institute of Technology, Atlanta, GA, 30332-0152, USA

This paper describes the details of a Quadrotor Unmanned Aerial Vehicle capable of exploring cluttered indoor areas without relying on any external navigational aids. A Simultaneous Localization and Mapping (SLAM) algorithm is used to fuse information from a laser range sensor, an inertial measurement unit, and an altitude sonar to provide relative position, velocity, and attitude information. A wall avoidance and guidance system is implemented to ensure that the vehicle explores maximum indoor area. A model reference adaptive control architecture is used to ensure stability and mitigation of uncertainties. Finally, an object detection system is implemented to identify target objects for retrieval. The vehicle is intended to be Georgia Tech Aerial Robotic Team's entry for the 2012 International Aerial Robotics Competition.

I. Introduction

The Army, Navy, and the Air Force have identified indoor reconnaissance and surveillance capability as a top research priority due to the changing nature of the battlefield. Miniature air vehicles are ideal candidates for such missions as they can use three dimensional maneuvers to overcome obstacles that cannot be overcome by ground vehicles. However, significant technological challenges exist in order to ensure reliable operation in such environments. Most current algorithms for Unmanned Aerial System (UAS) Guidance Navigation and Control rely heavily on GPS signals,^{1,2,3} and hence are not suitable for indoor navigation where GPS signal is normally not available. Furthermore, the indoor UAS must be sufficiently small in order to successfully navigate cluttered indoor environments, consequently limiting the amount of computational and sensory power that can be carried onboard the UAS. Finally, the UAS should be designed to be expendable due to the dangerous environments it needs to operate in, hence low-cost, low-weight designs need to be explored. These restrictions pose significant technological challenges for the design of reliable Miniature Air Vehicle (MAV) platforms capable of navigating cluttered areas in a GPS denied environment.

A. Problem Statement

The sixth mission of the International Aerial Robotics competition requires that a MAV weighing less than 1.5kg have the ability to enter and navigate within an unknown confined environment in search of a specific marked target without being detected. The mission also requires that the MAV locate and bring back a flash drive.

*Graduate Research Assistant, School of Aerospace Engineering

†Graduate Research Assistant, School of Aerospace Engineering

‡Student, School of Electrical Engineering

§Graduate Research Assistant, School of Aerospace Engineering

¶Graduate Research Assistant, School of Aerospace Engineering

||Graduate Research Assistant, School of Aerospace Engineering

**Graduate Research Assistant, School of Aerospace Engineering

††School of Electrical and Computer Engineering

‡‡Lockheed Martin Professor of Avionics Integration, School of Aerospace Engineering

B. Conceptual Solution

The Georgia Tech Aerial Robotics Team (GTAR) has developed an indoor MAV Unmanned Aerial System (UAS) capable of exploring cluttered indoor areas without relying on external navigational aids such as GPS. The MAV uses an off-the-shelf Quadrotor platform and is equipped with off-the-shelf avionics and sensor packages. We use an elaborate navigation algorithm that fuses information from a laser range sensor, inertial measurement unit, and sonar altitude sensor to form accurate estimate of the vehicle attitude, velocity, and position relative to indoor structures. We leverage the fact that all indoor structures have walls to design a guidance algorithm that detects and follows walls to ensure that the navigation solution maintains its fidelity and maximum indoor area is explored in a reasonable amount of time. We use a control architecture that augments a proven baseline proportional-derivative controller with an optional adaptive element that aids in mitigating modeling error and other system uncertainties.

C. Yearly Milestones

The GTAR Team aims to continue to develop a stable MAV capable of carrying an avionics payload for completing the 6th mission. The vehicles will have the ability to navigate and explore the indoor environment, detect and extract the flash drive. Subsequent yearly milestones include exploring the indoor environment without being detected and improving guidance efficiency. A simple velocity field method was employed in previous years and is now used as a backup guidance mode. A frontier-based guidance system was used in 2010 which forms the basis of the graph-based exploration system used in 2011 and 2012.

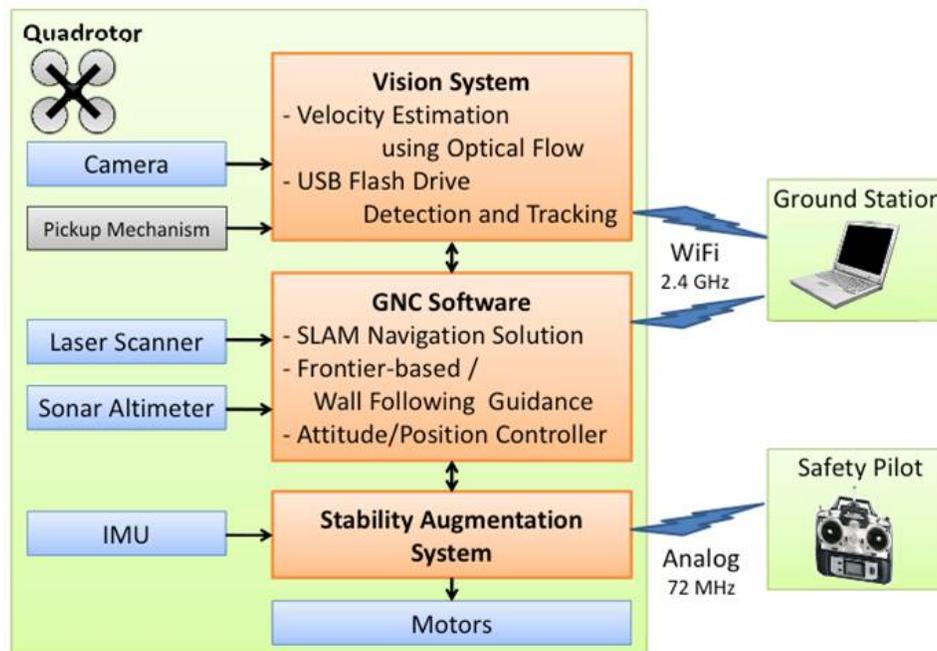


Figure 1. GTAR System Architecture

II. Description of Vehicle

A. Aerial Platform

Quadrotors have become a very popular choice for MAV due to their relatively high payload capacity and high maneuverability.⁴ Furthermore, unlike helicopters, Quadrotors avoid the use of mechanical parts for exerting moments and forces required for maneuvering. We selected the AscTec Pelican Quadrotor made by Ascending Technologies GmbH as the base airframe (see Fig.2(a)). The vehicle structure, motors, and rotors of AscTec Pelican were used without modification. Two such vehicles (GTQ1 and GTQ2) are being developed for the 2012 competition. The vehicle generates lift using four fixed pitch propellers driven by

electric motors. Control is achieved by creating a relative thrust offset between the propellers. Quadrotors can either be flown with diamond configuration (front, right, back, left motors are used to effect pitching, rolling, and yawing motion) or square configuration (front-right, back-right, back-left, front-left motors are used to effect pitching, rolling, and yawing motion). Although many Quadrotors in aerial robotics community fly with diamond configuration,^{6,7,4} we selected the square configuration (shown in Fig 2(b)) to allow for more flexibility in sensor mounting locations.

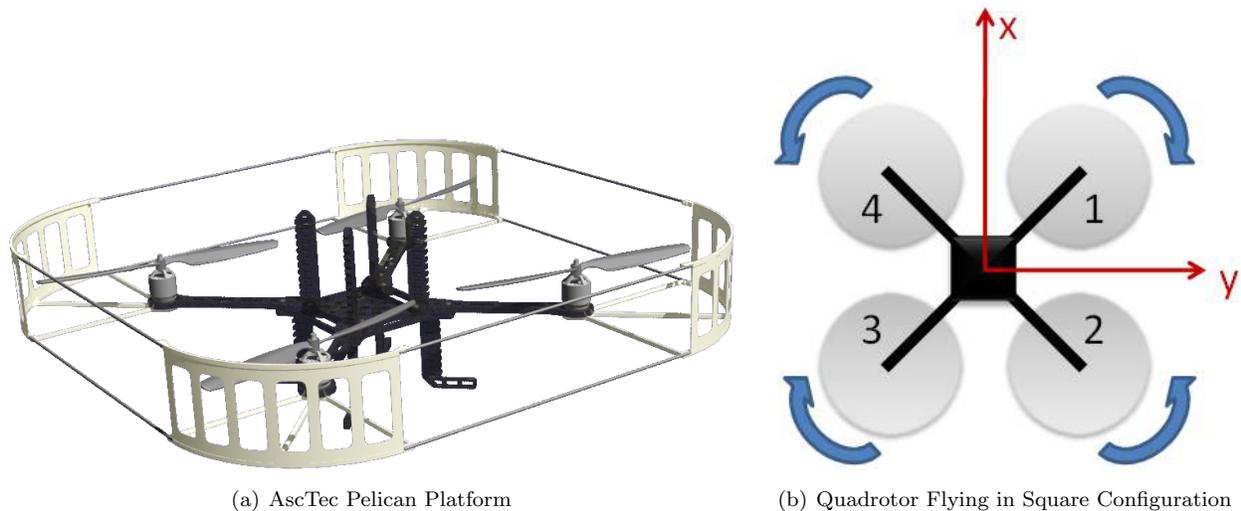


Figure 2. The 2012 GTAR Team chose to use the Ascending Technologies Pelican as aerial platform.

GTQ3 was initially designed as a risk mitigation vehicle. With it, the team would have access to a heterogeneous platform, reducing the probability of common mode failures via implementation of different avionics and hardware than those on GTQ1 and GTQ2. The vehicle's chassis was designed from scratch around the required avionics, chosen independently of those of the Pelican vehicles (see Fig.A). The vehicle was designed with ease of maintenance, and both weight and cost savings in mind. To save on material costs, a cheap, 5mm wood ply is used for the majority of the structural components. A 4'x8' sheet provides enough material to create about 20 vehicles. Carbon fiber rods, the most expensive structural components, provide an adjustable motor base. The components are laser cut in-house to keep labor costs at zero. Instead of screws, hot glue is used to keep the structure together and interlocking parts provide structural rigidity. Hot glue is relatively easy to adjust with the use of a heat gun, allowing for quick repairs and adjustments. The avionics platforms are mechanically independent of the vehicle frame and are adjustable in height.

The vehicle weighs about 0.85kg with the following avionics and equipment: four motors, propellers, and speed controllers, Xbee transceiver, Copter Control SAS, wiring harness, receiver, and power regulator. A 0.38kg 5000mah 3C battery allows for approximately 10 minutes of flight while the motors provide over 4kg of thrust for a payload capacity of around 3kg. An I2C to PWM converter allows use of the GTQ1/2 SAS board. The vehicle weighs about 1.25kg with the above battery, which leaves about 0.25kg for a LIDAR system (0.16g), camera (70g), and a Gumstix board for GNC software.

B. Avionics

The GTQ avionics are designed to be entirely independent of the vehicle ground station. All processing is performed onboard the vehicle, which eliminates the dependency on a strong wireless communication link to a remote computer. The onboard computer which best fit the requirements of small size and large processing power was a Core 2 Duo 1.86 GHz processor in the 95mm x 95mm COMExpress format. Additionally, an interface board was custom designed for the vehicle sensors. This computer handles both the vision system and GNC software (see Fig. 1). The stability augmentation system is handled by an ATMega128 microcontroller.



Figure 3. The GTQ3 Platform.

C. Guidance Navigation and Control System

1. Guidance System

The developed guidance system uses only locally available information gathered through the onboard sensors, which include a laser range scanner. The system maintains previous tree-based guidance systems as backup for the newly developed graph based frontier exploration system. Scan frontier points are used to add nodes to the undirected exploration graph at each scan point. A node is added (blue empty points, Fig.4(b)) as the midpoint of each independent frontier of at least a particular arclength. Each new node is checked against existing nodes. Should a node exist within a specified cartesian distance from the new node, a node will not be added, but a graph connection (brown lines Fig.4(b)) will be made between the existing node and the node at the current scan point.

As the vehicle captures a node (blue, green-filled points Fig.4(b)), a new scan is performed and the next node is chosen based on a weighting system. Distance to the node, the arclength of the frontier used to create it, and a directional bias are used to select the next target highlighted in red. The intended graph path is highlighted as well. The directional bias is used should multiple vehicles be operating at the same time as an exploration efficiency enhancement. Simulated truth results are shown in Fig.4(a) where the yellow path traces the vehicle's position over time. The vehicle has travelled approximately 120' in the figure, showing about 60 seconds of exploration.

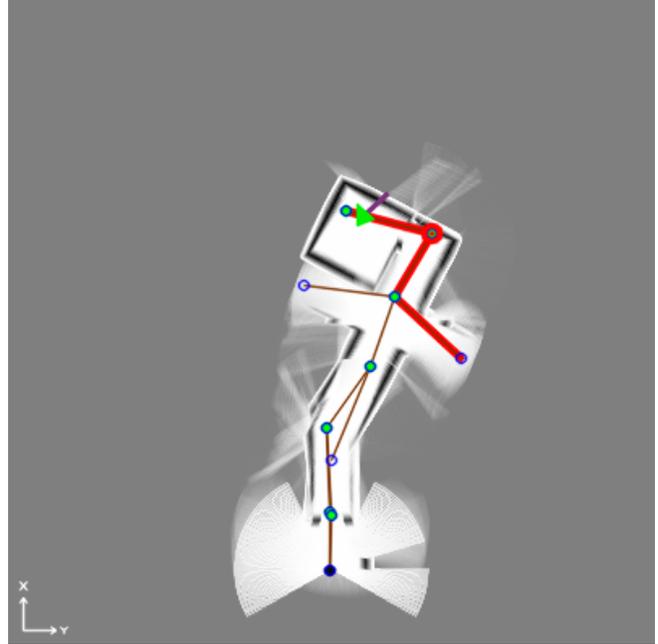
Should the vehicle not reach the target node in a specified amount of time, or should the GNC software detect that the vehicle is stuck in an unfavorable position, the vehicle will be commanded to backtrack to the last reached node. Stuck detection works by applying three criteria, all of which increase the confidence in the stuck detector output. Firstly, the mean velocity of the vehicle over a rolling time window is calculated and checked against a threshold set in the software. If the velocity is below this value, the probability of a stuck detection is increased. Secondly, in that same time window, should the velocity of the vehicle flip on itself more than a number of allowable times, the probability is yet again increased. Lastly, if the vehicle does not leave a circle of a specified radius within the time window but has traveled a specified distance, the stuck probability is again increased. Timeout is forced should probability reach a specified value.

2. Stability Augmentation System (SAS)

The Quadrotor platform is inherently unstable, that is, without control inputs, the platform would enter an uncontrolled drift in velocity and angular rates and collide with the ground or nearby obstacles. Quadrotors are also known to be notoriously hard to control even for human pilots, particularly because the relationship between thrust and stick deflection is nonlinear and because attitude is coupled heavily with velocity. Hence,



(a) Simulated truth map and vehicle exploration path



(b) Simulated SLAM solution and exploration graph overlay

Figure 4. Graph based guidance in the simulated 2011 competition arena

it is desirable to integrate angular rate damping to aid the pilot in controlling the Quadrotor. Let \hat{p} , \hat{q} , and \hat{r} denote the gyroscope measurements of the Quadrotor roll, pitch, and yaw rates, and δ_{ϕ_p} , δ_{θ_p} , and δ_{ψ_p} denote the pilot roll, pitch, and yaw stick deflections, then the actual stick deflection commands are assigned using the following proportional control logic:

$$\delta_{\phi} = \delta_{\phi_p} - K_p \hat{p}, \quad (1)$$

$$\delta_{\theta} = \delta_{\theta_p} - K_q \hat{q}, \quad (2)$$

$$\delta_{\psi} = \delta_{\psi_p} - K_r \hat{r}. \quad (3)$$

In equation 1, K_p , K_q , and K_r denote the linear gains chosen to provide appropriate rate damping.

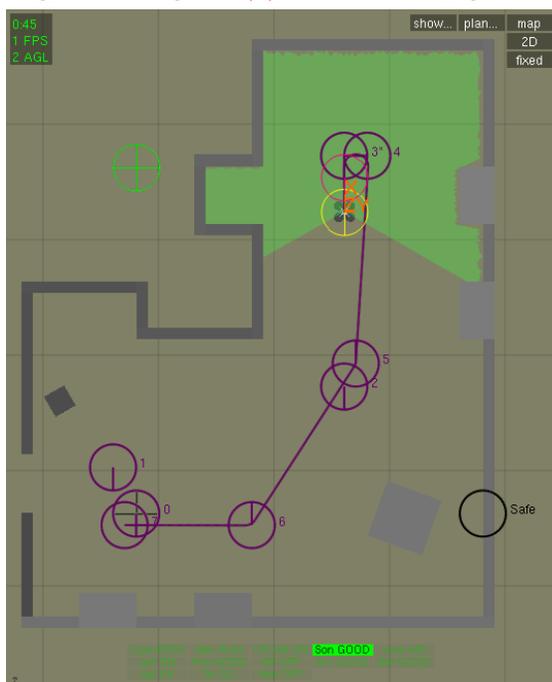
3. Control Algorithm

The complexity of the control system depends not only on the quantities being controlled, but also on the dynamics of the system itself. Unlike ground vehicles, unstable air vehicles are susceptible to oscillation and divergent flight when the control system is not properly tuned. Even for stable flying vehicles, coupling between lateral and longitudinal motion as well as aerodynamic interaction with the environment must be considered. The control architecture used by the GTAR 2011 team leverages the proven Model Reference Adaptive Control architecture developed for control of VTOL UAS throughout their flight envelop by Georgia Tech UAV Research Facility.^{8,9,10} In this architecture, a position control loop generates a velocity command, a velocity control loop generates an attitude command, and an attitude control loop generates servo commands to stabilize the vehicle by controlling the angular rate. Kannan has shown that such nested and cascaded control loop architecture with actuator saturation can indeed be used to control VTOL UAS.⁹ This system of nested control loops requires that the vehicle maintain an estimate of its position, velocity, attitude, and angular rate. For details the reader is referred to references 11, 8, 9, 12 for further details.

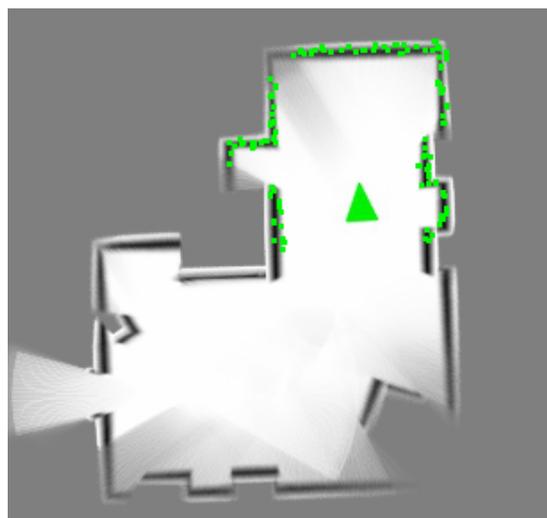
4. Navigation Algorithm

A variety of SLAM algorithm implementations are available for free use at the web site OpenSLAM.org. The algorithm used for the preliminary research, called CoreSLAM,¹³ was chosen primarily because it is simple, easy to implement, and it uses integer math where possible to improve computational speed.¹⁴ There are two main parts to any SLAM routine. The first task is to measure distance to obstacles or landmarks in the environment, and to map them given the vehicle's position and orientation (i.e. mapping). The second task is to determine the best estimate of the vehicle's position and orientation based on the latest scan (or series of scans) given a stored map (i.e. localization). The mapping and localization tasks are performed together to maintain the most current map and position estimate. The GTAR team has developed an Iterative Closest Point (ICP) scan-matching algorithm in-house. The algorithm were developed to prioritize localization with respect to the immediate environment, and place lesser emphasis on building and maintaining highly accurate global maps. Hence the "map" in the ICP algorithm consists entirely of a single previous scan. However, with the ICP algorithm individual scans can be saved and post-processed into a global map if desired.

Figure 5(a) shows a simulated building interior being explored by the Quadrotor with a scanning laser rangefinder. Figure 5(b) shows the map generated during a simulated flight.



(a) Mapping a simulated environment



(b) Map generated during simulation

Figure 5. The map maintained by the CoreSLAM routine represents an occupancy grid, where the value of each pixel in the image represents the likelihood that a particular grid square is occupied. Here, lighter colors represent free space, while darker colors represent obstacles and medium gray areas are unexplored. Areas with higher contrast represent greater certainty due to longer observation periods during the flight. The green triangle represents the vehicle's estimated position and heading.

The inherent nonlinearities in the vehicle dynamics and the measurement sensors are handled through the use of an Extended Kalman Filter (EKF). An existing EKF-based navigation filter architecture developed at the Georgia Tech UAV Research Facility is utilized as part of the indoor navigation system.² The navigation algorithm developed by the GTAR team augments the existing EKF based architecture to function without GPS signals by using the range information obtained from the laser range scanner and the sonar altimeter.¹⁵

D. Memory Stick Detection and Retrieval

Memory stick detection and retrieval is achieved with a combination of open-source software and an original tracking algorithm optimized for use on computers with low computational power. The object detection algorithm is part of the open-source software package OpenCV. The detector first is trained using a series

of images of the target object, called the positive image set. The positive image set can be created from multiple images of the object, or from a single image that is artificially distorted to simulate viewing the object from many angles. Approximately 1000 images were used. Additionally, a set of 1000 images without the object, called the negative set, is passed to the function. The output of the training algorithm is a cascade of Haar classifiers that can be used to efficiently identify the object in a sample image. Generating the classifier typically took 8 hours of processing time on a standard desktop computer with a Core 2 Duo processor. Processing an image with the classifier could be performed in 0.2 seconds on the GTQ onboard computer.

Once the target memory stick is located, a specialized guidance system generates a trajectory for the vehicle for landing on and retrieving the memory stick. The guidance system estimates the three-dimensional location of the memory stick using a downward-facing camera mounted on the vehicle. A cascade filter identifies the pixel location and pixel area of the memory stick in each image from the camera, and an EKF estimates the three-dimensional location based on the pixel measurements. The vehicle is then commanded to descend onto the location of the memory stick and retrieve it. One preliminary retrieval mechanism design is a pad of adhesive tape on the bottom of the vehicle.

Let the camera frame coordinates of the memory stick be given by $\bar{x}_c = [XYZ]^T$. Measurements from the camera are the square-root of the pixel area, A_{sqrtp} , and pixel location, x_p and y_p , and define measurement vector $\bar{y} = [A_{sqrtp} x_p y_p]^T$. The equations for the measurement model are given by

$$A_{sqrtp} = \frac{k}{X} \sqrt{A_{ft}} \quad (4)$$

$$x_p = \frac{k}{X} Y \quad (5)$$

$$y_p = \frac{k}{X} Z \quad (6)$$

where A_{ft} is the area of the memory stick in feet and k is the pixel to radian scale factor, both known constants. The linearized measurement model is generated by taking the first order Taylor series expansion of the previous equations.

$$\frac{\partial \bar{y}}{\partial \bar{x}_c} = \begin{bmatrix} -\frac{k}{X^2} \sqrt{A_{ft}} & 0 & 0 \\ -\frac{k}{X^2} Y & \frac{k}{X} & 0 \\ -\frac{k}{X^2} Z & 0 & \frac{k}{X} \end{bmatrix} \quad (7)$$

Converting to the inertial frame,

$$\frac{\partial \bar{y}}{\partial \bar{x}} = \frac{\partial \bar{y}}{\partial \bar{x}_c} \frac{\partial \bar{x}_c}{\partial \bar{x}} = \frac{\partial \bar{y}}{\partial \bar{x}_c} L_{ci} = \mathbf{C} \quad (8)$$

where L_{ci} is the rotation matrix converting from the cameraframe to the inertial frame. Inserting matrix \mathbf{C} into the EKF measurement update gives the corrected state.

$$\mathbf{K} = \mathbf{P}^- \mathbf{C}^T (\mathbf{C} \mathbf{P}^- \mathbf{C}^T + \mathbf{R})^{-1} \quad (9)$$

$$\hat{x} = \hat{x}^- + \mathbf{K} [\bar{y} - h(\hat{x}^-)] \quad (10)$$

$$\mathbf{P} = (\mathbf{I} - \mathbf{K} \mathbf{C}) \mathbf{P}^- \quad (11)$$

where \mathbf{R} is the measurement covariance matrix, \hat{x} is the inertial frame measurement estimate, and $h(\hat{x}^-)$ is the output of the nonlinear measurement model. This system estimates the three-dimensional location of the memory stick with sufficient accuracy to be able to land the aircraft on the target.

E. Flight Termination System

A manual takeover switch is provided so that a human safety pilot can take over control when required. The system is also provided with a remotely controlled “kill-switch” that ensures power to the motors is killed when triggered.

III. Payload

A. Sensor Suite

The GTAR entry uses three primary measurement sensors for navigation, stability and control. These devices are a laser range finder and a sonar altimeter for the GNC software, and an inertial measurement unit (IMU) for stability augmentation. The IMU employed by the GTAR team is the ADIS-16365-BMLZ built by Analog Devices Inc. It consists of a tri-axis digital gyroscope and tri-axis accelerometers that can measure forces up to ± 18 g. The laser range finder used is the Hokuyo URG-04LX-UG01. It is capable of measuring distances up to 4 m and has a maximum detection area of 240 degrees, with a resolution of 1 mm and 0.36 degrees respectively. The sonar altimeter used is the MB1040 LV MaxSonar EZ4 high performance ultrasonic range finder. It is capable of measuring distances up to 6.45 m away with resolution of 25.4 mm. These sensors are integrated around the Gumstix Overo Fire onboard computer which is a small and cost effective ARM Cortex-A8 OMAP3530 based computer-on-module. It is equipped with 256 MB Flash RAM, I2C and is UART and SPI capable. The onboard software leverages our previous work in rotorcraft control and allows us to fully utilize the GUST software suite of the GeorgiaTech UAV Research Facility.¹⁶

B. Communications

The Gumstix overo computer can communicate via 802.11g and Bluetooth wireless links. The computer will communicate with a ground computer using a wireless Local Area Network (LAN) link.

C. Power Management system

The system uses off-the-shelf battery packs which have a track record of proven safety. Two three cell Lithium Polymer Ion batteries are used, one drives the motors to provide lifting power, and the other runs the onboard computer. The batteries are charged off-board using off-the-shelf battery chargers. The vehicle comes with an integrated audible low-power warning, battery voltage can also be measured and transmitted to the ground station.

IV. Operations

A. Flight Preparations

Before each autonomous flight test or competition trial, a checklist of preparations are to be followed (see table A).

Table 1. Flight Checklist

Steps completed days before flight session	Charge flight batteries, transmitter batteries Load new software onboard and ground station Complete hardware-in-the-loop (HITL) tests to ensure proper operation of any code changes
Steps completed day of flight session	Ensure all flight test equipment is present. Set up ground station.
Steps completed before each flight	Clearly brief safety pilot of intention of flight Check structural integrity of vehicle and ensure proper center-of-gravity position.
During flight test	Pilot has primary discretion on whether to take manual control if vehicle is in jeopardy. Besides this discretion, safety pilot will only obey judges or ground station operator. Once the low voltage warning tone is heard, safety pilot takes control and lands the aircraft.

B. Man/Machine Interface

A ground station based on the GIT GUST software environment will continuously monitor the flight vehicle and display health and status information during the flight.¹⁶ The flight vehicle will send its current estimated position/heading, obstacle locations, and battery voltage via a wireless LAN data link. In addition, a frame-grabber is used to retrieve images from the incoming video stream for processing. Instructions from the ground station, including the adjustment of system parameters during manual flight, are transmitted over a wireless LAN data link. A safety pilot link is included, which operates via a separate 2.4GHz radio uplink.

V. Risk Reduction

A. Vehicle Status Monitoring

The flight vehicle continually monitors its surroundings for potential hazards and obstacles using the onboard laser range scanner. The information about potential hazard can be transmitted to the ground station for monitoring.

1. Shock and vibration isolation

The chosen onboard electronics have inherent tolerance to shock and vibration. Further vibration reduction is achieved through careful mounting of the hardware. The avionics package is mounted close to the center of gravity to minimize motion induced due to body rotations. The IMU is mounted directly on the avionics board. The laser range scanner and the sonar altimeter are mounted using a low cost vibration isolation mechanism.

2. Electromagnetic Interference (EMI)/Radio Frequency Interference (RFI) Solutions

The chosen quadrotor platform has brushless motors, which has reduced EMI signature. Further EMI mitigation is achieved by mounting the avionics package at the center of the airframe, and thus spatially separating it from the motors. Proper electric grounding and additional capacitors are used to provide further protection against EMI. A 2.4GHz transmitter was chosen for the video link, the safety pilot radio control link, and the data link. This eliminates the typical “servo jitter” affecting UAVs operating with 900MHz transmitters nearby. Possible interference between the different 2.4GHz systems is reduced by proper shielding and location of antennas.

B. Safety to Bystanders

The Quadrotor platform used in this work has a protective shroud that minimizes the risk of rotor strike and improves crash-worthiness. Further safety is incorporated by using off-the-shelf battery packs with a track record of proven safety. A manual takeover switch is provided so that a human safety pilot can take over control when required. Finally, the system is also provided with a remotely controlled “kill-switch” that ensures power to the motors is killed when triggered.

C. Modeling and Simulation

1. Modeling

Quadrotor dynamics flying in configuration shown in Fig.2(b) has been modeled in simulation. Assuming near hover aerodynamics, fuselage aerodynamics and forward flight rotor aerodynamics can be neglected. Hence, the total force acting on the Quadrotor is composed only of thrust and gravity forces. Newton’s second law in the body axis can be written as:

$$\begin{bmatrix} 0 & 0 & -(\tau_1 + \tau_2 + \tau_3 + \tau_4) \end{bmatrix}^T + F_g = m \frac{b}{dt} dv + \omega \times mv \quad (12)$$

where τ_i represents thrust magnitude on the i th rotor for $i = 1, 2, 3, 4$. F_g represents gravity force acting on the vehicle in the body frame, $v \in \mathbb{R}^3$ is the velocity in the body frame, $\frac{b}{dt} dv$ is the derivative of the body velocity with respect to the body frame, and $\omega \in \mathbb{R}^3$ is the angular rate of the body.

Neglecting forward flight aerodynamics, total moment acting on the Quadrotor composes of four different sources: hub yawing moment (M_{hy}), differential thrust moment (M_{dt}), inertial reaction moment (M_{ir}), and gyroscopic moment (M_{gy}).^{4,6,7} The primary moment contributions are from hub yawing moment and differential thrust moment while inertial reaction moment and gyroscopic moment provides insignificant contribution. Euler's law in the body axis can be written as

$$M = M_{hy} + M_{dt} + M_{ir} + M_{gy} = I\dot{\omega} + \omega \times I\omega \quad (13)$$

where I is the Quadrotor inertia matrix. The individual moment components are defined as follows:

$$M_{hy} = \begin{bmatrix} 0 & 0 & -M_1 + M_2 - M_3 + M_4 \end{bmatrix}^T \quad (14)$$

$$M_{dt} = \begin{bmatrix} l_y(-\tau_1 - \tau_2 + \tau_3 + \tau_4) & l_x(\tau_1 - \tau_2 - \tau_3 + \tau_4) & 0 \end{bmatrix}^T \quad (15)$$

$$M_{ir} = \begin{bmatrix} 0 & 0 & -(I_r\dot{\Omega}_1 - I_r\dot{\Omega}_2 + I_r\dot{\Omega}_3 - I_r\dot{\Omega}_4) \end{bmatrix}^T \quad (16)$$

$$M_{gy} = \begin{bmatrix} -q(I_r\Omega_1 - I_r\Omega_2 + I_r\Omega_3 - I_r\Omega_4) & p(I_r\Omega_1 - I_r\Omega_2 + I_r\Omega_3 - I_r\Omega_4) & 0 \end{bmatrix}^T. \quad (17)$$

where M_i is hub yawing moment magnitude of the i^{th} rotor acting on the body. Ω_i is angular velocity magnitude of the i^{th} rotor. I_r is moment of inertia of the rotor blade. l_x and l_y are distance from center of gravity to rotor hub in x and y direction. p and q are roll and pitch rates written in body axis. The total moment acting on the body is formed by combining the above terms:

$$M = \begin{bmatrix} l_y(-\tau_1 - \tau_2 + \tau_3 + \tau_4) - q(I_r\Omega_1 - I_r\Omega_2 + I_r\Omega_3 - I_r\Omega_4) \\ l_x(\tau_1 - \tau_2 - \tau_3 + \tau_4) + p(I_r\Omega_1 - I_r\Omega_2 + I_r\Omega_3 - I_r\Omega_4) \\ -M_1 + M_2 - M_3 + M_4 - (I_r\dot{\Omega}_1 - I_r\dot{\Omega}_2 + I_r\dot{\Omega}_3 - I_r\dot{\Omega}_4) \end{bmatrix}. \quad (18)$$

Vehicle's pitching motion can be generated by commanding differential Revolutions Per Minute (RPM) between two front motors and two back motors. Rolling motion can be generated by commanding differential RPM between two right motors and two left motors. Yawing motion can be generated by increasing motor RPMs on one diagonal and reducing motor RPMs on the other diagonal.⁶ Typical helicopter controls can therefore be mapped to motor commands x_i using Eq.(19).

$$\begin{aligned} x_1 &= \sqrt{\delta_T - \delta_\phi + \delta_\theta + \delta_\psi} & x_2 &= \sqrt{\delta_T - \delta_\phi - \delta_\theta + \delta_\psi} \\ x_3 &= \sqrt{\delta_T + \delta_\phi - \delta_\theta - \delta_\psi} & x_4 &= \sqrt{\delta_T + \delta_\phi + \delta_\theta + \delta_\psi} \end{aligned} \quad (19)$$

where δ_T , δ_ϕ , δ_θ , and δ_ψ represent thrust, roll, pitch, and yaw commands. In this form the commands are equivalent to a helicopter's collective, lateral cyclic, longitudinal cyclic, and tail rotor commands. A second order model is used to relate motor command x_i to motor RPM Ω_i . The second order model consist of two cascaded first order systems: the first one relates the motor commands to the motor states, and the second one relates the motor states to rotor RPM. Rotor aerodynamics is modeled using blade element and momentum theory. The relationship between motor command x_i , rotor angular velocity Ω_i , thrust τ_i , and moment M_i is modeled similar to reference 17. Vehicle's parameters including moment of inertias and maximum achievable RPM are currently being refined through experimentation.

2. Simulation

The GTAR team utilizes existing simulation software developed for research projects at Georgia Tech UAV lab.¹⁸ The simulation significantly reduces development time as the team can adopt navigation filter and controller already implemented in other UAVs. The simulation comes complete with modeling of uncertainties such as gusts, and modeling of indoor environments. All sensors are elaborately emulated and their noise properties are reproduced for testing purposes. Onboard code developed in simulation is directly used for autonomous flight. The setup is also capable of hardware-in-the-loop (HITL) test (simulating only vehicle dynamics and sensor readings). Figure 2 depicts a screen-shot of the vehicle in simulation.

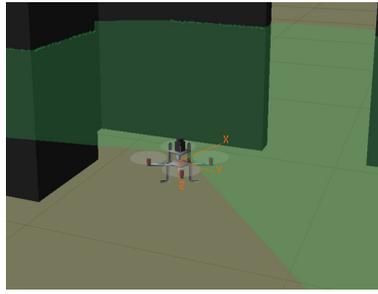


Figure 6. Quadrotor Performing Wall Following in Simulation

VI. Testing

The GTAR system is being rigorously subjected to flight testing at the indoor test flight facility at Georgia Tech. The VICON camera based object tracking system is being used to validate the navigation algorithm. We are using the protocols developed by the GTAR team for the 2009 effort for ensuring safety, efficiency, and reliability in flight tests.

VII. Conclusion

We presented the details of a Quadrotor Unmanned Aerial Vehicle intended for exploring indoor areas. The vehicle uses an off-the-shelf platform equipped with off-the-shelf avionics and sensor packages. Information from a scanning laser range sensor, inertial measurement unit, and an altitude measurement sonar are fused to form an elaborate navigation solution using Simultaneous Localization and Mapping (SLAM) methods. An important feature of this navigation architecture is that it does not rely on any external navigational aid, such as Global Positioning System signal. The information from the navigation solution is processed by a guidance logic which detects and follows walls in an indoor environment to ensure that maximum area of the indoor environment is traversed in a reasonable amount of time.

A cascaded inner-outer loop controller architecture which relates stick commands to attitude commands and attitude commands to velocity commands is used. The control architecture employs an optional adaptive element which can be used to mitigate modeling error and other system uncertainties. The control system also uses a linear Stability Augmentation System that uses rate feedback to dampen the vehicle angular rate response.

An elaborate simulation model of the vehicle has been developed and the navigation and control algorithms have already been validated in simulation. Efforts for flight testing of the vehicle are currently in progress. The Georgia Tech Aerial Robotics team intends to compete in the 2011 IARC competition with this vehicle.

Acknowledgments

The Georgia Tech Aerial Robotics team wishes to thank Jeong Hur, Dr. Suresh Kannan, Nimrod Rooz, Dr. Erwan Salaün, Stu Godlasky, Jeremy Montgomery, and Yuan Yao for valuable contributions.

References

- ¹Kayton, M. and Fried, W. R., *Avionics Navigation Systems*, John Wiley and Sons, 1997.
- ²Christophersen, H. B., Pickell, W. R., Neidoefer, J. C., Koller, A. A., Kannan, S. K., and Johnson, E. N., "A compact Guidance, Navigation, and Control System for Unmanned Aerial Vehicles," *Journal of Aerospace Computing, Information, and Communication*, Vol. 3, May 2006.
- ³Wendel, J., Maier, A., Metzger, J., and Trommer, G. F., "Comparison of Extended and Sigma-Point Kalman Filters for Tightly Coupled GPS/INS Integration," *AIAA Guidance Navigation and Control Conference*, San Francisco, CA, 2005.
- ⁴Bouabdallah, S., Noth, A., and R., S., "PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor," *Proc. of The IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2004.
- ⁵Chowdhary, G. et al., "Georgia Tech Aerial Robotics Team 2009 International Aerial Robotics Competition Entry," *Proc.*

of the 1st Symposium on Indoor Flight, International Aerial Robotics Competition, 2009.

⁶Portlock, J. and Cubero, S., "Dynamics and Control of a VTOL quad-thrustaerial robot," *Mechatronics and Machine Vision in Practice*, edited by J. Billingsley and R. Bradbeer, 2008.

⁷Guo, W. and Horn, J., "Modeling and simulation for the development of a quad-rotor UAV capable of indoor flight," *Modeling and Simulation Technologies Conference and Exhibit*, 2006.

⁸Johnson, E. and Kannan, S., "Adaptive Trajectory Control for Autonomous Helicopters," *Journal of Guidance Control and Dynamics*, Vol. 28, No. 3, May 2005, pp. 524–538.

⁹Kannan, S. K., *Adaptive Control of Systems in Cascade with Saturation*, Ph.D. thesis, Georgia Institute of Technology, Atlanta Ga, 2005.

¹⁰Chowdhary, G. and Johnson, E., "Flight Test Validation of Long Term Learning Adaptive Flight Controller," *Proceedings of the AIAA GNC Conference, held at Chicago, IL*, 2009.

¹¹Johnson, E. N., *Limited Authority Adaptive Flight Control*, Ph.D. thesis, Georgia Institute of Technology, Atlanta Ga, 2000.

¹²Chowdhary, G. V. and Johnson, E. N., "Theory and Flight Test Validation of Long Term Learning Adaptive Flight Controller," *Proceedings of the AIAA Guidance Navigation and Control Conference*, Honolulu, HI, 2008.

¹³Steux, B. and Hamzaoui, O. E., "Coreslam on openslam.org," website.

¹⁴Steux, B. and Hamzaoui, O. E., "CoreSLAM : a SLAM Algorithm in less than 200 lines of C code," Tech. rep., Mines ParisTech, Center for Robotics, Paris, France, 2009.

¹⁵Sobers, M. J., *INDOOR NAVIGATION FOR UNMANNED AERIAL VEHICLES USING RANGE SENSORS*, Ph.D. thesis, Georgia Institute of Technology, 2010.

¹⁶Johnson, Eric N, C. A. J., Watanabe, Y., Ha, J.-C., and Neidhoefer, J., "Vision Only Control and Guidance of Aircraft," *Journal of Aerospace Computing, Information, and Communication*, Vol. 23, No. 10, October 2006.

¹⁷Johnson, E. and Turbe, M., "Modeling, Control, and Flight Testing of a Small Ducted Fan Aircraft," *Journal of Guidance Control and Dynamics*, Vol. 29, No. 4, July/August 2006, pp. 769–779.

¹⁸Johnson, E. N. and Schrage, D. P., "System Integration and Operation of a Research Unmanned Aerial Vehicle," *AIAA Journal of Aerospace Computing, Information and Communication*, Vol. 1, No. 1, Jan 2004, pp. 5–18.