Army Research Laboratory

Control of an Autonomous Radio-Controlled Helicopter in a Modified Simulation Environment Using Proportional Integral Derivative Algorithms

by Ainsmar X. Brown and Richard D. Garcia

ARL-CR-609 **June 2008**

prepared by

National Institute of Aerospace
100 Exploration Way
Hampton, VA 23666

and

University of South Florida
4202 E. Fowler Ave.
Tampa, FL 33620

under contract

NAS1-02117

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer’s or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory
Aberdeen Proving Ground, MD 21005-5066

ARL-CR-609 **June 2008**

Control of an Autonomous Radio-Controlled Helicopter in a Modified Simulation Environment Using Proportional Integral Derivative Algorithms

Ainsmar X. Brown
National Institute of Aerospace

Richard D. Garcia
University of South Florida

prepared by

National Institute of Aerospace
100 Exploration Way
Hampton, VA 23666

and

University of South Florida
4202 E. Fowler Ave.
Tampa, FL 33620

under contract

NAS1-02117

REPORT DOCUMENTATION PAGE	***Form Approved OMB No. 0704-0188***

1. REPORT DATE *(DD-MM-YYYY)* June 2008	**2. REPORT TYPE** Final	**3. DATES COVERED (From - To)** September 2007–December 2007

4. TITLE AND SUBTITLE

Control of an Autonomous Radio-Controlled Helicopter in a Modified Simulation Environment Using Proportional Integral Derivative Algorithms

5a. CONTRACT NUMBER

NAS1-02117

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Ainsmar X. Brown* and Richard D. Garcia†

5d. PROJECT NUMBER

8BY0B2

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

National Institute of Aerospace
100 Exploration Way
Hampton, VA 23666

University of South Florida
4202 E. Fowler Ave.
Tampa, FL 33620

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Research Laboratory
ATTN: AMSRD-ARL-VT-UV
Aberdeen Proving Ground, MD 21005-5066

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

ARL-CR-609

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

*National Institute of Aerospace, 100 Exploration Way, Hampton, VA 23666
†University of South Florida, 4202 E. Fowler Ave., Tampa, FL 33620

14. ABSTRACT

The U.S. Army Research Laboratory (ARL), along with groups like the U.S. Army Aviation and Missile Research, Development, and Engineering Center, acts as a resource in developing on-field technologies under the U.S. Army Research, Development, and Engineering Command. Currently, ARL's Vehicle Technology Directorate is interested in expanding its Unmanned Vehicles Division to include rotary wing and microsystems control. ARL intends to use unmanned aircraft systems not only for reconnaissance missions, but also for targeting and lethal attacks. This project documents ongoing work expanding ARL's program in research and simulation of autonomous control systems. A proportional integral derivative control algorithm was modeled in MathWorks Simulink and communicates to a flight simulator modeling a physical radio-controlled helicopter. Waypoint navigation and flight-envelope testing were then systematically evaluated to the final goal of a feasible autopilot design.

15. SUBJECT TERMS

UAV, simulation, VTOL, autonomous

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Ainsmar X. Brown
a. REPORT UNCLASSIFIED	**b. ABSTRACT** UNCLASSIFIED	**c. THIS PAGE** UNCLASSIFIED	UL	42	**19b. TELEPHONE NUMBER** (*Include area code*) (410) 278-7024

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

List of Tables

INTENTIONALLY LEFT BLANK.

1. Introduction

The utilization of unmanned vehicles (UVs) in the military dates back to World War I (*1*) with free-falling munitions equipped with basic guidance systems. Today, technologies related to UVs have produced vehicles that are either on par or exceeding the capabilities of conventional manned vehicles. These advancements are specifically noteworthy in the area of unmanned aircraft systems capable of stationary hover and vertical flight. To date, there are only a handful of proven platforms that have all made significant contributions to the field of unmanned military aircraft.

These types of aircraft typically have highly specialized designs and systems-integration techniques that provide limited open-source content for external research and individuals not closely associated with the program. This, coupled with the cost and shear size—ranging from 200 lb for Yamaha's RMAX up to 4300 lb for Boeing's A160 Hummingbird—has severely limited their widespread development. Due to recent advancements in sensor and power-plant technologies, the need for aircraft in the aforementioned size category has significantly diminished. Along with the development of modern autopilots, such as those designed by companies like Micropilot and Procerus or in many cases universities with expansive computer engineering departments, the development of smaller vehicles, similar to those given in figure 1 (*2–5*), has become more feasible and desirable.

Although the aircraft shown in figure 1 are smaller, lighter, quieter, and cheaper than those in figure 2 (*6–10*), they are all products designed for the general consumer and do not meet general requirements for survivability, safety, etc. The current trend is for research groups to use these radio-controlled (RC) vehicles as test beds for their own autopilot software and hardware designs. This allows widespread research to be conducted without the high cost and security-restricted aircraft currently in operation with the armed forces. With only a few minor adjustments, these standard RC aircrafts become serious components of UAS research and development. One current issue affecting this type of development is the use of simulation and the role it should play throughout the development process. Current simulation software such as MathWorks Simulink[*] and Laminar Research's X-Plane can provide highly accurate and influential test data. Although this data can simplify and expedite the development process, its role as a development tool has been minimal.

[*] Matlab/Simulink is a trademark of The MathWorks, Inc.

Note: E-Flight Blade CX 2 photo © 2008 by Horizon Hobby, Inc.

Figure 1. Smaller-scale RC vehicles. From top to bottom, left to right: Align T-Rex, WowWee Flytech Dragonfly, Minicopter Maxi Joker 2, and E-Flite Blade[*] CX 2.

2. Background Concepts in Aeronautics

2.1 X-Plane Model

The equations of motion for a 6 degree-of-freedom helicopter can be modeled a number of ways. It should be noted that additional care must be taken with regard to angular orientation in a three-dimensional (3-D) space. This care is due to the ambiguity of a vehicle's orientation when the pitch attitude is equal to 90°. The traditional convention is to use Euler angles, and the second, less common method utilizes quaternion math to eliminate the ambiguity at a pitch attitude of 90°. The application of the quaternion method is described in the Green's thesis (*11*). For the purposes of this design, this helicopter will not operate under aggressive flight conditions. The main flight modes for the vehicle include hover and standard forward flight. More specifically, the helicopter will never need to achieve a pitch attitude anywhere near 90°, which would cause a singularity and invalidate the Euler angle equations.

The simulation, as previously stated, depends on modeling and visualizations from Laminar Research's X-Plane. X-Plane models low-speed, high-speed, and, to a lesser extent, transonic flight regimes using a multistep process. A summary of information proved in Meyer (*12*) as well as the equations behind the theoretical model as explained in Nelson (*13*) and Prouty (*14*)

[*] Blade and E-Flite are registered trademarks of Horizon Hobby, Inc.

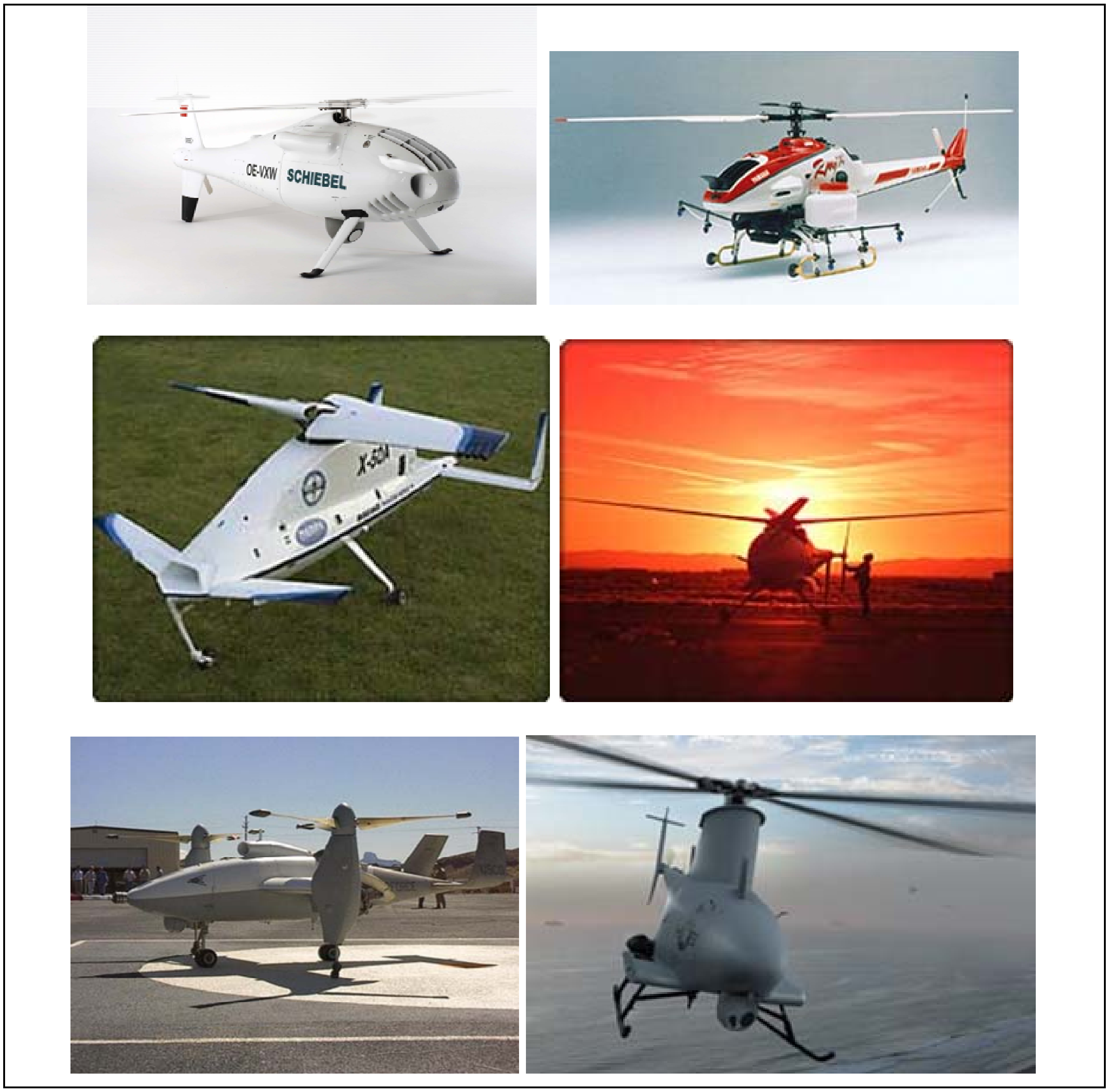

Figure 2. Several vertical takeoff and landing (VTOL) UASs theoretically capable of satisfying the proposed laser-targeting mission as described by the U.S. Army. From top to bottom, left to right: Schiebel's CamCopter, Yamaha's RMAX, Boeing/DARPA's X-50A as well as Boeing's A160 Hummingbird, Bell's Eagle Eye, and Northrop Grumman's Fire Scout.

follow. For reference, Prouty will provide the background on helicopter aerodynamics and flight modeling in this section while Nelson explains the general concepts in automatic vehicle control used in section 2.2.

The first step is to break down the surfaces of the aircraft into smaller segments in order to use blade-element theory. This builds up the incremental forces on the entire vehicle by finding the forces on each blade element, which can be thought of as a theoretical area over which a differential force is applied due to the aerodynamic properties of that segment. Although such elements can include a slice of a horizontal stabilizer or a slice of a main fuselage, they will often be limited to certain bodies depending on whether one is trying to determine lift or drag from a vehicle. A closed-form expression for the general forces created by an aerodynamic surface is given in equation 1 in the form of a thrust expression.

$$dT = \frac{1}{2}\rho V_E^2 c[a(\beta\text{-}\varphi\text{-}\alpha_i)dr \ \cos(\varphi+\alpha_i) - C_d dr(\sin\phi + \alpha_i)] \bullet \# \, of \, elements. \tag{1}$$

With an expression for the forces generated by the vehicle, linear and angular accelerations may be determined using Newton's first law.

The next step is to determine the velocities of each of the vehicle's components. Included in these equations are an induced angle of attack, propwash, and downwash, all of which are relevant to rotorcraft aerodynamics. Induced angle of attack comes from the tip vortices that are generated as air wraps itself around the edges of an aerodynamic surface. As a result, a lower amount of lift is experienced by a wing or a tail section. Because helicopters have much smaller lifting surfaces, if any at all as compared to fixed-wing aircraft, the induced angle of attack can be considered very small or even negligible in many cases. The equations for induced angle of attack and its role in the effective angle of attack are given in 2 and 3. The effective angle of attack is the net resultant angle of attack that a surface feels after considering dynamic losses and gains in lift due to various concepts in aerodynamics.

$$\alpha_{effective} = \alpha_{geometric} - \alpha_i \, . \tag{2}$$

$$\alpha_i = \frac{C_L}{\pi AR} \, . \tag{3}$$

The equation for propwash, which is especially important in the case of a helicopter, comes from the disk method used in momentum theory. The prop, or rotor in this case, is treated as a continuous disk and is sliced into concentric rings. Equation 4 is solved in terms of power required.

$$P = \sqrt{\frac{T^3}{2\rho A}} \, . \tag{4}$$

All aerodynamic coefficients are taken from lift-curve slope plots for two-dimensional airfoils. Prandtl–Glauert is used for compressible flow, and the diamond airfoil method is used for supersonic flow. The Prandtl–Glauert equation, a simple correction factor for changes in fluid density at higher speeds, is given in equation 5, where *a* represents the lift-curve slope.

$$a_{comp} = \frac{a_{incompressible}}{\sqrt{1-M^2}} \, , \ \text{and} \ a = {dC_L}/{d\alpha} \, . \tag{5}$$

As of version 8.60, X-Plane had been using a rather unconventional reference system in relation to the accepted norm in aerospace science and engineering. In X-Plane, V_x and V_z are velocities in reference to the surface of the earth while rate of climb is given as V_y. In the context of this report, the traditional aerospace convention will always be used, where Vx and Vy are in-plane velocities, and Vz is the climb and descent rate.

2.2 Proportional Integral Derivative Control Scheme

As is the case with most general control problems, there are a number of ways to implement a solution for acceptable controllability for a system. The only real requirement is to have a system that sends commands to each of the four main helicopter control outputs, which include lateral cyclic, longitudinal cyclic, tail rotor collective, and main rotor collective. Different combinations of these controls can change the performance of an aircraft depending on its mode of flight.

The proportional integral derivative (PID) used here includes a double-loop system for hover control and a triple-loop system for forward flight. The hover controller uses an angular orientation as the innermost loop and adds a body-frame velocity feedback loop to control the vehicle. The forward flight controller adds a global position controller as the outermost loop, similar to the approach described in Shim (*15*). To attempt to design a collective controller with acceptable response, a complete PID controller with K_p, K_d, and K_i gains is used. The main rotor collective receives feedback from the vehicle's measured altitude, *h*, and the vehicle's measured climb speed, *Vz*.

$$Collective = K_p h + K_d dh + \int K_i h dh \; + K_p Vz + K_d dVz + \int K_i Vz dVz. \tag{6}$$

In principle, terms 2 and 5 in equation 6 overlap, and terms 1 and 6 overlap, but because these values will eventually come from different sensors, it is useful to track and respond to them separately.

Controls of the lateral and longitudinal cyclic pitch of the main rotor are given in equations 7 and 8.

$$Longitudinal\ cyclic = K_{p\theta}\theta + K_{d\theta} d\theta + \int K_{i\theta}\theta d\theta + K_{pVx} Vx + K_{dVx} dVx + \int K_{iVx} Vx dVx\,, \tag{7}$$

and

$$Lateral\ cyclic = K_{p\phi}\phi + K_{d\phi} d\phi + \int K_{i\phi}\phi d\phi + K_{pVy} Vy + K_{dVy} dVy + \int K_{iVy} Vy dVy. \tag{8}$$

The final control output to be calculated is the tail rotor collective. Equation 9 shows the expression for the PID controller on the tail rotor. In forward flight, yaw attitude control is augmented with a feedback loop to the main rotor lateral control. This gives the system a faster overall response and limits the amount of work required by the tail rotor to maintain heading.

Feeding yaw attitude to the main rotor lateral cyclic control also improves long-period turning and allows the helicopter to turn smoothly like a fixed-wing aircraft. The augmented lateral cyclic control of the main rotor is given in equation 10.

$$Tail\ rotor_{forward\ flight} = K_p\psi + K_d d\psi + \int K_i \psi d\psi\ . \tag{9}$$

$$Lateral\ Cyclic_{augmented} = K_p\phi + K_d d\phi + \int K_i \phi d\phi + K_{p\psi}\psi + K_{d\psi} d\psi + \int K_{i\psi}\psi d\psi\ . \tag{10}$$

Through trial and error in simulation, it becomes evident that eliminating some feedback control improves the performance of the PID controller. In particular, setting the integral term for roll angle to 0 in the case of hover improved the stability of the helicopter. The likely cause is that the helicopter becomes overly constrained with too many controls and needs to have the control algorithm relaxed to regain stability. Another instance that includes eliminating position feedback is hover. As long as the vehicle state is updated to the controller quickly enough, the vehicle is able to maintain sufficient hover status. Considering how most GPS systems have a tendency to retain non-negligible random error in their outputs, even when tracking a single position, maintaining hover without GPS can be considered useful under short-term conditions.

All PID controllers also include low and high cut-offs for the maximum control output. Normalizing maximum control output to 1, cutoffs range from 0.1 to 0.65. The benefit of using cutoffs is to maintain the level of responsiveness generally given with larger gains while also avoiding long-period overcompensation in the controls. Without the cutoffs, the controls will respond quickly with large gains, but they will have trouble maintaining stability within an appropriate time frame. Cutoffs are especially important for the integral feedback controls, which employ an additive feedback that can grow to excessive magnitudes if no bounds are placed on them.

3. Simulation

The basic layout of the simulation environment used in this work was designed by Garcia (*16*) in previous work at the University of South Florida. Using flight data from a commercial version of X-Plane simulator, and feeding the data into the processing and control-modeling environment in MathWorks Simulink, a fully functional simulation environment is realized. This allows the user to design any type of control algorithm necessary and to easily integrate that algorithm into the Simulink model. This also allows the user to experiment anything from waypoint navigation to extreme 3-D competition maneuvers. A sample of the generated PID block set is given in figure 3.

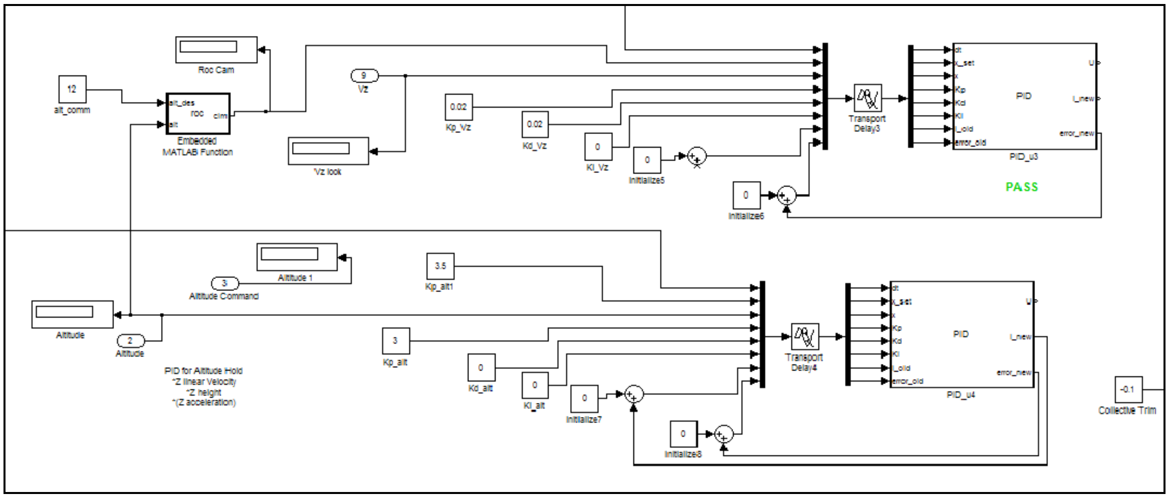

Figure 3. Simulink PID controller block models.

Originally designed around a fuzzy logic controller, the simulator was converted to use PID control laws. Figure 4 shows a generalized view of the components involved in the setup. The loop starts with the X-Plane data-processing center. Data is first provided by X-Plane to the Simulink model using the User Datagram Protocol communication protocol. From there, current vehicle latitude, longitude, angular rates, velocities, and angular orientation are sent through processing blocks. The first of these blocks is responsible for filtering either noise and/or drift from the sensors. The vehicle states are then sent to the second block responsible for transforming data from a fixed, earth-centered coordinate system to the body-fixed coordinate system of the helicopter.

Next, the waypoint-processing station determines the error in position of the aircraft and utilizes this error to decide if the vehicle should move toward its next waypoint, land, or continue its current course of action.

Once the waypoint station has made a decision on what the helicopter's state should be, it passes along its information to the logical controller selector. This block set is responsible for deciding which version of the PID controller should be activated to properly execute the specified flight mode. Thus the PID controller that is active at a given time will change depending on whether the vehicle is to hover, navigate directly to a waypoint, navigate with a user-specified orientation to the next waypoint, land, or end its mission. Generally speaking, the hover PID will be used when the current command is to sustain hover or transition to the next waypoint. Hover is used to change between two waypoints because heading changes occur more smoothly in the hover configuration. The hover PID is also used if the helicopter is on the ground and is told to take off or if it is in the air and is told to land.

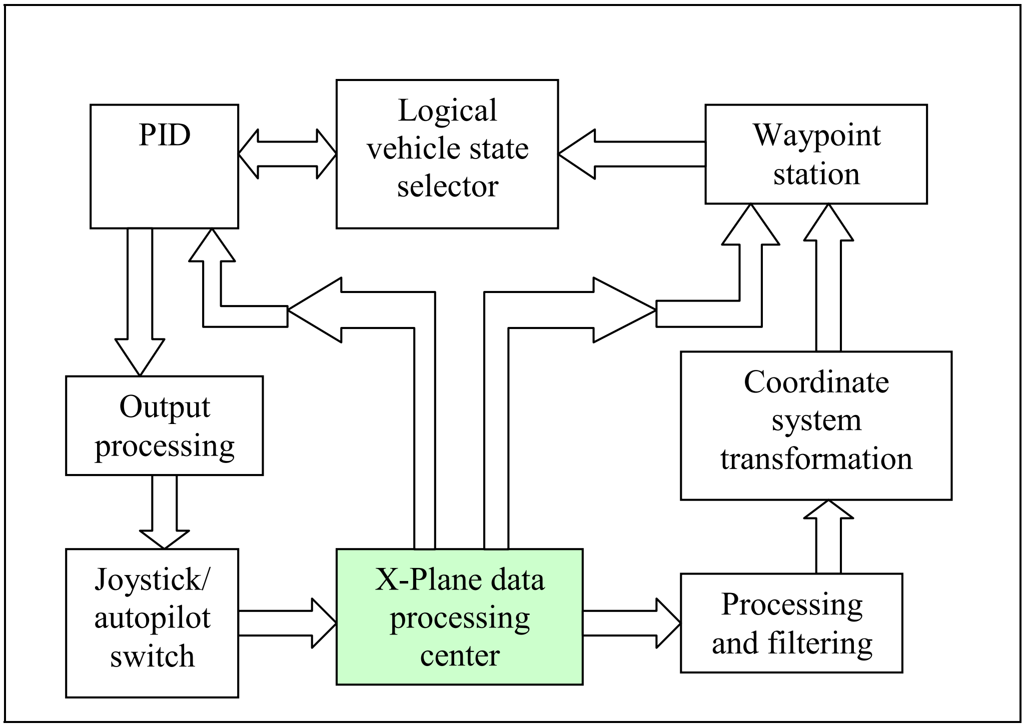
PID
Logical vehicle state selector
Waypoint station
Output processing
Coordinate system transformation
Joystick/ autopilot switch
X-Plane data processing center
Processing and filtering

Figure 4. Generalized diagram of a control system designed in Simulink.

When the selector determines which PID to use, the data input is passed to all the controllers, and a switch on the other side of the PID block chooses only one set of controller outputs to pass into the joystick/autopilot switch. In this block, the user is given the choice of allowing the autopilot to run a mission or take control directly through a radio controller–style computer joystick. This added dimension of controllability turns out to play a major role in the flight-testing process. For instance, if the user sees that the autopilot is not functioning correctly and determines the problem might be in the tail rotor controller, he or she can isolate that one particular controller and manually control it. If the vehicle's response to the autopilot is still problematic even in manual control of that specific control output, the user can deduce that his or her initial hunch was wrong and look elsewhere for a solution.

4. Key Flight Modes

4.1 Trim State

Although PID controllers will eventually settle into a final value after receiving stable gains, there may still be errors present even with a strong Ki term. As such, the helicopter must be trimmed out to satisfy the flight requirements. In the context of this simulation, this means adding constants to the set of control outputs once the PID controller has determined the best output command. In a real flight, this means steady-state deflections to the control surfaces in addition to the commands being sent by the pilot or the autopilot. In the results section of this report, one may notice some steady-state errors in the vehicle orientations. It is accepted in helicopter dynamics that in the trim state, there will need to be a non-zero value for either the roll orientation or the yaw orientation. This model contains an orientation error in the roll direction. The result is that the helicopter will always fly with a nonzero roll attitude.

4.2 Waypoint Navigation

The Waypoint Data Center is a Simulink S-function written in C that considers a number of variables to determine what would be the best next step for the UAS. In takeoff, the waypoint data center considers how close the helicopter is to a commanded altitude. Once the vehicle is within the desired altitude threshold, the system counts how many seconds the helicopter has maintained such an altitude. If the time requirement is met, the Waypoint Center will command the next waypoint. As the helicopter approaches each waypoint, the Waypoint Center continues to monitor heading, altitude, and the difference between desired and actual latitude and longitude coordinates. When all requirements are met for the current waypoint, the waypoint center commands the next location. When the vehicle has passed through all the waypoints, the Waypoint Center sends a command for the helicopter to land. Once the vehicle lands, another command is sent to reverse the pitch in the main rotor to induce reverse thrust. This plants the helicopter to the ground and successfully ends the mission. Figure 5 provides a flowchart of the process during flight.

4.3 Elementary Pilot Control Augmentation System

Another feature of modern autopilots is the ability to augment pilot control inputs. Using preexisting controllers in the simulation, this feature was implemented. The control gains were derived from a modified hover controller. The control scheme consists of augmented climb and descent, which is mutually exclusive to the rest of the control. The other part of the control includes low-speed forward, backward, left, and right flights, as well as vehicle yaw. The vehicle response is of the same quality of the secondary forward flight autopilot (method B) as given in the method-A plot in section 5. These flight modes were augmented by the controller based on the pilot's input. If the right joystick is pointed exclusively in any of these four directions, the helicopter will move in the given direction. If the pilot does not cleanly point the joystick in any one direction, the vehicle will default into hover.

5. Results, Discussion, and Conclusion

The helicopter-controller process is one that continues to evolve over time. The following plots, discussion, and figures not only display the final results of the helicopter simulation and experimental setup, but also provide a perspective of the data evolution. This provides a better understanding of possible issues that may arise en-route to the final result.

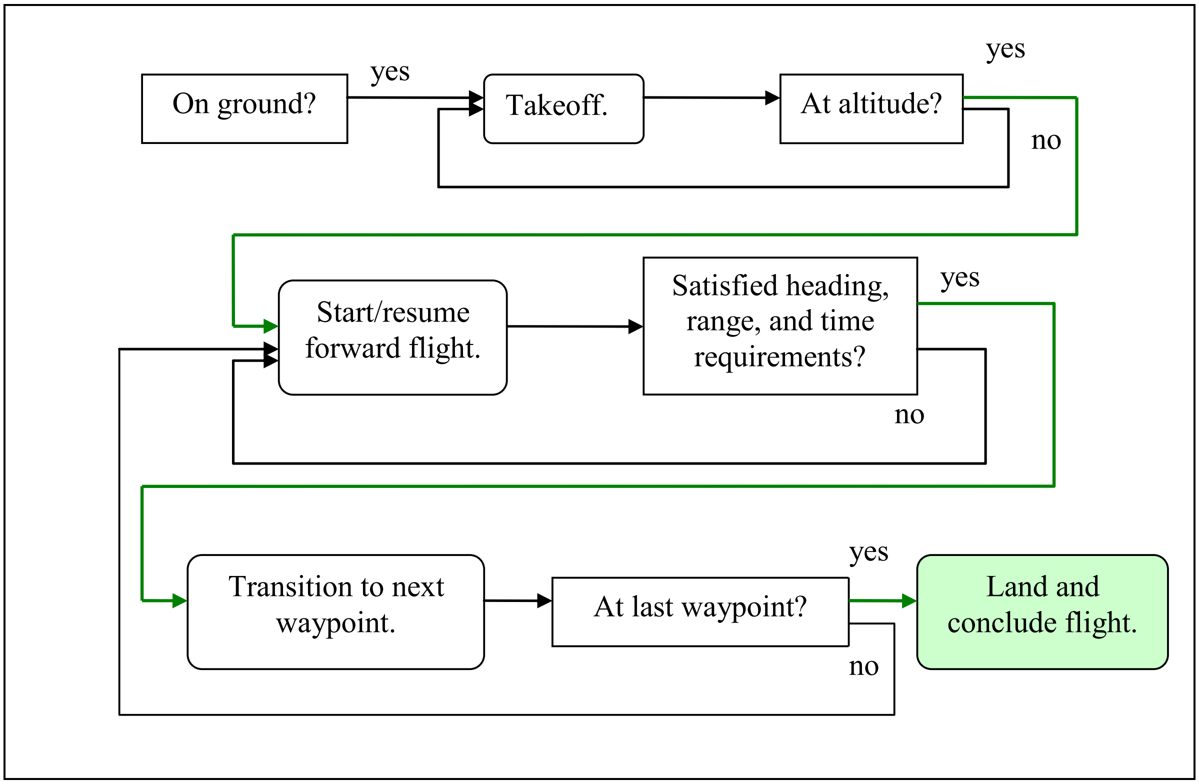
On ground?
yes
Takeoff.
At altitude?
yes
no
Start/resume forward flight.
Satisfied heading, range, and time requirements?
yes
no
Transition to next waypoint.
At last waypoint?
yes
no
Land and conclude flight.

Figure 5. Flowchart of waypoint navigation process.

Figure 6 shows one of the first attempts to achieve hover with the PID controller. This case had no wind but still managed to drift ~205 ft over 70 s, which is extremely insufficient for a practical mission. Later in this section, the hover state results are given for an improved PID system, and the differences are quite noticeable.

Perhaps the single most glaring concern when transitioning a simulated algorithm to a real-life model is the difference sensor noise can make in the performance of the system. In an attempt to address this concern, a white noise generator is attached to the X-Plane data for angular rates and GPS coordinates. The basics of the noise-generating algorithm given in Simulink are explained in Matlab/Simulink (*17*). Equations 11 and 12 also give the expressions for the main variables used to adjust the white noise generator to the desired output. These equations include expressions for the correlation time and the power spectral density.

$$t_c = \frac{1}{100}\frac{2\pi}{f_{max}} \, . \tag{11}$$

$$PSD(\omega) = N_O / 2 \, . \tag{12}$$

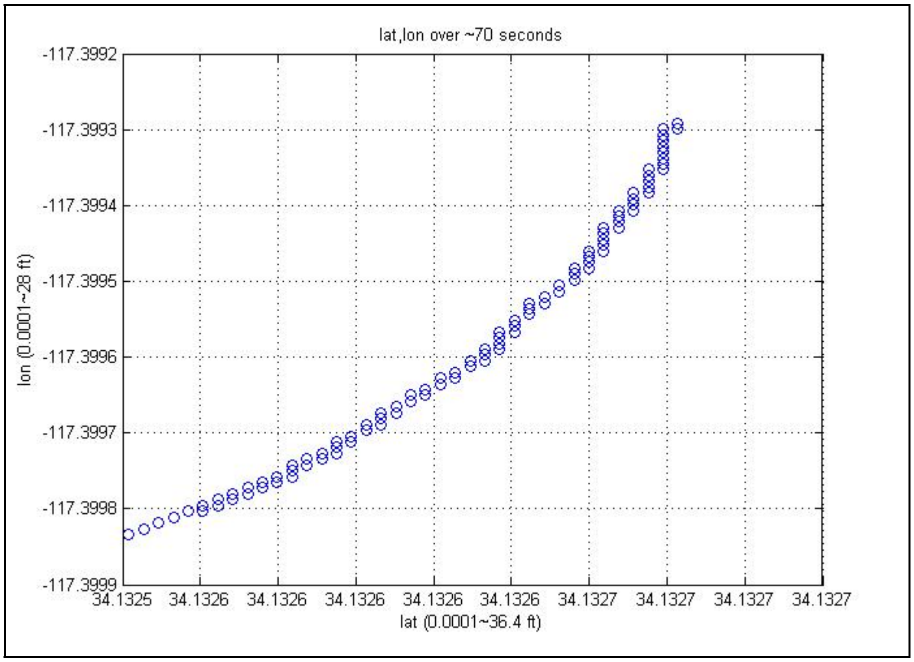

Figure 6. Position hold before PID controller tuning.

The white noise here, by definition, has a power spectrum density that is even across all ranges and a covariance of 0 with the input from the system. White noise is considered a reasonable noise model because of its random nature, regardless of the input properties. Once the white noise has been developed, the next step is to implement some type of filtering to correct the errors. Using the built-in Kalman filters in Simulink, reasonable results may be attained as shown in figure 7.

The basis of the Kalman filter is to model both process and measurement noise and use the time history of the signals to determine the best correction for the data set. The group of fundamental equations is given in table 1, and additional explanation is provided in Grewal and Andrews (*18*). In general, there are four steps to using these equations. When given the initial conditions, one can find $P_k(–)$ with $P_{k-1}(+)$, Φ_{k-1}, and Q_{k-1}. Next, the Kalman gain must be found with $P_k(–)$, H_k, and R_k, which are already given. P_k (+) is then obtained with the Kalman gain and $P_k(–)$. Finally, the state-estimate observational update is calculated with the Kalman gain, initial conditions, and system inputs.

Although the capability of simulating noisy sensors and the possible filtration processes have been shown here, this is by no means a comprehensive study on filtering and its role in the UAS design process. The specifics of system noise and filtering is currently outside the scope of this work and better detailed in other sources. A dedicated report on this topic will likely be included as an expansion of this work in the future. For the remainder of this report, any sensor or state data given does not include noise or filtering. Therefore, the data sets can be considered ideal.

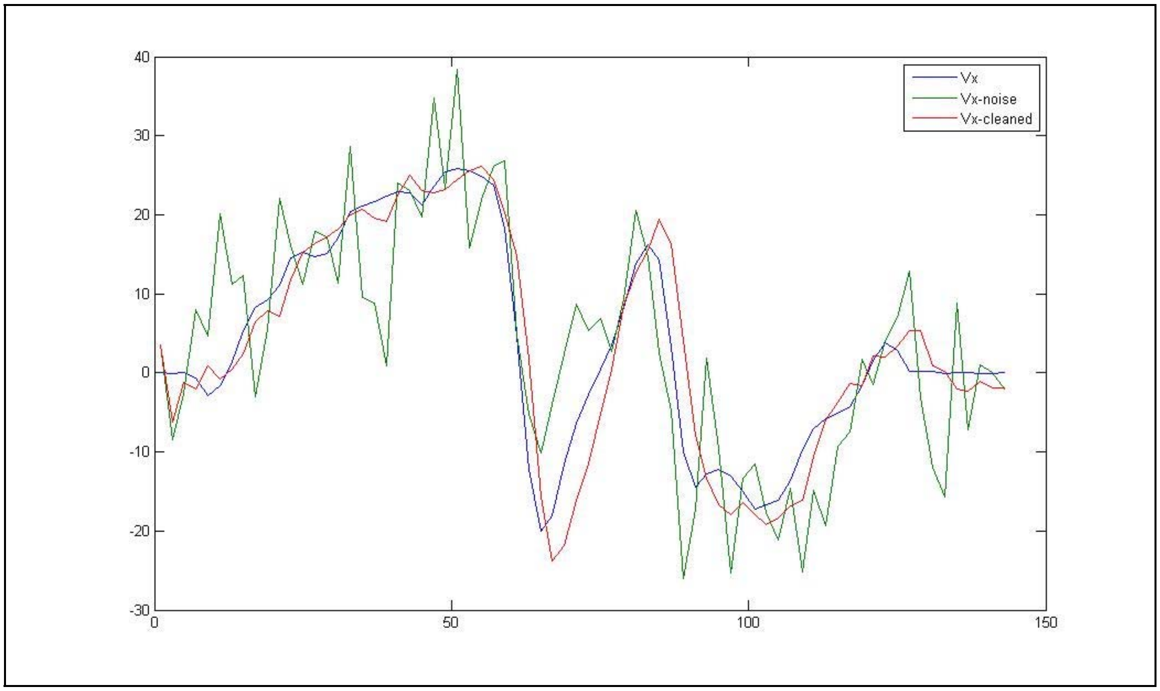

Figure 7. Simulink filtering of body-X velocity reading as might be expected from real-life accelerometers.

Table 1. Fundamental Kalman filter equations.

Characteristic	**Discrete Time Kalman Filter Equations**
System dynamics	$x_k = \Phi_{k-1}x_{k-1} + w_{k-1}$
State measurements	$z_k = H_k x_k + v_k$
Initial conditions	$E\langle x_0 \rangle = \hat{x}_0; \quad E\langle \tilde{x}_0 \tilde{x}_0^T \rangle = P_0$
State estimation	$\hat{x}_k(-) = \Phi_{k-1}\hat{x}_{k-1}(+)$
Error covariance	$P_k(-) = \Phi_{k-1}P_{k-1}(+)\Phi_{k-1}^T + Q_{k-1}$
State estimate observational update	$\hat{x}_k(+) = \hat{x}_k(-) + \overline{K}_k[z_k - H_k\overline{x}_k(-)]$
Error covariance update	$P_k(+) = [I - \overline{K}_k H_k]P_k(-)$
Kalman gain	$\overline{K}_k(+) = P_k(-)H_k^T[H_k P_k(-)H_k^T + R_k]^{-1}$

The hover state stands as a critically important flight mode for the VTOL as one might expect. The experiment process started with tuning the PID controller for a hover condition. Once this step was successfully completed, the process of achieving forward flight became somewhat simplified. Figures 8–13 demonstrate the hovering performance in different scenarios.

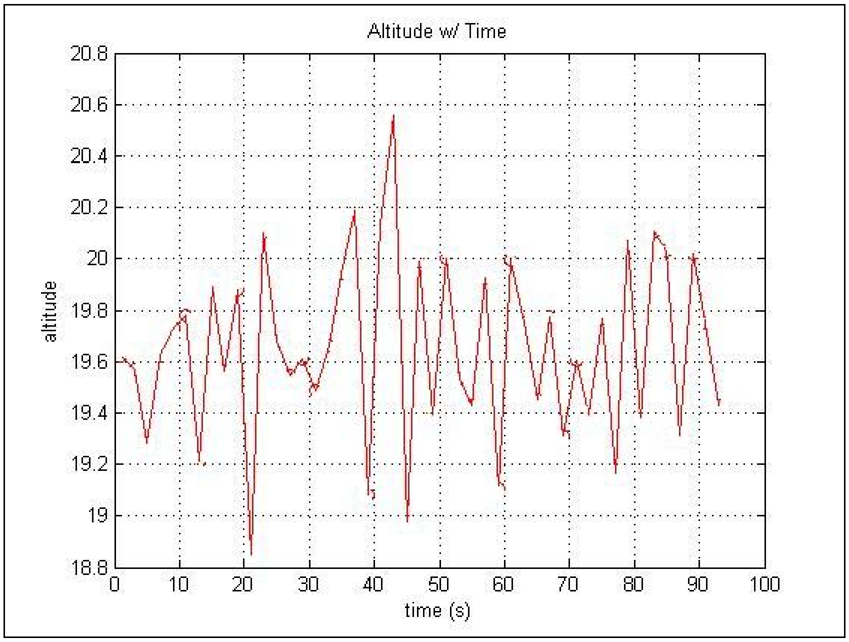

Figure 8. Altitude with time tracking 20 ft with no winds.

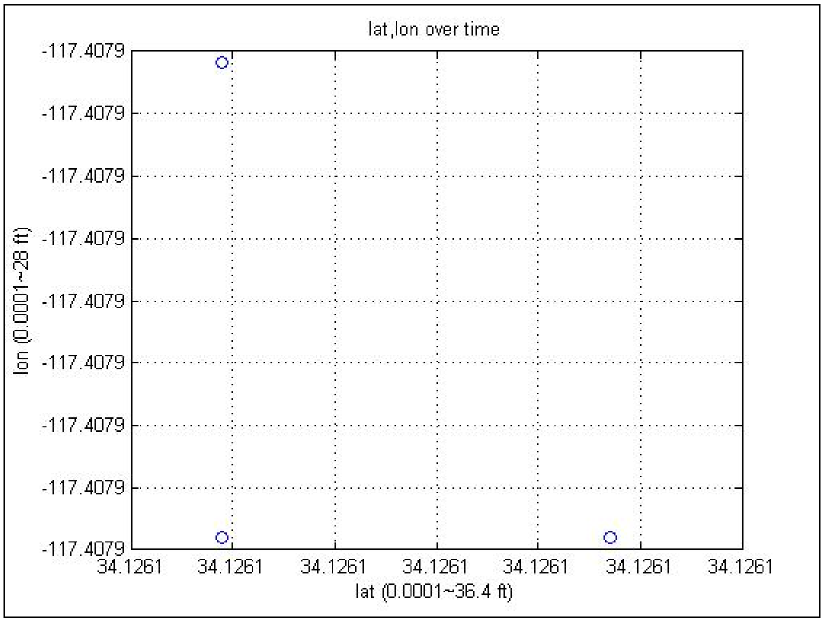

Figure 9. Latitude and longitude with time and no winds.

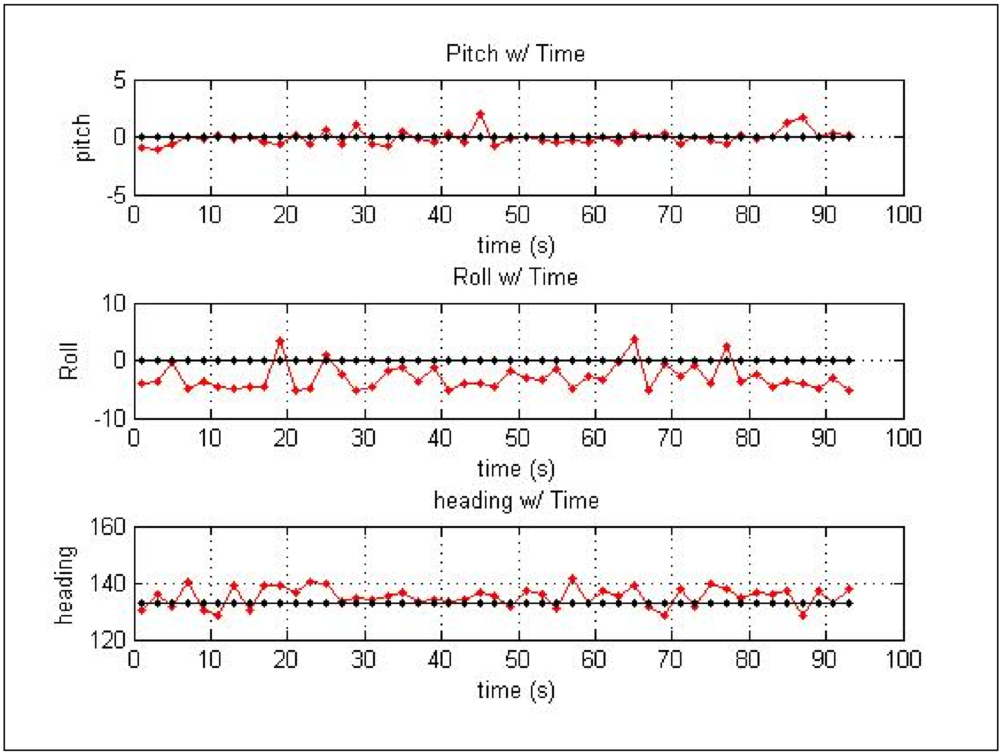

Figure 10. Pitch, roll, and heading with no wind.

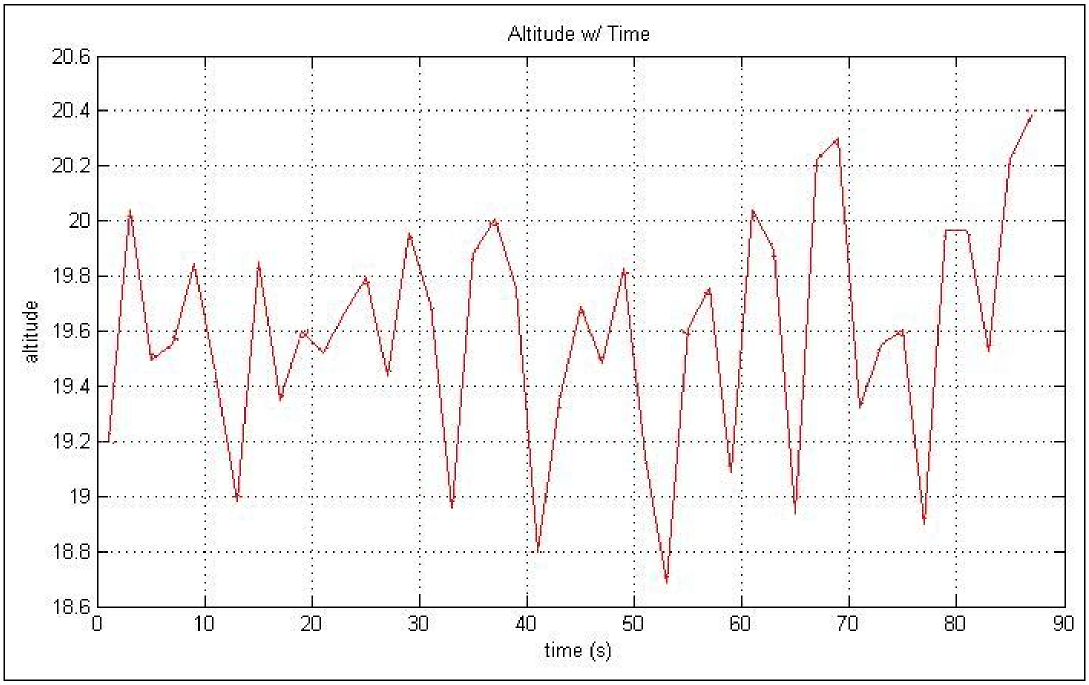

Figure 11. Altitude with time tracking 20 ft in 20-mph winds.

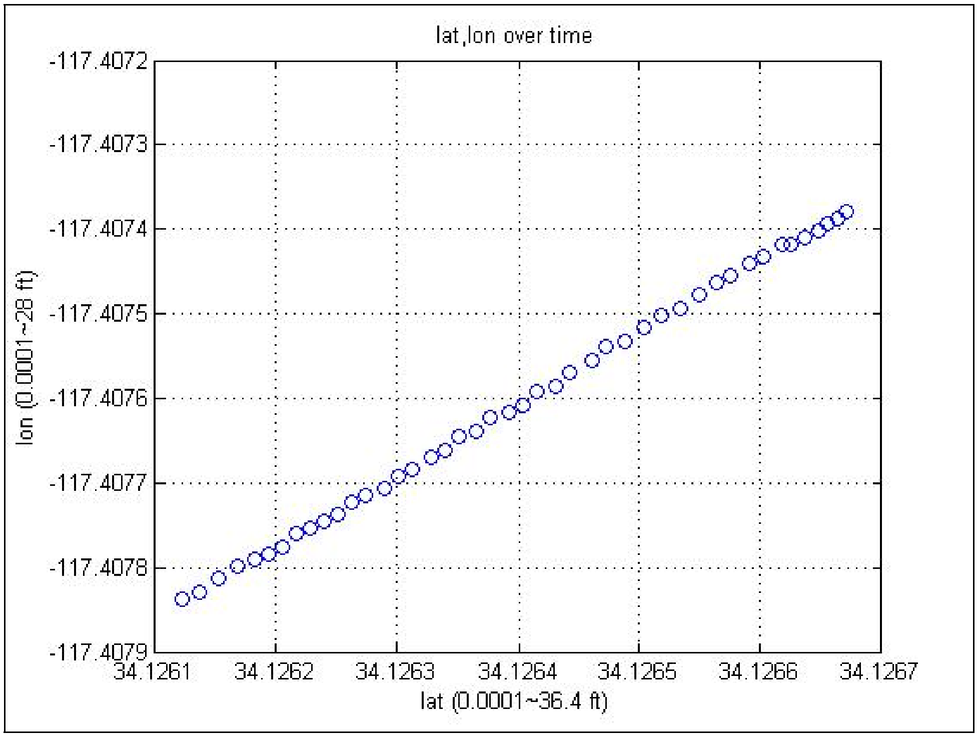

Figure 12. Latitude and longitude with time in 20-mph winds.

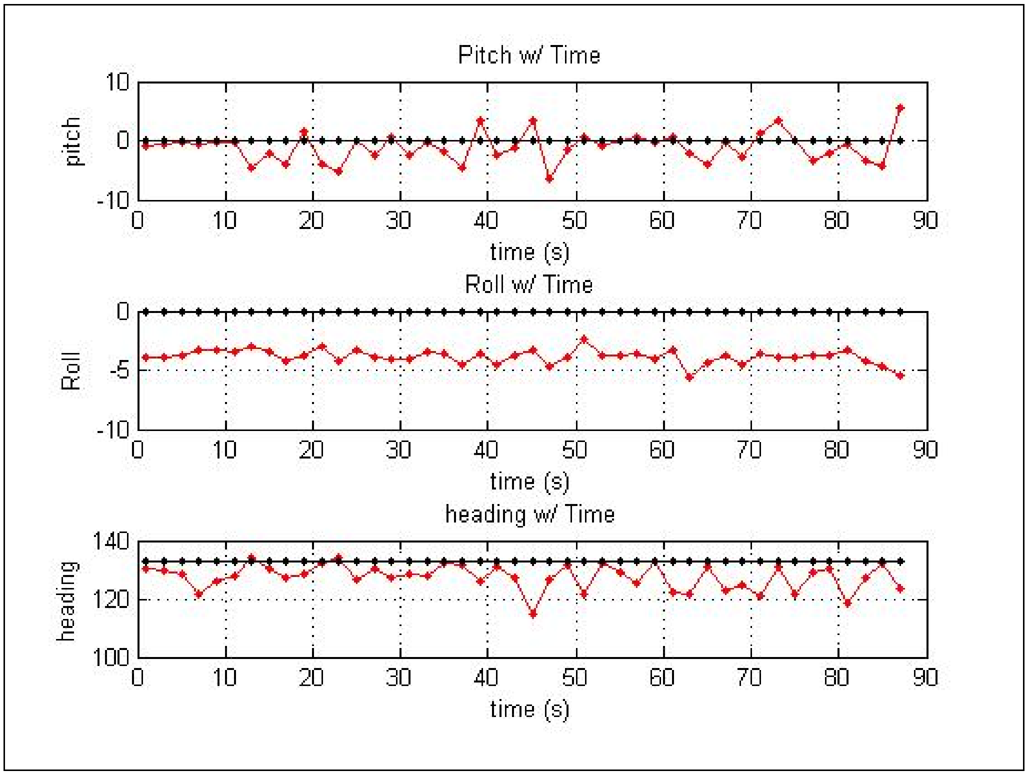

Figure 13. Pitch, roll, and yaw in hover state against 20-mph winds.

The previous figures show conditions for flight with and without wind. Note that the commanded pitch, roll, and yaw in figures 10 and 13 are shown as black lines where the results are shown as red lines. The results show that the responses without wind track very well, especially in the latitude and longitude position. Due to the expected noisy and inaccurate data that will come from an actual GPS, the hover controller does not make use of a position feedback. A controller such as this can be quickly integrated into the Simulink model and be considered acceptable over short periods of time. By setting the controller gains high but also limiting their range of authority with high and low cutoffs, the helicopter is able to maintain nearly dead-on position for 90 s. Figure 9 shows three points that, when looking at the two axis, are given to be virtually the same point. In fact, figure 9 has the same number of data collection points as figures 8 and 10, but their actual values are repeated as one of the three visible in 9. Figures 11–13 show the same helicopter under 20-mph winds. Though the vehicle does well at maintaining its desired heading, its position is much less stable than without the wind conditions. These winds are representative of a level five on the standard Beaufort scale.

Just as an exercise, since the proper implementation of position feedback for a PID controller in hover in an experimental setup has yet to be determined, a modest position feedback in hover for the simulation was implemented. This was done by only including proportional terms of –0.1 and –0.4 to both pitch- and roll-direction error, respectively. To attempt to make the control feedback a bit more akin to a physical autopilot controller, the position updates were reduced to 5 Hz to mimic the traditional update rate of GPS receivers. Figure 14 shows the results over a 200-s time span in X-Plane.

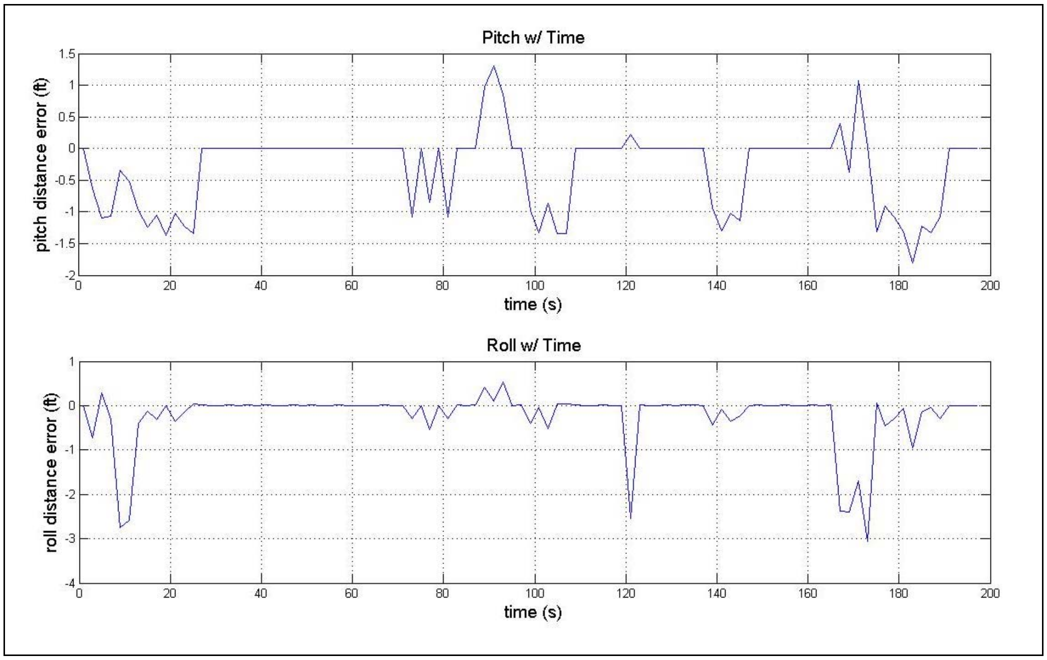

Figure 14. Body direction pitch and roll position error with time during hover.

With waypoint navigation, there are multiple approaches that can be considered valuable. In the case of getting from point to point quickly, either to deliver goods or follow faster ground and air vehicles, approach A is preferred. This approach will be characterized by the helicopter pointing its nose in the direction of the waypoint and tilting its body forward to enter high-speed forward flight. As it approaches the current waypoint, its flight mode will change as it meets some boundary conditions. At this point, the vehicle picks up the next waypoint and adjusts its heading to continue on. The following plot is shown in figure 15.

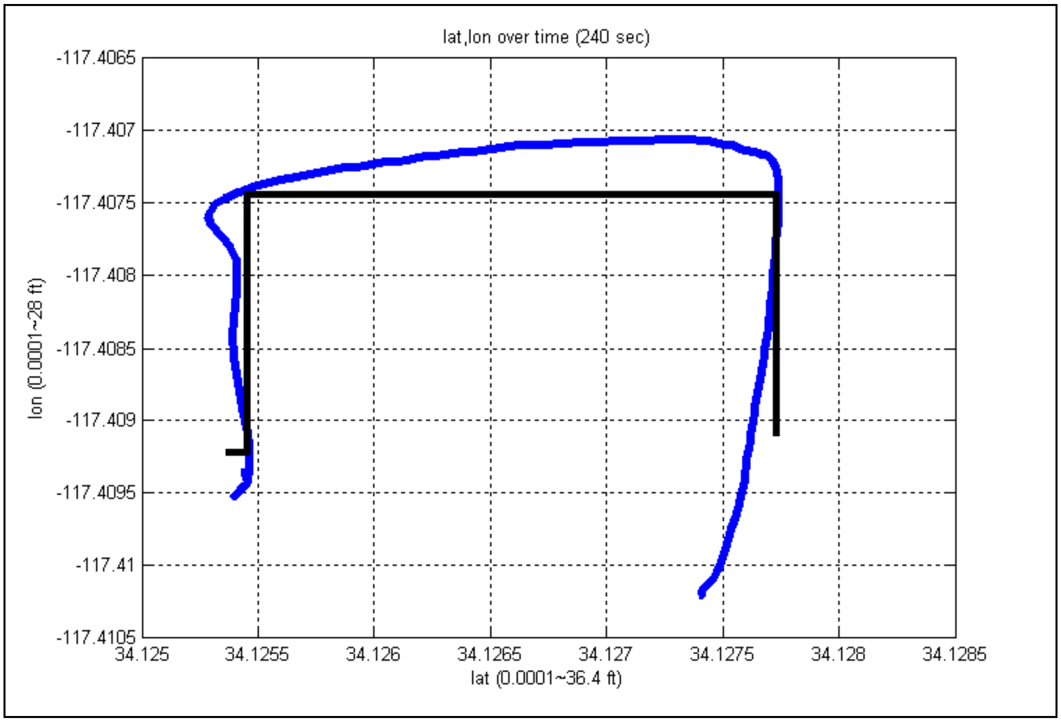

Figure 15. Waypoint navigation with method A.

The second approach is designed for use in close quarters and surroundings. This type of navigation will be designated as approach B. In this mode, the helicopter is allowed to travel to different waypoints at low speed and any given heading. This means the helicopter may have a steady-state yaw attitude with respect to the next waypoint. This will result in much smaller, but still nonzero, roll and pitch attitudes to gain the desired velocity vector in addition to normal steady-state trim conditions. The kink between the third and fourth waypoint in figure 16 is due to the helicopter attempting to correct for its desired heading. The vehicle was commanded to fly backward and experienced some difficulty but was able to recover as shown by the plot. Some of this instability is due to X-Plane's poor treatment of the physics of small-scaled vehicles. In reality, tail control tends to be a bit better for RC helicopters than is given in X-Plane. The extension after the last waypoint in both plots is due to the landing sequence, which is separate from the waypoint navigation.

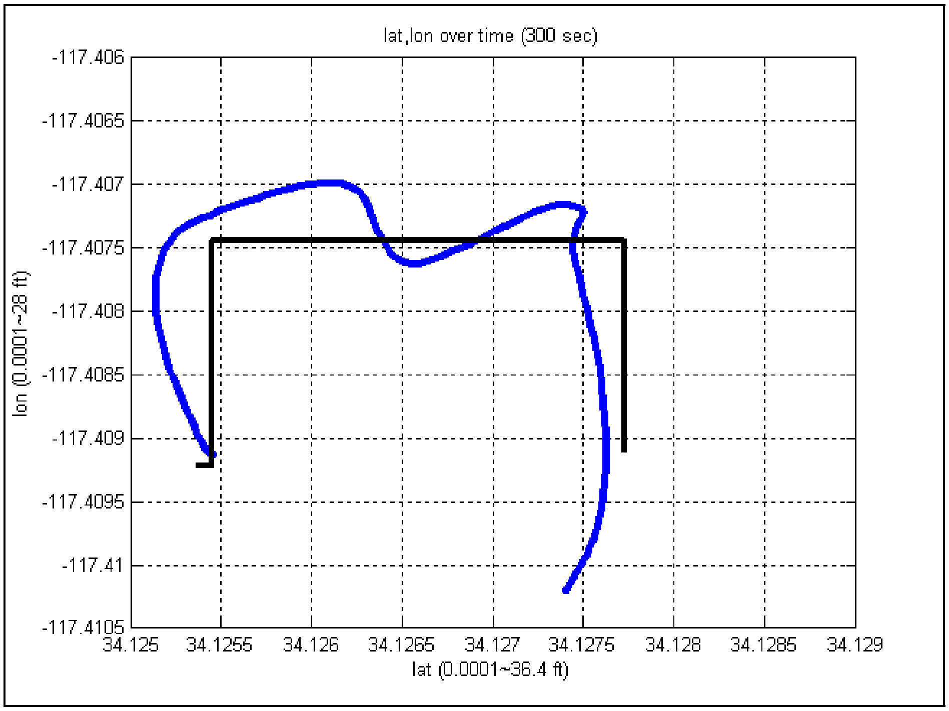

Figure 16. Waypoint navigation as given in figure 14 but with method B.

6. Future Work

6.1 Validation and Long-Term Goals

The design of this simulation was based on having a tool to test control algorithms for actual UASs. Thus as a means to validate the results of these simulations, building an RC helicopter with autopilot hardware and software on board is a necessary step that will follow. Although one may argue that the Federal Aviation Administration–certified status of X-Plane would be enough, RC helicopter setups can vary a great deal even between two instances of the same aircraft design. This, coupled with typical simulation inaccuracies, makes hardware validation a logical step. Perhaps, another method for validation would be a comparison to previous simulations of unmanned aircraft in the X-Plane and/or Simulink or some other comparable environment. Results from a recent literature search have given some indication as to what one should expect in this regard.

As a basic test for validation, the flight controller in this report has been adjusted to control a Kyosho Concept 60SR II Graphite helicopter as described in the simulation and flight-testing section of the dissertation by Shim (*15*). The actual vehicle was also modeled in X-Plane according to the estimated vehicle properties given in table 2.1 of Shim (*15*). The graphic in figure 17 shows the model designed with consideration of all the large onboard avionics and

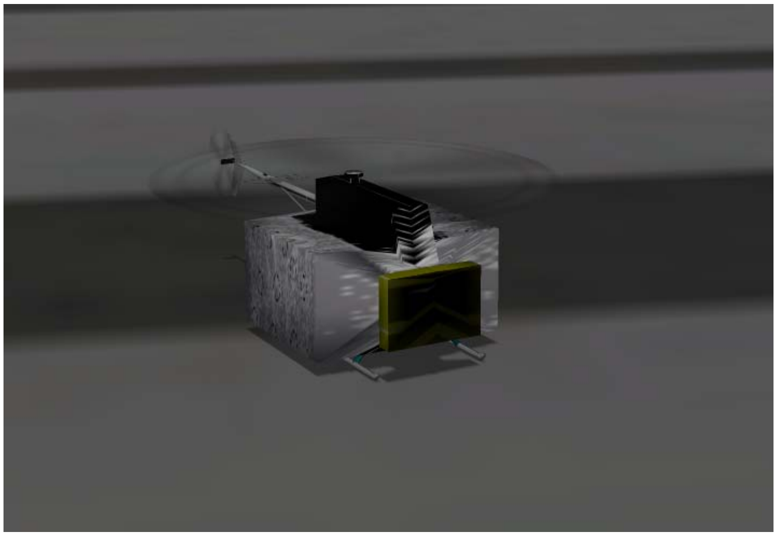

Figure 17. Berkeley Bergen Industrial Twin Minor-2 model in X-Plane.

autopilot equipment typical to a vehicle built in 1999. Results from emulating the flight plan in Shim are given in figures 18 and 19. The object of these figures is to show a similar flight plan as given in the paper by Shim. The relative precision of the flight plan in Shim's paper is about 1 m. Figures 18 and 19 show the autopilot's ability to navigate the modeled Kyosho helicopter to within 4 ft of the desired waypoint, which is on the same order of magnitude as a meter (3.28 ft). The main conclusion is that the controller is adequate enough to follow the waypoints given in the dissertation with the same vehicle parameters also provided in the dissertation. The controller was also able to do this with little or no changes to the gains. Differences in the actual approaches can be attributed to things such as the fact that the PID controller designed in Shim's paper is different than that which has been designed in this report. Additionally, Shim's paper takes advantage of the limited flight envelope used for the waypoint navigation. As a result, he is able to use linearized stability derivatives while the model in X-Plane uses an aerodynamic method with various aerodynamic losses, as mentioned in section 2.1, to generate the flight mechanics involved. The results of this validation thus point at the value of actually testing this controller on an experimental test bed such as an RC helicopter.

In anticipation of the first suggestion for validation as a project, section 6.2 discusses the equipment and expected procedure toward building such a mechanism.

6.2 Final Thoughts and Suggested Approach to Future Goals

Building inexpensive UASs with commercially available equipment was once nearly impossible, but in the last 10 years, this impossibility has become more like a common practice to hobbyist and researchers alike. The general approach is to start with a moderately priced RC vehicle. In this case, the preferred platforms include Thunder Tiger's Raptor and Minicopter's Joker. These vehicular platforms would satisfy the payload requirement of ~5 lb and are readily accessible.

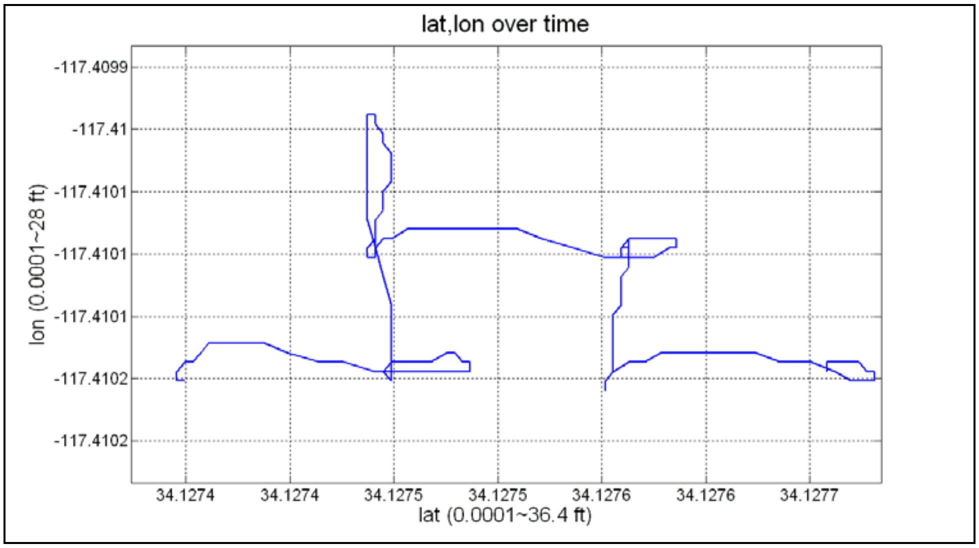

Figure 18. Low-speed, close-range navigation plot of latitude and longitude of Berkeley vehicle in X-Plane/Simulink simulation environment.

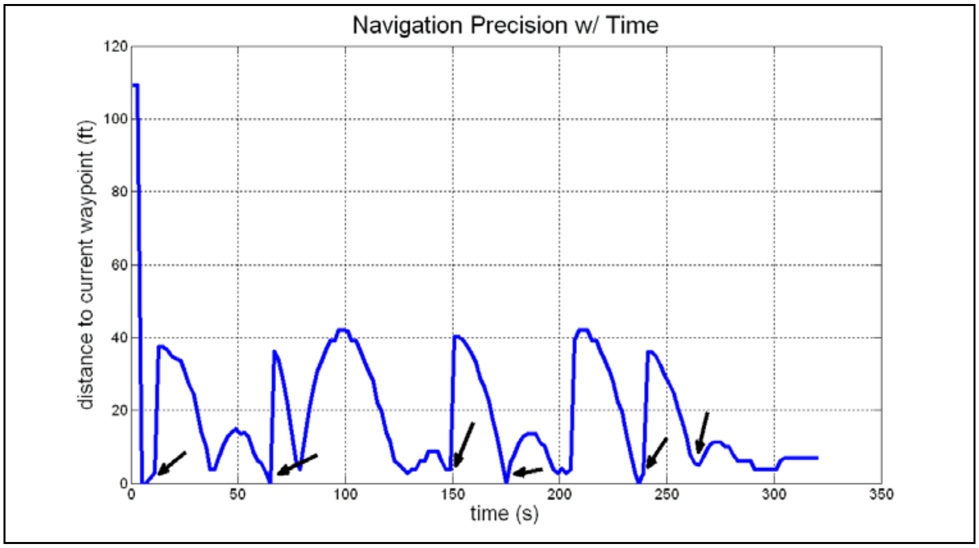

Figure 19. Relative distance to current waypoint (arrows).

The autopilot will likely be programmed in C and placed on a Linux-based onboard PC with a Pico ITX board (see figure 20) (*19*). Other possibilities include loading the autopilot on Qwerk, Gumstix, field-programmable gate arrays, or Microchip PIC* platforms. All of these computational platforms offer a small form factor as well as relatively advanced computational power outputs. There have also been several papers published in recent years documenting the use of such devices on unmanned vehicle test beds. The likelihood of these other platforms being used depends on the specific project and the requirements included with each project.

* PIC is a trademark of the Microchip Company.

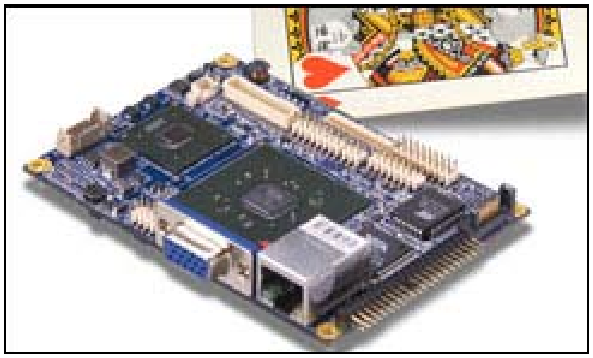

Figure 20. Pico-ITX board form-factor comparison.

Finally, to bring the UAS together, multiple sensors and mounting equipment will be required. This list includes an inertial measurement unit/inertial navigation system sensor, GPS, compass, and altimeter laser and dampers (used to free the sensors of excessive vibrations). After fitting all these items to the UAS platform and completing the proper ground test, validation of the PID simulation will be possible.

Other research that will stem from this report includes a detailed analysis on sensor-filtering techniques, which may include Kalman filtering, some adaptive filtering, or something like Bayesian filtering. This determination depends on what would best fit the data specific to this UAS project. To further improve the performance of the simulator and autopilot, designing some type of intelligent control would be beneficial. This adaptive PID would have the ability to dynamically alter the gains on its own without human input. Overall, the design of this test bed as a research vehicle will allow virtually unlimited possibilities in the future for research. Anything from formation flying, high-level control augmentation, mission-specific scenario simulation, and pursuit and evasion to machine-learning techniques and vision-based navigation will become much more feasible after this step in the Vehicle Technology Directorate's UAS program has been taken.

7. References

1. Pubic Broadcasting Station, Nova Science Programming. Spies That Fly. http://www.pbs.org/wgbh/nova/spiesfly/uavs.html (accessed 19 November 2007).

2. Wikimedia Commons. www.commons.wikimedia.org/wiki/Image:Align_T-Rex_2386_mini.jpg (accessed 20 November 2007).

3. WowWee FlyTech Dragonfly. www.productwiki.com/wowwee-flytech-dragonfly/ (accessed 20 November 2007).

4. Minicopter Canada. www.minicopter-canada.com/ (accessed 20 November 2007).

5. E-flite Advanced Electric Flight. www.e-fliterc.com/Products/Default.aspx?ProdID= EFLH1250 (accessed 20 November 2007).

6. Schiebel. www.schiebel.com (accessed 19 November 2007).

7. Yamaha. www.yamaha-motor.co.jp (accessed 19 November 2007).

8. Boeing. www.boeing.com, X-50A and A-160 (accessed 19 November 2007).

9. UAV Center. www.uavcenter.com/english/wwuavs/north_america/evtol.asp (accessed 19 November 2007).

10. Northrop Grumman. www.is.northropgrumman.com (accessed 20 November 2007).

11. Green, W. E. A Multimodal Micro Air Vehicle for Autonomous Flight in Near-Earth Environments. Ph.D. Thesis, Drexel University, Philadelphia, PA, 2007.

12. Meyer, A. About X-Plane. http://x-plane.com/about.html (accessed 19 November 2007).

13. Nelson, R. C. *Flight Stability and Automatic Control*; McGraw Hill: New York, 1998.

14. Prouty, R. W. *Helicopter Performance, Stability and Control*; Krieger Publishing: Malabar, FL, 2002.

15. Shim, H. D. Hierarchical Flight Control System Synthesis for Rotorcraft-Based Unmanned Aerial Vehicles. Ph.D. Thesis, The University of California-Berkeley, Berkley, CA, 2000.

16. Garcia, R. D. Designing an Autonomous Helicopter Testbed: From Platform Selection to Software Design. Ph.D. Thesis, the University of South Florida, 2008.

17. Matlab and Simulink R2007b. Band-limited white noise help file; The MathWorks: Natick, MA, 2007.

18. Grewal, M. S.; Andrews, A. P. *Kalman Filtering: Theory and Practice*; 2nd ed.; Wiley: New York, 2001.

19. VIA. http://www.via.com.tw/en/initiatives/spearhead/pico-itx/ (accessed 4 December 2007).

INTENTIONALLY LEFT BLANK.

Appendix. Simulink Block Diagrams and Vehicle Specifications for Simulation Tables

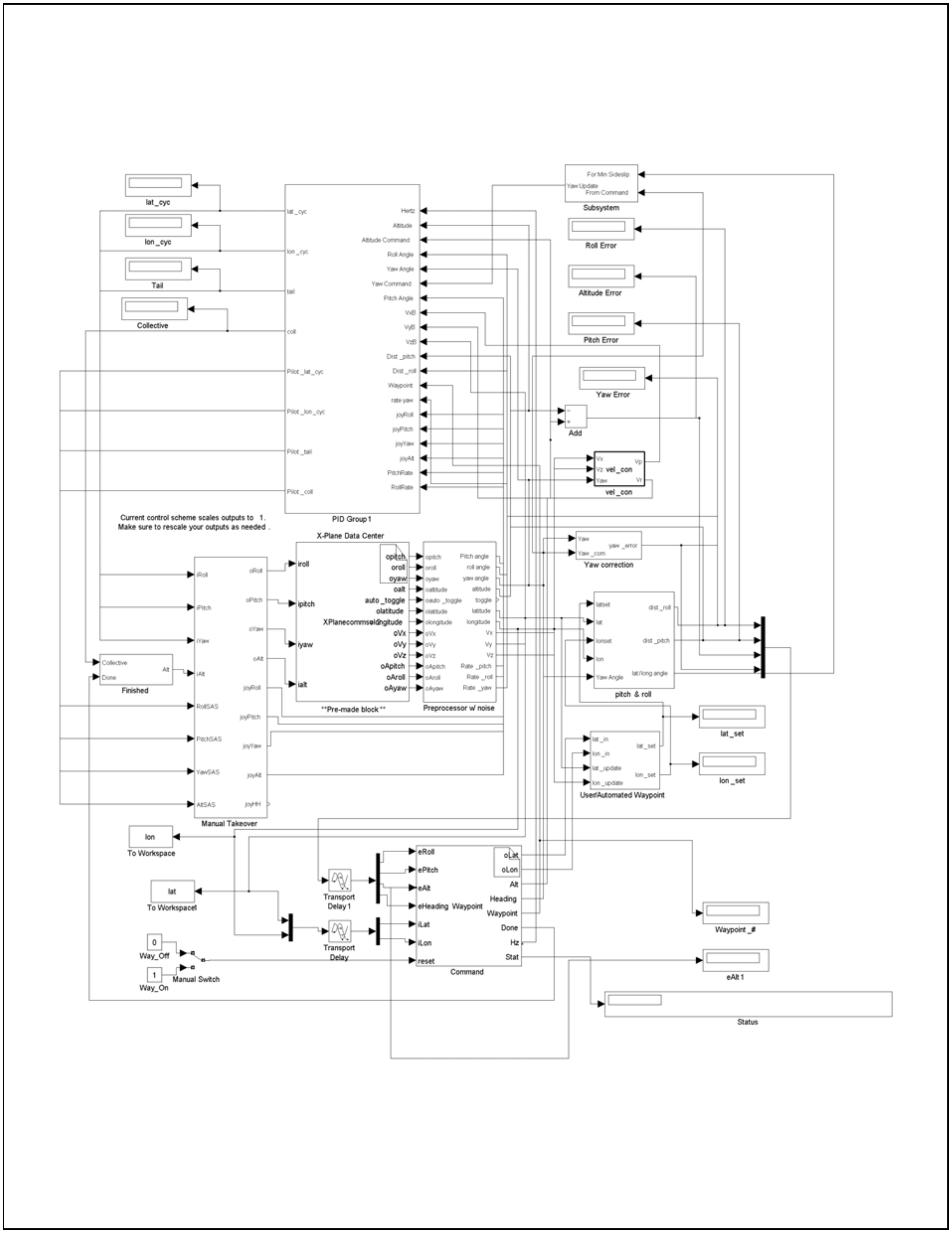

Figure A-1. Complete Simulink block set (subsystems not shown).

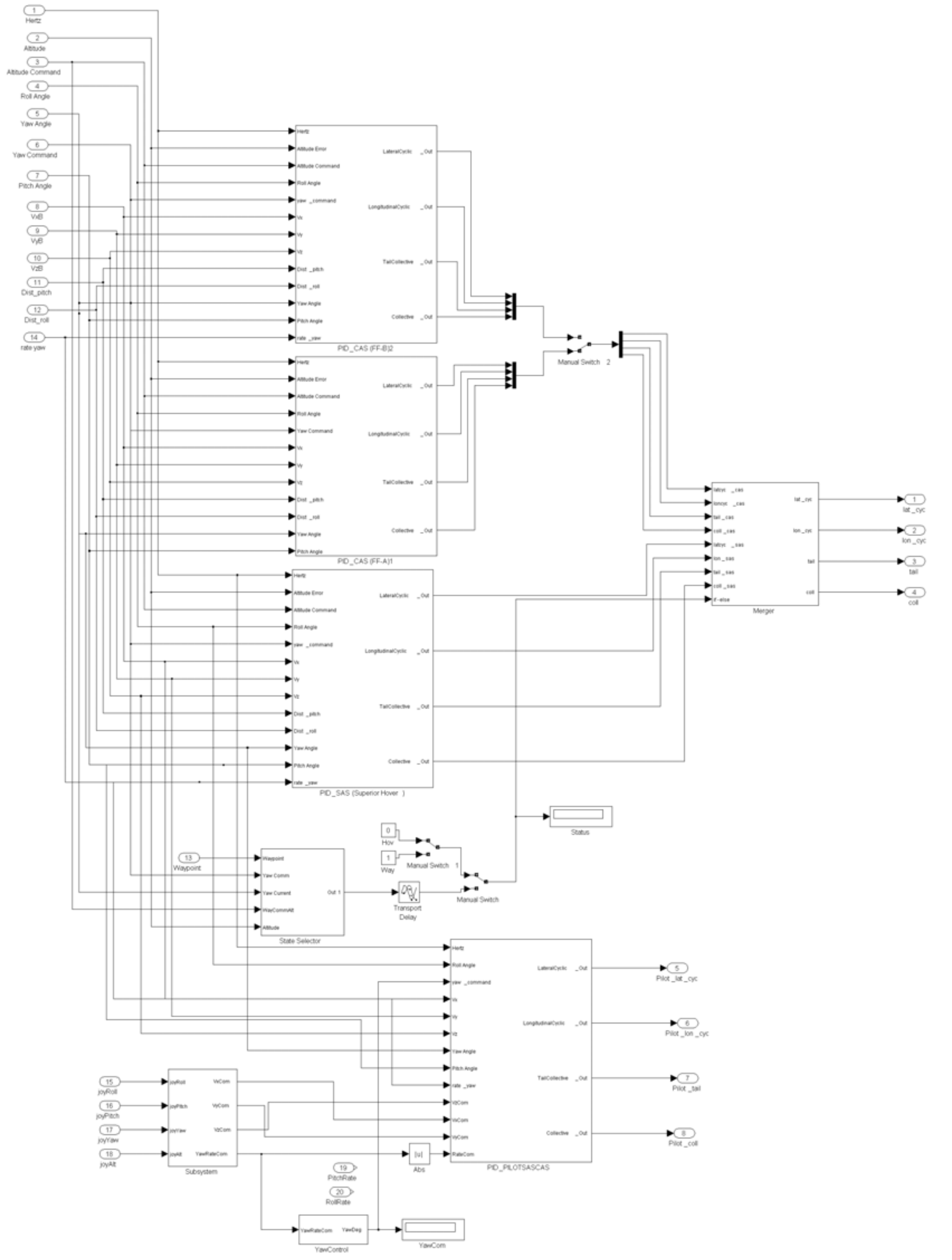

Figure A-2. Proportional integral derivative controller subsystems.

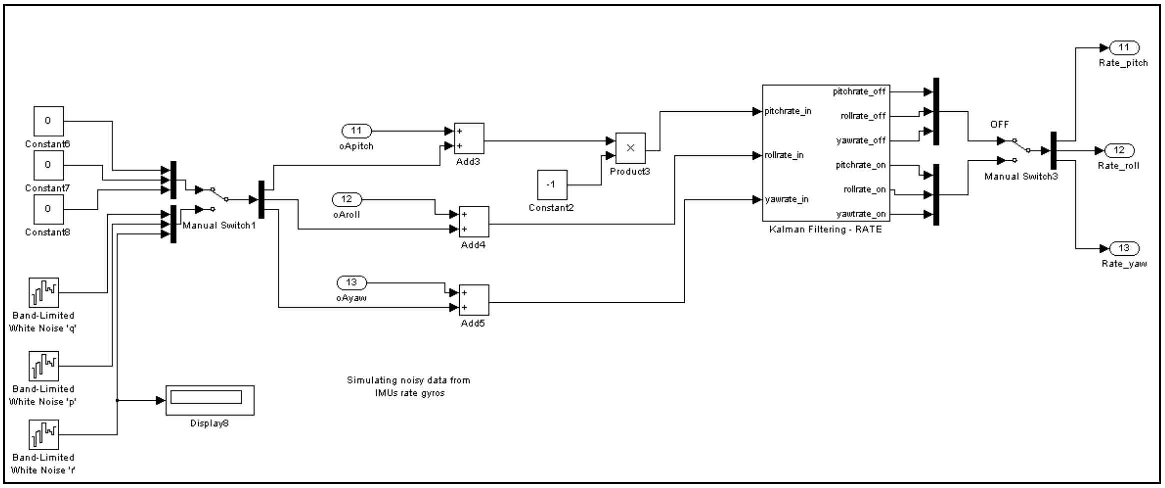

Figure A-3. Sensor noise in simulation.

Table A-1. Helicopter characteristics.

Vehicle Property	**Value**
Main rotor radius	3 ft
Tail rotor radius	0.7 ft
Body max radius	0.5 ft
Max engine power	2.2 hp
Redline rpm (main)	1250
Redline rpm (tail)	5000
Vertical stabilizer length	1.4 ft
Horizontal stabilizer length	1.4 ft
Empty weight	13 lb
Max takeoff weight	16 lb

Table A-2. Control system gains (the University of South Florida model).

Control Gains	Hover	FF-A	FF-B	Pilot CSAS
Kp θ	–1.75	(–)0.1	(–)0.02	–1.75
Kd θ	–0.001	(–)0.0005	(–)0.0005	–0.001
Ki θ	–0.5	(–)0.0001	(–)0.0	–0.5
Kp ϕ	1.2	0.1	0.6	1.2
Kd ϕ	0.0	0.025	0.04	0.0
Ki ϕ	0.0	0.0	0.0	0.0
Kp ψ	0.004	0.0015	0.005	0.004
Kd ψ	0.4	0.0001	0.002	0.4
Ki ψ	0.001	0.08	0.001	0.001
Kp Vx	–0.5	0.0	–0.01	–0.5
Kd Vx	–0.0025	0.0	–0.0025	–0.0025
Ki Vx	–0.2	0.0	0.0	–0.2
Kp Vy	2	0.0	0.1	2
Kd Vy	0.001	0.0	0.1	0.001
Ki Vy	0.004	0.0	0.0	0.04
Kp Vz	0.05	0.005	0.05	2
Kd Vz	0.0015	0.0	0.0015	0.0015
Ki Vz	0.0	0.0	0.0	0.02
Kp h	1.5	0.05	1.5	0.0
Kd h	0.1	0.001	0.1	0.0
Ki h	2	0.0	2	0.0
Kp Coupling	0.0	0.001	0.0	0.0

Note: CSAS = control and stability augmentation system.

Table A-3. Control system gains (Berkeley model).

Control Gains	**Hover**	**FF-A**	**FF-B**	**Pilot CSAS**
Kp θ	–1.75	(–)0.2	–1.75	–1.75
Kd θ	–0.001	(–)0	–0.001	–0.001
Ki θ	–0.05	(–)0.01	–0.5	–0.5
Kp ϕ	1.2	0.3	1.2	1.2
Kd ϕ	0.0	0.035	0.0	0.4
Ki ϕ	0.0	0.0	0.5	0.0
Kp ψ	0.004	0.0015	0.004	0.004
Kd ψ	0.4	0.0001	0.4	0.0004
Ki ψ	0.001	0.08	0.001	0.001
Kp Vx	–0.8	0.0	–0.08	–0.5
Kd Vx	–0.0025	0.0	–0.0025	–0.0025
Ki Vx	–0.2	0.0	–0.2	–0.2
Kp Vy	2	0.0	2	2
Kd Vy	0.001	0.0	0.001	0.001
Ki Vy	0.04	0.0	0.04	0.04
Kp Vz	0.05	0.005	0.05	4
Kd Vz	0.0015	0.0	0.0015	0.0015
Ki Vz	0.0	0.0	0.0	0.02
Kp h	1.5	0.05	1.5	0.0
Kd h	0.1	0.001	0.1	0.0
Ki h	2	0.0	2	0.0
Kp Coupling	0.0	0.001	0.0	0.0

Note: CSAS = control and stability augmentation system.

List of Symbols, Abbreviations, and Acronyms

(–) = priori value

(+) = posteriori value

^ = estimate of specified value

θ = pitch attitude

ψ = yaw attitude

ϕ = roll attitude

Φ = state transition matrix

ρ = density of fluid

λ = inflow coefficient

α = aerodynamic angle of attack

A = specified area

AR = aspect ratio of lifting surface

C = airfoil chord length

C_L = lift coefficient

C_D = drag coefficient

E<> = expected value of that given in angle brackets

H = measurement sensitivity matrix

I = identity matrix

k = current discrete time step

K_p = proportional gain

K_d = derivative gain

K_i = integral gain

M = local Mach number

P = covariance

Q = plant covariance

T = thrust

v = measurement noise

V_E = velocity of element

w = process noise

X = body x coordinate

x = vehicle state

Y = body y coordinate

Z = body z coordinate

z = state measurement

NO. OF COPIES	ORGANIZATION
1 (PDF ONLY)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA 8725 JOHN J KINGMAN RD STE 0944 FORT BELVOIR VA 22060-6218
1	US ARMY RSRCH DEV & ENGRG CMD SYSTEMS OF SYSTEMS INTEGRATION AMSRD SS T 6000 6TH ST STE 100 FORT BELVOIR VA 22060-5608
1	DIRECTOR US ARMY RESEARCH LAB IMNE ALC IMS 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB AMSRD ARL CI OK TL 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB AMSRD ARL CI OK T 2800 POWDER MILL RD ADELPHI MD 20783-1197
	ABERDEEN PROVING GROUND
1	DIR USARL AMSRD ARL CI OK TP (BLDG 4600)

NO. OF
COPIES ORGANIZATION

1 NATIONAL INST OF AEROSPACE
PRES & EXEC DIR
R LINDBERG
100 EXPLORATION WAY
HAMPTON VA 23666

1 NATIONAL INST OF AEROSPACE
VICE PRES OF RSRCH PRGM
C LOWE
100 EXPLORATION WAY
HAMPTON VA 23666

1 NATIONAL INST OF AEROSPACE
DIR OF NASA LANGLEY PRGMS
B WALKLEY
100 EXPLORATION WAY
HAMPTON VA 23666

1 PAT MEIER ASSOCIATES
P MEIER-JOHNSON
38 MILLER AVE
NO 331
MILL VALLEY CA 94941

ABERDEEN PROVING GROUND

4 DIR USARL
AMSRD ARL VT UV
A BROWN
H EDGE
R GARCIA
D WILKERSON