MODELING AND SIMULATION FOR SMALL AUTONOMOUS HELICOPTER DEVELOPMENT

Eric N. Johnson*
work conducted employed at
The Charles Stark Draper Laboratory, Inc.
Cambridge, Massachusetts USA
now employed at
Lockheed Martin Aeronautical Systems
Marietta, Georgia USA

Paul A. DeBitetto*
The Charles Stark Draper Laboratory, Inc.
Cambridge, Massachusetts USA

Abstract

The Charles Stark Draper Laboratory, Inc., Massachusetts Institute of Technology, and The Boston University have cooperated to develop an Autonomous Aerial Vehicle (AAV) that competed in and won the 1996 International Aerial Robotics Competition, sponsored by the Association for Unmanned Vehicle Systems, International (AUVSI). Development of the vehicle continues to support on-going research in the area of autonomous systems. A simulation capability has been developed to support the design, development, and test of the navigation, control, guidance, and vision processing sub-systems, as well as human-machine interfaces and procedures. The use of the simulation described in this paper is identified as a key factor in the success of the program at the competition and operations since.

Introduction

The International Aerial Robotics Competition, organized by the Association for Unmanned Vehicle Systems, International (AUVSI), provides a unique opportunity to develop an autonomous vehicle system with many of the same features, components, and potential pitfalls as fielded unmanned vehicle systems - and to test that system in a real-life/competition environment. A team was formed between students, faculty, and staff of the Massachusetts Institute of Technology, Boston University, and the Charles Stark Draper Laboratory, Inc. that competed in and won the 1996 competition, held in Orlando, Florida.

The competition consists of a 60 by 120 ft. field with five randomly placed drums [1]. Each drum has one of two kinds of labels. A contestant's vehicle must start and finish in a 15 by 15 ft. area in a corner of the field, provide a map of the location and classification of the drums, and pick up a small disk on one of the drums. During a contest attempt, the vehicle and any ground equipment must operate completely autonomously, i.e. no human operators.

The Draper small Autonomous Aerial Vehicle (AAV) system consists of an aerial vehicle (Figure 1), a Ground Control Station (GCS), a vision processor, and a safety pilot. Figure 2 is a functional block diagram and shows how the aerial vehicle, GCS, and vision processor fit together. The aerial vehicle performs navigation and control on-board using a redundant suite of sensors including; a Differential GPS (DGPS) unit, an Inertial Measurement Unit (IMU), a sonar altimeter, and a flux compass. Guidance and operator control happens on the GCS. The helicopter also carries a camera and transmitter that provides real-time video images to the ground. A ground based vision processor then converts the image data into drum position and classification estimates.

The contest field size requirements, the restricted take-off and landing area, a desire for flight characteristics that allow manual control, and the payload capacity desired led to the selection of a helicopter for the aerial vehicle. The aerial vehicle is an off-the-shelf radio-controlled helicopter with a six foot rotor diameter produced by TSK, called the Black Star. The helicopter has an empty weight of 15 lb. and a payload capacity exceeding 9 lb. The helicopter is powered by a 32 cc gasoline engine.

* Member AIAA

On-board Processing

A PC-104 stack is used as the main processing module for the vehicle. In particular, an Ampro 486 DX2 processor is used. The navigation and control algorithms run on-board.

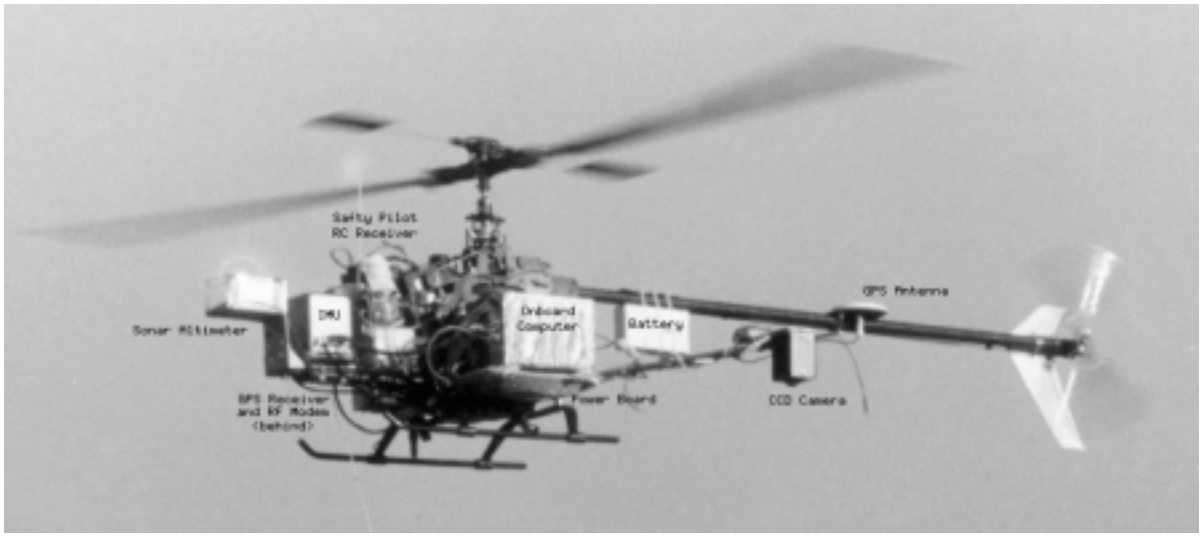

Figure 1. 1996 Draper Small AAV

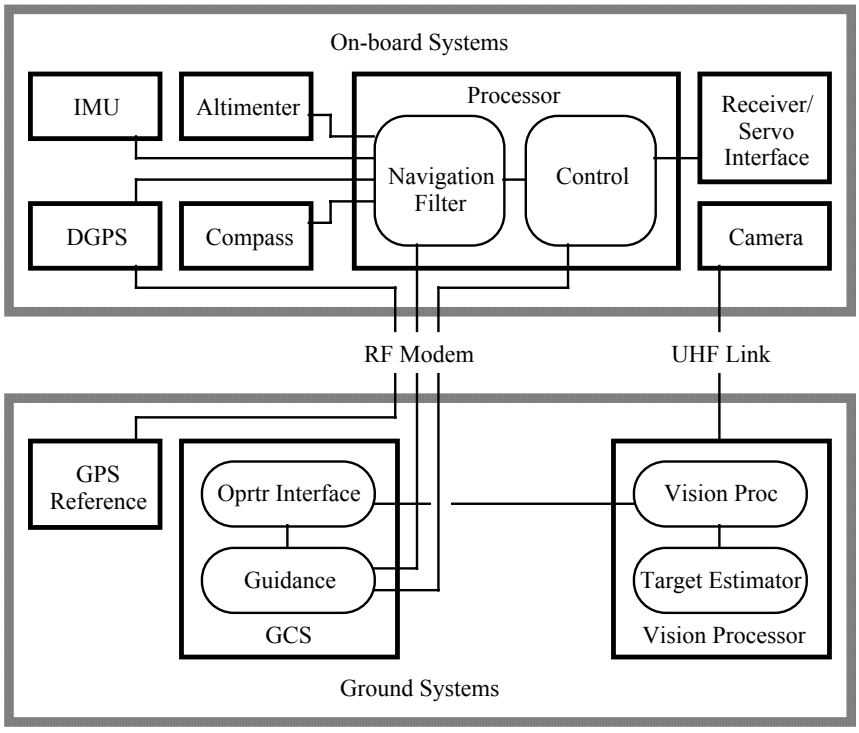

Figure 2. System Functional Block Diagram

Navigation

Navigation Sensors

A DGPS system with local ground reference station is employed to supply low bandwidth position and velocity measurements to a navigation filter. The NovAtel RT-20 system was used.

An IMU was employed to provide high bandwidth angular rates and linear acceleration measurements to the navigation filter for updating the attitude, angular rates, linear position and velocity state estimates. The sensor used is the Systron Donner Motion Pak, consisting of three quartz rate gyros and three quartz flexure accelerometers.

The sonar altimeter provides high rate/resolution altitude information at close range. A flux compass is utilized to prevent long-term drift of the navigation filter heading estimate.

Navigation Algorithm

The navigation filter merges the sensor information from the IMU, DGPS unit, sonar, and compass. Because these sensors have unique update rates and error properties, an extended Kalman filter was selected to accomplish the task. When updates of the DGPS, sonar, or altimeter occur, an optimal discrete update [2] is made of the state vector and the state error covariance matrix.

Guidance and Control

Receiver/Servo Interface

An on-board receiver/servo interface allows the safety pilot to select between *ground transmitter controlled* and *on-board processor controlled* modes. The interface is powered by the R/C receiver battery (independent of the power supply for the on-board processor, DGPS, other sensors, and modem), and communicates with the on-board computer via a standard serial port.

Control Law

The control system is designed to generate effector commands for the roll and pitch cyclic, collective pitch, tail rotor pitch, and throttle based on a commanded position, heading, and velocity. It utilizes feedback from the navigation system, including position, velocity, attitude, and angular rate estimates. The ability to command a velocity in addition to position and heading allows the control system to eliminate steady state errors during forward flight or other type of commanded movement.

The control system is divided into four loops: roll, pitch, yaw, and collective/throttle. The loops are used to control lateral position, longitudinal position, heading, and altitude respectively. The trim positions of cyclic, collective, and tail rotor are used as internal states, this allowing physical interpretation of these states and allowing the safety-pilot to pre-trim the helicopter to minimize initial transients in the system.

Guidance

The guidance system generates position, heading, and velocity commands based on the current state of the helicopter as reported by the navigation system and a waypoint list. The guidance system commands a straight line course between waypoints at a nominal velocity, and transitions between these trajectories using a nominal acceleration. The values used for the nominal acceleration and velocity can be changed by an operator at the ground station.

Communication

The radio modem link performs three main functions. First, it sends navigation information from the helicopter to the ground computer. Second, it relays control system commands from the ground computer to the helicopter. Finally it is used to send differential GPS data from the ground GPS receiver to the on board GPS receiver.

Ground Control Station (GCS)

The GCS including the operator interface utilizes an Intel 486-based laptop. During autonomous flight, displays shown on the GCS allow monitoring of the navigation system estimates, state of the communication link to the helicopter, waypoint list, DGPS solution status, state of the control system, and state of the vision processing. Also, the interface is used to perform other operations.

The display consists of four elements. The primary flight display, depicted in Figure 3 depicts state information such as position, velocity, and attitude as reported by the navigation system. The navigation display is an electronic map of the flight area including the helicopter position as reported by the navigation system, the location of the DGPS reference station, the waypoint list, and the commanded position given the control system. The annunciator panel shows the status of most sub-systems. The annunciator for any particular system can take on various colors and text labels depending on the state the system. The systems pages provide access to the details of specific sub-systems.

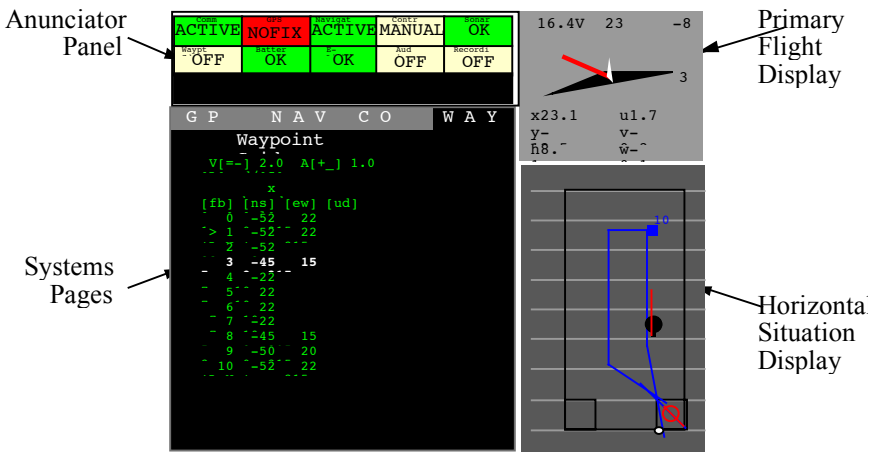

Figure 3. Operator Interface Displays

Vision Processing

A greyscale CCD camera with a UHF transmitter was mounted on the helicopter. It was downward facing, so drum labels will be visible. Contest vision processing is broken into three steps: Drum location, drum classification, and target estimation.

The drum detector is composed of three parts: a gradient operator, matched filter, and a peak extractor. The matched filter processes the gradient image. Since the labels characteristically have many edges, they are easy to pick out against the relatively smooth background in the gradient image by matching to regions of clustered edges. The expected size of the labels is calculated from the height of the helicopter and this determines the size of the matched filter template. The matched filter is also designed to be insensitive to a long edges such as those from the rim and sides of the drum. Then the peaks of the matched filter output are the most likely positions of the labels, and hence the drum positions.

The label classifier matches the pixels around the label position estimates to templates. The orientation of the input pixels is normalized to the vertical, so only two templates are necessary for each label type (The second template is just the first rotated by 180 degrees). The scores of the template match affect the confidence that the contact is an actual drum.

An optimal estimator combined outputs from the vision processor along with its reported confidences into a single list of probable drum locations and classifications. The five solutions with the highest confidence were turned into the judges as the system output.

Simulation Capability

A simulation capability was put in place to support the design, development, and test of the navigation, control, guidance, and vision processing sub-systems, as well as man-machine interfaces and procedures. Also, it was used for safety-pilot training. The simulation capability includes:

modeling:

- Main and tail rotor forces and moments, including tip-path dynamics of the main rotor and Bell/Hiller flybar arrangement.
- Aerodynamic forces and moments due to the fuselage and tail surfaces
- Ground contact forces and moments of main gear and tail
- Sensor models of DGPS, IMU, and sonar altimeter
- Servo dynamics

hardware/software compatibility:

- Hardware compatibility with receiver/servo interface, allowing piloted flight using actual safety-pilot transmitter
- Hardware compatibility with ground computer (GCS), allowing tests of all man-machine interfaces
- Software compatibility with on-board processing, allowing the same software to be used in the simulation as used on board the actual AAV

scene generation and analysis:

- Scene generation including contest elements, such as the field and drums with labels
- Parameter logging, un-logging, and plotting
- Linear model extraction
- Time control: Fast time, slow time, pause, step, and real time

This simulation capability has played a key role in the success of Draper small AAVs to date, and is described in the following sections. The ways in which this simulation capability was utilized is also described in following sections.

Modeling for Simulation

Dynamic modeling for this project included a 6-degree-of-freedom rigid-body representation of the motion of the helicopter. In addition, main rotor and main rotor flybar first order flapping dynamics are included. The main and tail rotor forces and moments are determined using an actuator disk theory formulation. Also, a ground contact model was implemented. This model was based largely on the material in reference [3], with important extensions described in this section.

Rotor Thrust and Power

Since the main rotor tip speed of the TSK Black Star is approximately 400 feet per second, and the largest dimension of the contest arena is 120 ft, the maximum expected tip speed ratio is less than 0.05. Asymmetric wake and retreating blade stall can be neglected in rotor modeling for this range of tip speed ratios, dramatically simplifying the calculations necessary for rotor thrust and power.

Using the formulation for a perfect actuator disk in an oblique flow, the following two equations must be solved simultaneously to get the main rotor thrust and induced velocity [3] and [4], this is accomplished numerically:

$$T = \rho \frac{abc}{4} \left(w_b - v_i \right) \Omega R^2$$

$$v_i = \frac{\eta_{GE} T}{2\pi R^2 \rho \sqrt{u^2 + v^2 + \left(w_r - v_i \right)^2}}$$

where:

T	Main rotor thrust
v_i	Main rotor induced velocity at the rotor

and:

a	2-D lift curve slope of rotor blade
b	Number of blades (2)
c	Chord of rotor blades
R	Radius of rotor blades
Ω	Angular rate of main rotor rotation
u , v , w	Velocity of the rotor with respect to the air expressed in the body frame, u forward, v to the right, and w down
ρ	Air density
w_r	The velocity of the rotor with respect to the air, normal to the disk plane; it can be calculated by: $w_r = w + a_1 u - b_1 v$
w_b	The velocity of the average blade with respect to the air, normal to blade; it can be calculated by: $w_b = w_r + \frac{2}{3} \Omega R \theta$
η_{GE}	Reduction in induced velocity due to ground effect; it is calculated assuming that ground effect is of the form [4]: $\eta_{GE} = \frac{K_{GE}}{1 + \left\| \frac{2R}{h} \right\|^2}$

where:

a_1 , b_1	Main rotor first order flapping, pitch and roll
θ	Main rotor pitch (collective pitch angle)
K_{GE}	An empirically determined value, setting the magnitude of ground effect
h	The height of the main rotor above the ground

Once rotor thrust and induced velocity are determined, induced and parasite power can be

calculated with [3] (parasite power) and [4] (induced power):

$$P_{Induced} = T\left(v_i - w_r\right)$$

$$P_{Parasite} = \frac{\rho}{8} c_{Do} R^2 bc\Omega\left[\Omega^2 R^2 + 4.6\left(u^2 + v^2\right)\right]$$

where:

$P_{Induced}$	Induced power of the main rotor
$P_{Parasite}$	Parasite power of the main rotor
c_{Do}	2-D zero-lift drag coefficient of a rotor blade

The tail rotor thrust, power, and torque are calculated in the same manner as the main rotor. The most significant difference is that ground effect is neglected.

Main Rotor and Flybar Flapping

As described above, first order flapping dynamics are included for the main rotor. In addition, the TSK Black Star has a Bell/Hiller flybar main rotor arrangement, as can be seen in Figure 1. This arrangement is common in small helicopters, and serves to stabilize the helicopter in roll and pitch - making manual control considerable easier.

The flybar adds dynamics to the system that need to be modeled for Guidance, Navigation, and Control (GN&C) analysis or training. The approach taken, discussed briefly here, was to use the first order flapping dynamics as developed for the main rotor in [3] and applying them to the flybar - as though it was another rotor disk. Once this is completed the mechanical connection between the swash plate and the flybar must be included as a gain between commanded main rotor flapping and commanded flybar flapping. Also, a gain between flybar flapping and commanded main rotor flapping must be included.

Ground Interaction

In order to simulate entire missions, take-offs and landings must be available in the simulator. To this end a ground interaction model was include, that is the forces and moments generated as the helicopter landing skids or tail boom hit the ground.

Contact points on the helicopter body (the skids and tail) were modeled as 3-D springs and dampers, with static and dynamic friction coefficients. Once the proper velocities and deflections have been calculated, the force in the vertical direction:

$$F_z = -K_d d - K_{\dot{d}} w_p$$

for positive deflections, where:

d , w_p	Calculated vertical deflection and deflection rate of the ground contact point
F_z	Vertical force due to contact point
K_d , $K_{\dot{d}}$	Spring stiffness and damping parameters

When a contact point is not skidding with respect to the ground, a static friction coefficient is used, and a spring and damper generate forces and moments to keep the contact point were it is on the ground. This approach prevents the helicopter from moving around when small forces and moments are generated by the rotors while still firmly planted on the ground. Another consequence is that two additional states result for each ground contact point. The physical interpretation of these two states is the longitudinal and lateral deflection of the contact point with respect to the body of the helicopter.

To use the lateral deflection as an example, the side force due an particular contact point is proportional to this lateral deflection and the rate of change of lateral deflection. Prior to skid, the rate of change of the lateral deflection is the lateral velocity of the contact point with respect to the ground:

$$\dot{s}_y = -v_p$$

$$F_y = K_s s_y + K_{\dot{s}} \dot{s}_y$$

where:

s_y	Lateral deflection of contact point
$\dot{s}_y$	Lateral deflection rate of contact point
v_p	Lateral velocity of the contact point with respect to the ground
F_y	Lateral force due to contact point
K_s, $K_{\dot{s}}$	Spring stiffness and damping parameters

Once the static friction force is exceeded (proportional to the vertical force at the contact point), a more conventional ground interaction model

is used, with a spring and damper force normal to the ground, and a friction force generated opposite the direction of the skid.

Sensor Models

Sensor models were included, particularly for navigation algorithm development. DGPS system errors are modeled as colored Gaussian noise. The IMU sensor model errors included: limits, scale factor, bias, and Gaussian noise. The compass and sonar models both also included limits, scale factors, biases, and Gaussian noise. All sensor model noise parameters were based on engine running static tests. In addition to error analysis, having sensor models that included all the same calibration, reference frame, and unit issues as the real sensors dramatically reduced the amount of software development that needed to be done using the real sensor hardware.

Analysis and Data Visualization

Scene Generation

A scene generation system was developed for Silicon Graphics, Inc. (SGI) workstations utilizing the OpenGL graphics language. The scene includes the contest arena, helicopters, drums, clouds, and trees. Piloted simulations revealed the importance of detailed textures and background (including trees and clouds) so that the human pilot can judge the motion of the helicopter relative to the background. A sample image is shown in Figure 4, depicting the view of the helicopter as seen from a 6 foot tall person standing just outside the contest volume.

Figure 4. Draper Small AAV Simulator Scene Generation

For the helicopter scene object, it was found to be quite useful for GN&C development and training to show the main rotor coning and first order flapping generated by the helicopter model as changes in geometry of the spinning disk drawn in the scene. The situation in analogous to depicting rudder and elevator deflections on the scene object for an airplane.

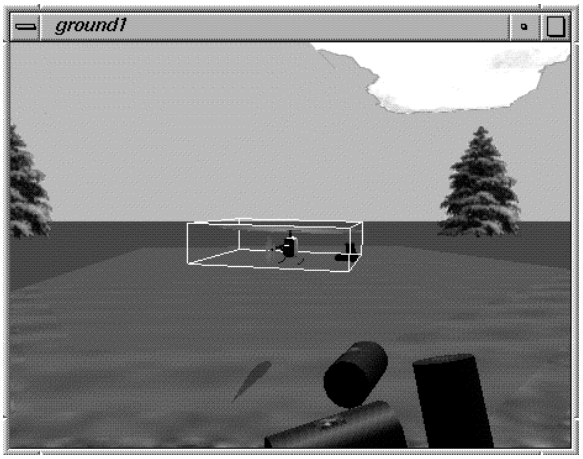

Figure 5. Navigation Solution Depicted in Scene

In addition, the depiction of the navigation solution was also depicted. This was a useful analysis tool for working with the navigation sensor fusion algorithms. A sample image is shown in Figure 5. If the navigation system is working properly, the white box (depicting the navigation solution) will contain the helicopter.

Plotting, Logging, and Un-Logging

The simulation utilized the CSIM Framework, a Draper Laboratory simulation environment which, among other things, includes utilities for real-time data plotting and logging. In addition, all simulation parameters can be examined and modified during execution. There is also a logged data playback feature.

Linear Model Extraction

A linear model extraction utility was also implemented, to facilitate linear analysis of the plant. It was implemented as a command that could be executed at any time. This command would linearize the plant, including the rotor flapping states, the rigid body states, the effectors, and the sensors at the current flight condition. The text file output was then imported into one of the common control system analysis packages.

Uses of Simulator

GN&C Development and Test

One of the most common uses of the simulator was to connect the GCS to the simulator running on an SGI workstation. This mode is suitable to test all Guidance, Navigation, and Control (GN&C) algorithms, most software, and GCS operator interface. This configuration is depicted in Figure 6.

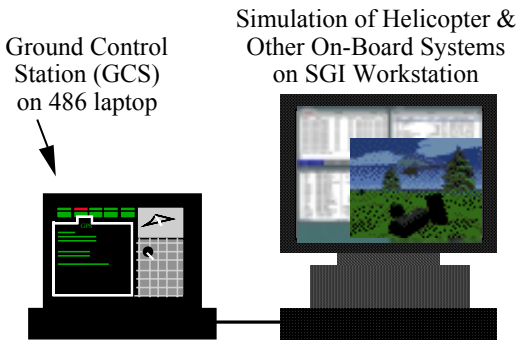

Figure 6. GCS Hardware-in-the-Loop Configuration

The software which normally runs on the on-board processor is also compiled into the simulation - allowing this software to be tested during the simulation. Also, this allowed GN&C development to occur using the same software which runs on the helicopter - avoiding the need to "port" the code to the new platform once the algorithms were developed or a change was made. This software included the control system, the navigation system, and the communication protocol with the GCS and vision processing element.

This capability was used extensively prior to hardware completion and prior to flight test. In addition, anomalies discovered in flight test were, as standard practice, corrected in the simulation BEFORE a modification was tried in flight test. In addition, a linear model extraction utility was implemented and used to facilitate off-line linear analysis.

Software Development and Test

The next level of hardware-in-the-loop is to include the receiver/servo interface and the safety pilot transmitter. This allows the test of the safety-pilot interface, the receiver-servo interface hardware, and more software. Also, the simulation can be used for training all flight test operators in this configuration: the GCS operator and the safety pilot. This configuration is depicted in Figure 7.

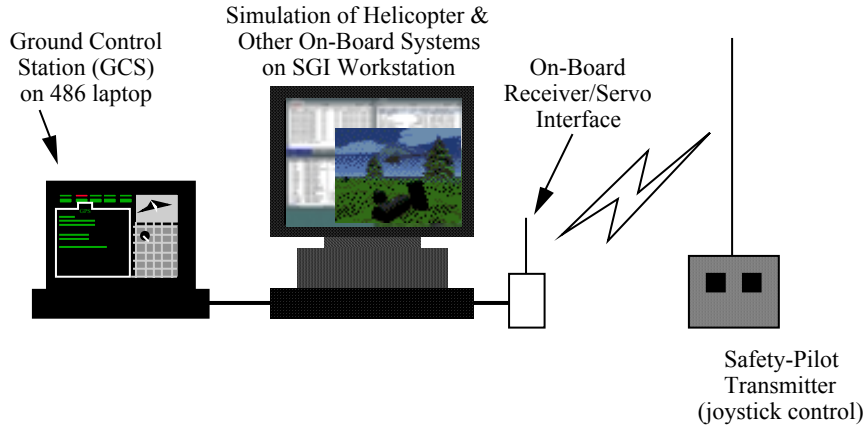

Figure 7. Hardware-in-the-Loop Configuration

The hardware-in-the-loop simulation configuration depicted in Figure 7 facilitates real-time testing of almost all software used with the vehicle (both on and off board). Software work performed outside the safety of the simulator was confined primarily to the low-level interfaces to the navigation sensors. In addition to the obvious benefit of testing the software, this also had the effect of making the software implementation job much less tedious. This was particularly true of the GCS software, where new functionality could be immediately tested in a flight test environment. This gave immediate feedback as the success or failure of a change.

Operator Interfaces and Procedures

The hardware-in-the-loop simulation configuration depicted in Figure 7 facilitates a detailed mission rehearsal, including both the GCS operator and safety-pilot. Tests can be performed repeatedly in the simulator before they are done for the first time in flight test. This allowed a myriad of operator interface design issues and test procedures to be ironed out prior to using up valuable flight test time.

Training

Because the hardware-in-the-loop simulation configuration depicted in Figure 7 facilitates a detailed mission rehearsal, including both the GCS operator and safety-pilot, it is also an effective training tool. This gives the GCS operator and safety-pilot the experience necessary to handle in-flight problems more effectively. In addition, it is an excellent environment to learn and practice normal and emergency procedures.

Image Processing Development and Test

The use of logged data playback allowed almost all vision processing development to take place prior to its initial flight test, depicted in Figure 8. Flight tested were performed while the navigation system solution and the on-board video image were recorded. The navigation data and video images were then played back and fed to the vision processing sub-system in real-time.

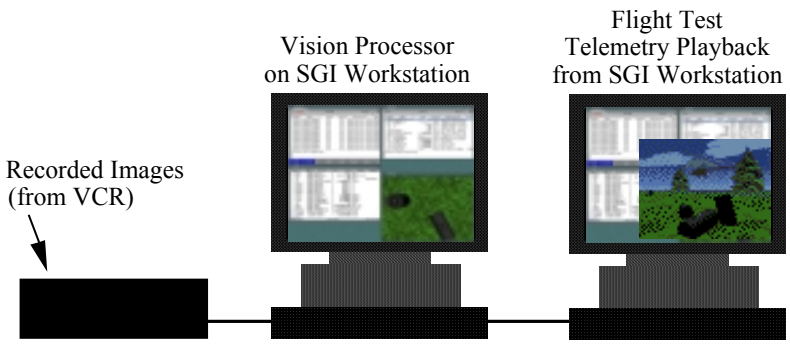

Figure 8. Vision Processing Test Configuration

In addition to the above method, which utilizes recorded video to generate realistic images to the vision processing system, another configuration was also used. Specifically, the scene generation system was set up to generate a scene that represented the vantage point of the on-board camera. Although testing image processing systems with synthetic images is not a rigorous test, since real-life degradation effects were not present, it was nonetheless a valuable test for checking algorithms and software for varied geometries, and under a more controlled set of conditions than available during flight tests. As a result, this was another useful form of testing.

Results and Conclusions

The MIT/Boston University/Draper lab entry won the 1996 International Aerial Robotics Competition. Seven flights were made. During the winning flight, all five drums were located, and two classified correctly. No attempt was made to pick up the disk. It was the first time in the history of that contest that a vehicle had flown to a series of waypoints completely autonomously take-off to landing, only touching the ground in the designated area. In addition, this AAV also executed a mission.

The following conclusions were drawn relating to the use of simulation for AAV development and test:

1) Simulation is an essential part of guidance, navigation, and control design and development. The extensive use of simulation for this purpose generated a design that only needed to be tuned in flight test, saving the program several weeks and perhaps a crash of the aerial vehicle. Also, using the simulation as a tool to correct problems uncovered in flight test saved considerable time as well.

2) Simulation is an essential part of software development. Developing interfaces, real-time code, and simply executing all possible code paths is greatly facilitated by a simulation capability. In addition, the act of software development itself is considerably less tedious, and efforts are more focused when additions and changes can be immediately tested. Also, it is believed that the use of simulation for software development saved the program several weeks by allowing software to developed concurrently with the associated hardware.

3) Simulation is an essential part of operator interface and development. Mission rehearsals led to many changes to the interfaces. A design without test in the simulation would have required changes during the flight test phase.

4) Utilizing logged telemetry data and recorded on-board video can be effectively used to develop and test vision processing schema. Tuning of the vision processing was done almost completely in the lab using this technique.

5) The maintaining of a well-trained crew to operate the system is critical during a flight test program. Human operators often need to make up for equipment failures or newly discovered problems. Hand signals, checklists, consistent terminology, and simulation training/mission rehearsal where effective tools.

In short, the extensive use of the simulation described in this paper was a key factor in the success of the program at the Aerial Robotics Competition and operations since. The close relationship between flight test and simulation allowed the flight validation of the guidance, navigation, and control sub-systems to take place within a two week period, clearly indicating that most development work was successfully completed in simulation. It allowed most software problems to be addressed before flight test. It made software development less tedious,

since additions could be utilized immediately in the simulator. Mission rehearsals and operator training led to better designed human-machine interfaces, better procedures, and a better trained crew. The use of logged data playback allowed almost all vision processing development to take place prior to its initial flight test.

References

[1] R. C. Michelson, "Official Competition Rules for the 1996 Aerial Robotics Competition," Association for Unmanned Vehicle Systems International, 1996.

[2] A. Gelb, "Applied Optimal Estimation," The MIT Press, 1984.

[3] R. K. Heffley and M. A. Minch, "Minimum-Complexity Helicopter Simulation Math Model," NASA CR-177476, 1988.

[4] W. Z. Stepniewsky and C. N. Keys, "Rotary-Wing Aerodynamics," Dover Publications, Inc., 1984.