REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
30-05-2008	Final Report	1-Aug-2007 - 30-Apr-2008

4. TITLE AND SUBTITLE
Miniature Rotorcraft Flight Control Stabailization System

5a. CONTRACT NUMBER
W911NF-07-1-0508

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611102

6. AUTHORS
Nathan Slegers

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
University of Alabama - Huntsville
Office of Sponsored Programs
University of Alabama in Huntsville
Huntsville, AL 35899 -

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
53331-MA-II.1

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT
Attitude determination algorithms for a small rotorcraft are proposed and compared. The first algorithm is based on the well known QUEST algorithm used for spacecraft and satellites. Due to large vibration in sensors a pseudo-measurement is developed from gyroscope measurements and rotational kinematics. The pseudo-measurement is used within QUEST to smooth attitude estimations. A second simple gyro-compensated tilt sensor and compass is proposed and compared to the optimal QUEST solution. Both algorithms are shown to have similar performance. The simple gyro-compensated tilt sensor and compass is demonstrated on a small autonomous hovering rotorcraft.

15. SUBJECT TERMS
Unmanned Systems, Rotorcraft, Sensors

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Nathan Slegers
U	U	U	SAR		19b. TELEPHONE NUMBER 256-824-6543

Report Title

Miniature Rotorcraft Flight Control Stabailization System

ABSTRACT

Attitude determination algorithms for a small rotorcraft are proposed and compared. The first algorithm is based on the well known QUEST algorithm used for spacecraft and satellites. Due to large vibration in sensors a pseudo-measurement is developed from gyroscope measurements and rotational kinematics. The pseudo-measurement is used within QUEST to smooth attitude estimations. A second simple gyro-compensated tilt sensor and compass is proposed and compared to the optimal QUEST solution. Both algorithms are shown to have similar performance. The simple gyro-compensated tilt sensor and compass is demonstrated on a small autonomous hovering rotorcraft.

List of papers submitted or published that acknowledge ARO support during this reporting period. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

Number of Papers published in peer-reviewed journals: 0.00

(b) Papers published in non-peer-reviewed journals or in conference proceedings (N/A for none)

Number of Papers published in non peer-reviewed journals: 0.00

(c) Presentations

Number of Presentations: 0.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts): 0

Peer-Reviewed Conference Proceeding publications (other than abstracts):

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts): 0

(d) Manuscripts

Number of Manuscripts: 0.00

Number of Inventions:

Graduate Students

NAME	PERCENT_SUPPORTED
Troy Mitchell	1.00
FTE Equivalent:	**1.00**
Total Number:	**1**

Names of Post Doctorates

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Names of Faculty Supported

NAME	PERCENT_SUPPORTED	National Academy Member
Nathan Slegers	0.10	No
FTE Equivalent:	**0.10**	
Total Number:	**1**	

Names of Under Graduate students supported

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Student Metrics

This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period:	0.00
The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:.....	0.00
The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:.....	0.00
Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):	0.00
Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:	0.00
The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense	0.00
The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields:	0.00

Names of Personnel receiving masters degrees

NAME
Troy Mitchell
Total Number: 1

Names of personnel receiving PHDs

NAME
Total Number:

Names of other research staff

NAME	PERCENT SUPPORTED
FTE Equivalent:	
Total Number:	

Sub Contractors (DD882)

Inventions (DD882)

FINAL TECHNICAL REPORT
W911NF-07-1-0508

Miniature Rotorcraft Flight Control Stabilization System

by:

Nathan Slegers
University of Alabama in Huntsville
Huntsville, Alabama

ABSTRACT

Attitude determination algorithms for a small rotorcraft are proposed and compared. The first algorithm is based on the well known QUEST algorithm used for spacecraft and satellites. Due to large vibration in sensors a pseudo-measurement is developed from gyroscope measurements and rotational kinematics. The pseudo-measurement is used within QUEST to smooth attitude estimations. A second simple gyro-compensated tilt sensor and compass is proposed and compared to the optimal QUEST solution. Both algorithms are shown to have similar performance. The simple gyro-compensated tilt sensor and compass is demonstrated on a small autonomous hovering rotorcraft.

SYMBOLS

$Dist$	=	magnitude of flight vector
d_{lon}, d_{lat}	=	flight distance in the lateral and longitudinal direction
$\dot{d}_{lon}$, $\dot{d}_{lat}$	=	flight position derivatives in the lateral and longitudinal direction
F_F	=	forgetting factor
g	=	Magnitude of earth's gravity
$\vec{g}^B, \vec{g}^I$	=	gravity vector expressed in the body and inertial frames
int_{lon}, int_{lat}	=	lateral and longitudinal integral error
int_{err}, int_{Alt}	=	position and altitude integral error
int_x, int_y	=	integral error of position vector in the body frame
K_P, K_I , K_D	=	proportion-integral-derivative gains
K_{lon}, K_{lat}	=	longitudinal and lateral cyclic control to flap gains
K_β	=	main rotor hub torsional stiffness
K_λ	=	rotor wake intensity factor
lon_{point}	=	longitudinal point of helicopter from GPS data
lat_{point}	=	lateral point of helicopter from GPS data
$\vec{m}^B, \vec{m}^M, \vec{m}^I$	=	earth's magnetic vector in the body, M, and inertial reference frames
Opt_{Tail}	=	optimum tail control
Opt_{Tail_MAX}	=	maximum limit for optimum tail saturation
$Position_1$	=	desired longitudinal GPS position
$Position_2$	=	desired lateral GPS position
p, q, r	=	vehicle angular (roll, pitch, and yaw) rates expressed in the rotorcraft body reference frame
r_x, r_y	=	position vector in the body frame
$\dot{r}_x$, $\dot{r}_y$	=	velocity vector in the body frame
T_{IB}	=	Transformation matrix from the inertial reference frame to the rotorcraft body reference frame
T_{IM}	=	Transformation matrix from the inertial reference frame to the intermediate M reference frame
T_{MB}	=	Transformation matrix from the intermediate M reference frame to the body reference frame
$Tail_{Switch}$	=	bandwidth of tail control
u_{lon}, u_{lat}	=	longitudinal and lateral cyclic control inputs
u_{col}	=	main rotor collective pitch control input
x, y, z	=	position vector components of the center of mass expressed in the inertial reference frame
z_{error}	=	altitude error in the body frame
z_{MAX}	=	maximum limit for altitude saturation
z_{TR}	=	vertical tail offset from the center of gravity along z axis
$\vec{\omega}_{B/I}$	=	Angular velocity of the rotorcraft with respect to the inertial frame.
φ, θ, ψ	=	Euler roll, pitch, and yaw angles
φ_{True}	=	roll angle from desired position vector
φ_{error}	=	roll error in the body frame
θ_{True}	=	pitch angle from desired position vector
θ_{error}	=	pitch error in the body frame
ψ_{DES}	=	desired yaw orientation

I. INTRODUCTION

Autonomous rotorcraft are providing improved capability in performing a diverse set of military missions such as reconnaissance, targeting, border patrol and environmental sensing. The range of applications envisioned

for future micro rotorcraft in both the civilian and military sectors is truly staggering: reconnaissance, automated targeting, sensor emplacement, monitoring, surveying, weapons compliance, hostage release, urban maneuvering, battle damage assessment, etc. A common difficulty in applying miniature rotorcraft to these areas is the complexity and specialization of the control. Inexpensive commercial autopilots for unmanned vehicles are often designed for fixed wing aircraft having different sensor requirements than small rotorcraft. In general, rotorcraft have extreme vibration that make miniature inertial measurement difficult. Larger systems can filter fast dynamics to minimize the impact of high frequency vibrations and noise; however miniature rotorcraft require sensors with higher frequency content in order to respond to their fast dynamics. Typical sensors include MEMS accelerometers which are sensitive to vibration. Inclusion of alternative and or redundant sensors may be used to reduce vibration sensitivity and add useful additional feedback.

The estimation of orientation using multiple vector measurements is a standard attitude determination problem used by spacecraft and satellites. This problem can be separated into two types of solutions; deterministic and optimal algorithms. Deterministic algorithms use two measurement vectors (four pieces of information) and discard part of the measurements so that the orientation parameters can be found. A common deterministic algorithm is the TRIAD algorithm (TRI-axial Attitude Determination system). The algorithm was originally developed by Black [2] with no name given. The method was later published by Lerner [3] under the name "algebraic method." The algorithm then appears as the TRIAD algorithm in [4]. The TRIAD algorithm uses two measurements, the first assumed to be more reliable than the second. Two right-handed orthornomal triads are formed and used to find the desired transformation matrix. Drawbacks of the TRIAD algorithm are that it accommodates only two observations, and some accuracy is lost because part of the second measurement is discarded.

Optimal attitude determination algorithms differ from deterministic algorithms by computing a best estimate which minimizes a loss function *J*. A general statement of the optimal attitude determination problem was first posed by Wahba [5]. A solution was later posed by Wahba et al [6]. However, it was Davenport and Keat (see ref [4] and [7]) who used a quaternion representation that leads to an eigenvalue equation solution called the q-method. Shuster later developed an efficient approximate method to the q-method called QUEST [4] that allows the approximation of the optimal quaternion without solving the eigenvalue problem.

The work reported here evaluates the well know attitude determination algorithms for use on small rotorcraft. In order to reduce noise from vibration two alternative methods are proposed and evaluated. The first alternative

algorithm adds into QUEST a pseudo-measurement vector derived from rotational kinematics. As a comparison, a simple gyro-compensated tilt sensor and compass is also developed. Finally, the orientation algorithms are tested on a small autonomous hovering rotorcraft.

II. ROTORCRAFT ORIENTATION

The orientation of a rotorcraft can be defined by the standard aerospace sequence of body fixed rotations using Euler angles. The inertial axis has the I_I-axis horizontal to ground directed north, the J_I-axis also horizontal and pointing east and K_I directly down. The first rotation is by the angle ψ about the K_I axis resulting in a M frame, followed by a second rotation θ about the J_M axis resulting in a N frame. Finally, a rotation φ about the I_N axis results in the body frame B. The individual transformations for each rotation are provided in Eqs. (1) to (3) where the common shorthand notation for trigonometric functions is employed where $\sin(\alpha) \equiv s_\alpha$, $\cos(\alpha) \equiv s_\alpha$ and $\tan(\alpha) \equiv t_\alpha$.

$$T_{IM} = \begin{bmatrix} c_\psi & s_\psi & 0 \\ -s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

$$T_{MN} = \begin{bmatrix} c_\theta & 1 & -s_\theta \\ 0 & 1 & 0 \\ s_\theta & 0 & c_\theta \end{bmatrix} \tag{2}$$

$$T_{NB} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\phi & s_\phi \\ 0 & -s_\phi & c_\phi \end{bmatrix} \tag{3}$$

The complete transformation from the inertial frame I to the body frame B is given as T_{IB} where,

$$\begin{aligned} [T_{IB}] &= [T_{NB}][T_{MN}][T_{IM}] \\ &= [T_{MB}]\,[T_{IM}] \end{aligned} \tag{4}$$

$$T_{MB} = \begin{bmatrix} c_\theta & 0 & -s_\theta \\ s_\phi s_\theta & c_\phi & s_\phi c_\theta \\ c_\phi s_\theta & -s_\phi & c_\phi c_\theta \end{bmatrix} \tag{5}$$

$$T_{IB} = \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\phi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & s_\phi c_\theta \\ c_\phi s_\theta c_\psi + s_\phi s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi & c_\phi c_\theta \end{bmatrix} \tag{6}$$

Using the transformation T_{IB} in Eq. (6) the components of a vector in I and B can be related. Consider specifically the gravity vector $\vec{g}$

$$\vec{g}^{\,I} = \begin{Bmatrix} g_{XI} \\ g_{YI} \\ g_{ZI} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ g \end{Bmatrix} \tag{7}$$

$$\vec{g}^{\,B} = \begin{Bmatrix} g_{XB} \\ g_{YB} \\ g_{ZB} \end{Bmatrix} = \left[T_{IB}\right]\vec{g}^{\,I} = g\begin{Bmatrix} -s_\theta \\ s_\phi c_\theta \\ c_\phi c_\theta \end{Bmatrix} \tag{8}$$

If earth's gravity is measured in B it is straight forward to solve for φ and θ using Eq. (8).

$$\theta = \sin^{-1}\left(\frac{-g_{XB}}{g}\right) \tag{9}$$

$$\phi = \text{atan2}\left(g_{YB}, g_{ZB}\right) \tag{10}$$

From Eqs. (5), (9), and (10) T_{MB} can also be calculated. Two of the three Euler angles have been found from one measurement vector. Since $\vec{g}^{\,B}$ is independent of ψ it is clear we have no information about the third Euler angle. In general, each vector measurement provides only two pieces of information and the third is constrained by the magnitude since an orthonormal transformation preserves magnitude. To find the final Euler angle a second

measurement vector is required. Consider a magnetometer measuring earth's magnetic vector $\vec{m}$ where it is assumed the inertial frame's I-axis is aligned north and horizontal to the earth's surface.

$$\vec{m}^B = \begin{Bmatrix} m_{XB} \\ m_{YB} \\ m_{ZB} \end{Bmatrix} \tag{11}$$

$$\vec{m}^I = \begin{Bmatrix} m_{XI} \\ 0 \\ m_{ZI} \end{Bmatrix} \tag{12}$$

Since, T_{MB} is known from Eqs. (5), (9), and (10) the magnetic vector in the M frame can be found.

$$\vec{m}^M = \begin{Bmatrix} m_{XM} \\ m_{YM} \\ m_{ZM} \end{Bmatrix} = \left[T_{MB}\right]^T \vec{m}^B = \begin{Bmatrix} c_\theta m_{XM} + s_\phi s_\theta m_{YM} + c_\phi s_\theta m_{ZM} \\ c_\phi m_{YM} - s_\phi m_{ZM} \\ -s_\theta m_{XM} + s_\phi c_\theta m_{YM} + c_\phi c_\theta m_{ZM} \end{Bmatrix} \tag{13}$$

Comparing Eq. (13) to $\vec{m}^I$ results in three equations.

$$\begin{Bmatrix} m_{XI} \\ 0 \\ m_{ZI} \end{Bmatrix} = \left[T_{IM}\right]^T \vec{m}^M = \begin{bmatrix} c_\psi & -s_\psi & 0 \\ s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} m_{XM} \\ m_{YM} \\ m_{ZM} \end{Bmatrix} \tag{14}$$

Using only the second equation the final Euler angle ψ is found as

$$\psi = \text{atan2}\left(-m_{YM}, m_{XM}\right) \tag{15}$$

Using two vector measurements ($\vec{g}^B$ and $\vec{m}^B$) the three Euler angles can be found. More specifically, two pieces of information were used to find φ and θ and one piece was used to find ψ. In general, at least two measurement vectors are needed, we can discard one piece of information and use three to solve for the three unknowns or use all four. If all four pieces of information are used, the problem of orientation is over-determined and a least squares solution may be employed.

The orientation solution presented in Eqs. (9,10) and (15) is specific to the Euler angle representation. It is possible to describe the orientation using alternative formulations. In general, we can relate a vector in two frames (In our case I and B) using Eq. (16)

$$\vec{v}_{kb} = \left[T_{ab}\right]\vec{v}_{ka} \tag{16}$$

where, $\vec{v}_{kb}$ is an arbitrary k vector expressed in frame "b", $\vec{v}_{ka}$ is an arbitrary k vector expressed in frame "a", and T_{ab} is the 3 x 3 orthonormal transformation matrix from "a" to "b". The orthornomal transformation matrix can be defined using any valid set of orientation map. Note, in Eq. (6) Euler angles were used to describe T_{IB}. A common alternative to Euler angles is a quaternion representation. Consider a quaternion $\mathbf{q}$ where,

$$\mathbf{q} = \begin{Bmatrix} \vec{q} \\ q_4 \end{Bmatrix} \tag{17}$$

$$\vec{q} = \begin{Bmatrix} q_1 \\ q_2 \\ q_3 \end{Bmatrix} \tag{18}$$

It is well known that the transformation using a quaternion $\mathbf{q}$ can be expressed using the four quaternion components [1].

$$\left[T_{IB}\right] = \begin{bmatrix} 2q_4^2 - 1 + 2q_1^2 & 2\left(q_1q_2 + q_4q_3\right) & 2\left(q_1q_3 - q_4q_2\right) \\ 2\left(q_1q_2 - q_4q_3\right) & 2q_4^2 - 1 + 2q_2^2 & 2\left(q_2q_3 - q_4q_1\right) \\ 2\left(q_1q_3 + q_4q_2\right) & 2\left(q_2q_3 + q_4q_1\right) & 2q_4^2 - 1 + 2q_3^2 \end{bmatrix} \tag{19}$$

Since Eq. (19) uses four quaternion components to express T_{IB} which is determined by only three parameters the quaternion is constrained to have unity magnitude, thus having only three unique parameters.

$$\mathbf{q}^T\mathbf{q} = 1 \tag{20}$$

A third representation is to use the nine direction cosines in Eq. (21). As always T_{IB} is defined by only three parameters. Therefore, in the case of direction cosines six unique constraint equations exist [1].

$$T_{IB} = \begin{bmatrix} l_{11} & l_{12} & l_{13} \\ l_{21} & l_{22} & l_{23} \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \tag{21}$$

Any of the three representations can be used to define orientation (3 Euler angles, Quaternions, or Direction Cosines). However, as shown in the previous example, at least two vector measurements are required to solve the problem.

III. ATTITUDE DETERMINATION

Using Eq. (16) the attitude determination problem can be generalized in the following statement.

Using at least two measurement vectors find the Euler angles, Quaternion, or Direction Cosines that define the orientation matrix.

This problem can be separated into two types of solutions; deterministic and optimal algorithms. Deterministic algorithms use two measurement vectors (four pieces of information), discards part of the measurements so that the orientation parameters can be found, similar to the Euler angle example. The TRIAD algorithm uses two measurements as in Eqs. (22) and (23), the first assumed to be more reliable than the second. .

$$\vec{v}_{1B} = \left[T_{IB}\right]\vec{v}_{1I} \tag{22}$$

$$\vec{v}_{2B} = \left[T_{IB}\right]\vec{v}_{2I} \tag{23}$$

Two right-handed orthornomal triads of vectors $\left\{\vec{t}_{1B}, \vec{t}_{2B}, \vec{t}_{3B}\right\}$ and $\left\{\vec{t}_{1I}, \vec{t}_{2I}, \vec{t}_{3I}\right\}$ are formed where,

$$\begin{aligned} \vec{t}_{1B} &= \vec{v}_{1B} & \vec{t}_{1I} &= \vec{v}_{1I} \\ \vec{t}_{2B} &= \frac{\vec{v}_{1B} \times \vec{v}_{2B}}{\left|\vec{v}_{1B} \times \vec{v}_{2B}\right|} & \vec{t}_{2I} &= \frac{\vec{v}_{1I} \times \vec{v}_{2I}}{\left|\vec{v}_{1I} \times \vec{v}_{2I}\right|} \\ \vec{t}_{3B} &= \vec{t}_{1B} \times \vec{t}_{2B} & \vec{t}_{3I} &= \vec{t}_{1I} \times \vec{t}_{2I} \end{aligned} \tag{24}$$

Using the two triads, the solution to T_{IB} is shown in Eq. (25).

$$\left[T_{IB}\right]=\left[\vec{t}_{1B}\quad \vec{t}_{2B}\quad \vec{t}_{3B}\right]\left[\vec{t}_{1I}\quad \vec{t}_{2I}\quad \vec{t}_{3I}\right]^{T} \tag{25}$$

The solution to T_{IB} satisfies Eq. 22 exactly and in the absence of noise Eq. (25) also satisfies (23) exactly. In the presence of noise (23) will not be satisfied. The TRIAD algorithm is limited to using only two observations, and some accuracy is lost because part of the second measurement is discarded.

Optimal attitude determination algorithms differ from deterministic algorithms by computing a best estimate minimizing a loss function *J*. A general statement of the optimal attitude determination problem was first posed by Wahba [5] in Eq. (26) where, the task is to find a that minimizes *J* using the N measurements.

$$J\left(T_{IB}\right)=\frac{1}{2}\sum_{k=1}^{N}w_k\left|\vec{v}_{kB}-T_{IB}\vec{v}_{kI}\right|^2 \tag{26}$$

Davenport and Keat [4,7] developed the optimal q-method using a quaternion representation leading to an eigenvalue equation solution. The q-method [7] rewrites (26) using quaternions leading to an alternative gain function.

$$g\left(\mathbf{q}\right)=1-J\left(T_{IB}\right)=\mathbf{q}^{T}K\mathbf{q} \tag{27}$$

Where, *K* is a 4 x 4 matrix defined in Eq. (28).

$$K=\begin{bmatrix} S-\sigma I & Z \\ Z^{T} & \sigma \end{bmatrix} \tag{28a}$$

$$B=\sum_{k=1}^{N}w_k\left(\vec{v}_{kB}\vec{v}_{kI}{}^{T}\right) \tag{28b}$$

$$S=B+B^{T} \tag{28c}$$

$$Z=\left[B_{23}-B_{32}\quad B_{31}-B_{13}\quad B_{12}-B_{21}\right]^{T} \tag{28d}$$

$$\sigma=tr\left[B\right] \tag{28e}$$

In order to minimize *J,* Eq. (27) must be maximized taking into the unit quaternion constraint in Eq. (20). The constraint in (20) can be accounted for by appending a Lagrange multiplier to (27) resulting in a new gain function.

$$g'(\mathbf{q}) = \mathbf{q}^T K\mathbf{q} - \lambda\mathbf{q}^T\mathbf{q} \tag{29}$$

Differentiating Eq. (29) leads to a stationary point when Eq. (30) is satisfied.

$$K\mathbf{q} = \lambda\mathbf{q} \tag{30}$$

Equation 30 is recognized as an eigenvalue problem, thus the optimal quaternion is an eigenvector of *K*. Substitution of (30) into (27) yields

$$g(\mathbf{q}) = \mathbf{q}^T\lambda\mathbf{q} = \lambda\mathbf{q}^T\mathbf{q} = \lambda \tag{31}$$

Or $g(\mathbf{q})$ is maximized for the largest eigenvalue of K and $\mathbf{q}_{opt}$ is the corresponding eigenvector. The QUEST algorithm developed by Shuster [4] approximates the optimal q-method solution without solving the eigenvalue problem. QUEST uses the approximation in Eq. (32) to quickly solve for $\mathbf{q}_{opt}$.

$$\lambda_{\max} \approx \sum_{k=1}^{N} w_k \tag{32}$$

Both the TRIAD and QUEST algorithms are well known and have been used extensively for attitude determination of spacecraft and satellites. These systems can use multiple vector measurements from magnetometers, sun sensors, star sensors, and directional antennas. Theoretically, both methods can be used on small unmanned rotorcraft and fixed wing systems. In both cases, earth's gravity and magnetic field can be used as the two measurement vectors. Practically, accelerometers on unmanned systems measure not only earth's gravity but also the systems acceleration and noise. Often significant filtering is used to reduce noise and transient acceleration at the cost of sensor delay. In moderate dynamic systems this approach is often adequate and reasonable estimates can be achieved.

To demonstrate the performance of both the TRIAD algorithm and QUEST on an example small rotorcraft a set of data was generated as shown in Fig. 1. Using the known Euler angles, $\vec{g}^I$, and $\vec{m}^I$ measurement data was generated by adding Gaussian noise with a standard deviation of 0.30 G's onto $\vec{g}^B$ and a standard deviation of 2% the earth's magnetic field onto $\vec{m}^B$ as shown in Fig. 2.

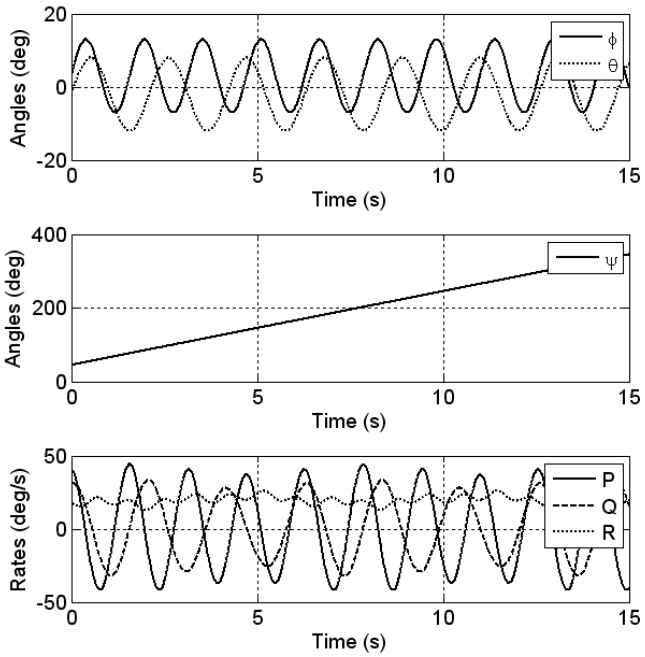


Figure 1. True Orientation and Angular Velocities

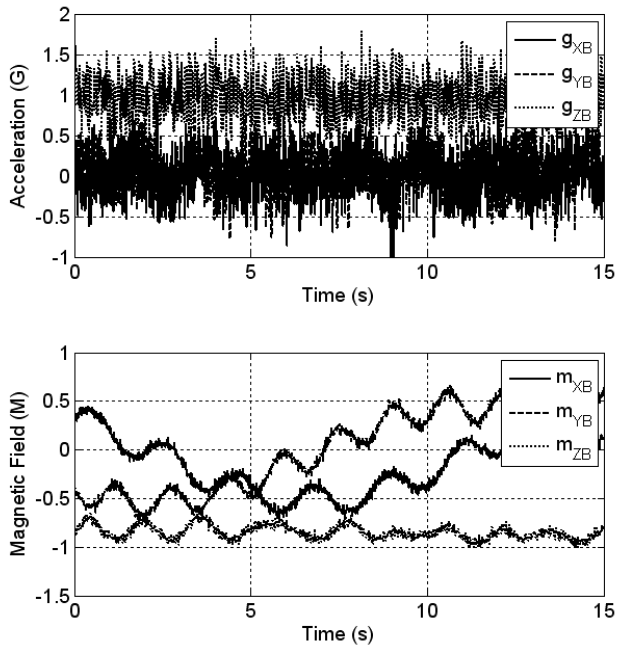


Figure 2. Measurements including noise

Results from the TRIAD and QUEST algorithms are shown in Figs. 3 and 4 with errors in Figs. 5 and 6. Both the acceleration and magnetometer measurements have been pre-filtered with a first-order filter having a time

constant of 0.125 seconds. As expected, the large high frequency errors from the poor sensors results in large error of near 10 deg for both φ and θ and 20 deg for ψ. Also as expected, QUEST performs slightly better than the TRIAD algorithm because of the optimal use of measurement information. However, in both cases the estimates are poor. Improvements could be made by filtering the data more; however, the increased delay in estimation becomes problematic.

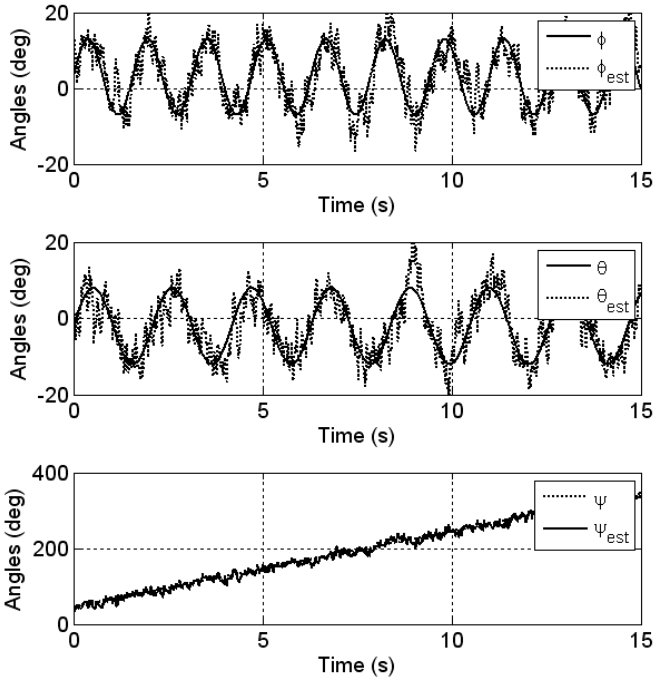


Figure 3. TRIAD Euler angle estimates

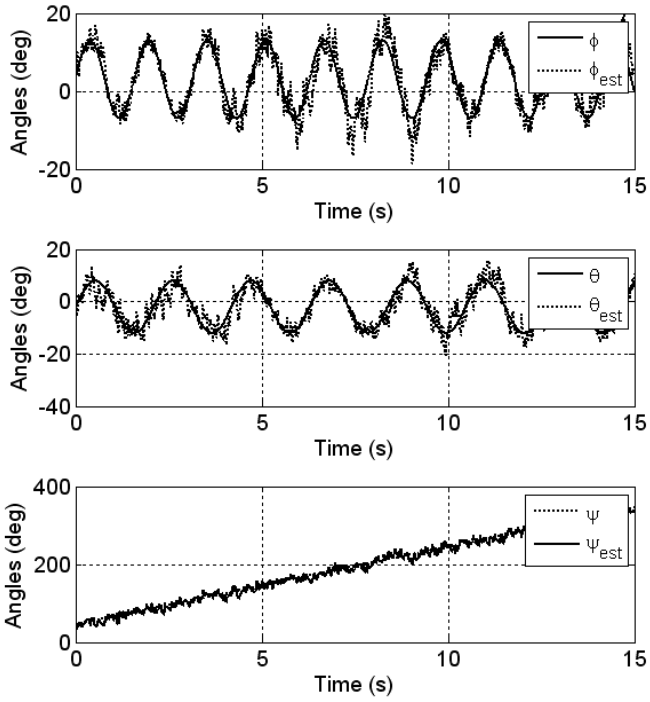


Figure 4. QUEST Euler angle estimates

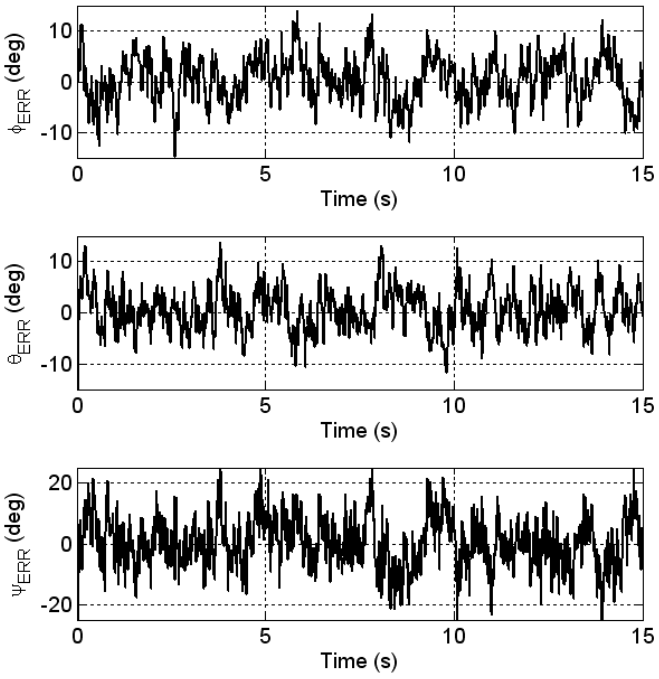


Figure 5. TRIAD Euler angle errors

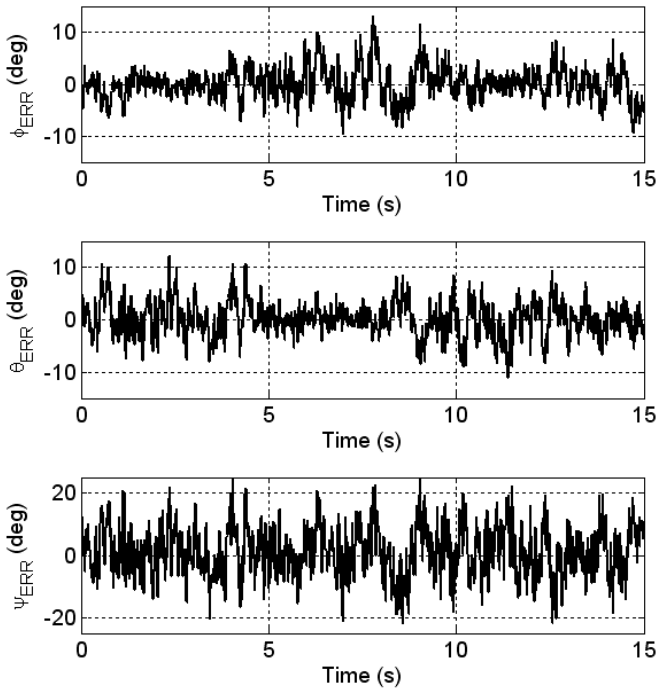


Figure 6. QUEST Euler angle errors

IV. IMPROVED ATTITUDE ALGORITHMS

Small unmanned rotorcraft provide a severe dynamic environment for attitude estimation due to their fast dynamic response and extreme vibration. Sources of vibration include the main rotor, tail rotor, and blade flapping dynamics. In addition, their small size requires light and small sensors, both resulting in low quality measurements. The additional filtering needed to eliminate vibration results in additional delay in orientation estimation. However, due to the fast dynamics, this additional delay leads to possible control instability. An alternative to passive filtering is required to improve orientation estimation.

Consider two sequential orientation estimates using gravity vector measurements corrupted by vibration. From one estimate to the next the estimated orientation may change significantly over a short period of time as seen in both Figs. 5 and 6. If the orientation did indeed change significantly it would require a large angular velocity. In the absence of any other information it is not possible to determine if the change in gravity measurement is from a true rotation, acceleration or noise. The incorporation of a three-axis MEMs gyro can be used to measure $\vec{\omega}_{B/I}$ the angular velocity of the rotorcraft with respect to an inertial reference frame.

$$\vec{\omega}_{B/I} = p\vec{I}_B + q\vec{J}_B + r\vec{K}_B \tag{33}$$

The kinematic differential equations relating the change in orientation with $\vec{\omega}_{B/I}$ for Euler angles, Quaternions, and Direction Cosines are shown in Eqs. (34-36) and can be used to quantify expected changes in orientation over measurements [1].

$$\begin{Bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{Bmatrix} = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi / c_\theta & c_\phi / c_\theta \end{bmatrix} \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \tag{34}$$

$$\begin{Bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{Bmatrix} = \begin{bmatrix} 0 & r & -q & p \\ -r & 0 & p & q \\ q & -p & 0 & r \\ -p & -q & -r & 0 \end{bmatrix} \begin{Bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{Bmatrix} \tag{35}$$

$$\begin{bmatrix} \dot{l}_{11} & \dot{l}_{12} & \dot{l}_{13} \\ \dot{l}_{21} & \dot{l}_{22} & \dot{l}_{23} \\ \dot{l}_{31} & \dot{l}_{32} & \dot{l}_{33} \end{bmatrix} = \begin{bmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{bmatrix} \begin{bmatrix} l_{11} & l_{12} & l_{13} \\ l_{21} & l_{22} & l_{23} \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \tag{36}$$

A. Psuedo-Measurement Vector

QUEST has the ability to easily incorporate additional vector measurements. However, by inspection of Eqs. (22) and (23) the angular velocity $\vec{\omega}_{B/I}$ cannot be directly added as a third vector measurement into the q-method or QUEST because $\vec{\omega}_{B/I}$ in the I frame is unknown. The proposed solution is to create a third pseudo-measurement vector from directly from $\vec{\omega}_{B/I}$ by using the rotational kinematic differential equations.

The first step is to select a pseudo-vector $\vec{v}_{pI}$ in the inertial frame I with the only restriction being it cannot be collinear with either the gravity or magnetic vector. Using the previous estimate of $\mathbf{q}$ along with (19) the previous pseudo-vector in B is,

$$\vec{v}_{pB} = \left[T_{IB}\right]\vec{v}_{pI} \tag{37}$$

At the next measurement Eq. (35) is used to predict $\tilde{\mathbf{q}}$ by propagation of the quaternion kinematics according to Eqs. (38, 39)

$$\tilde{\mathbf{q}} = \mathbf{q} + \Delta t \dot{\mathbf{q}} \tag{38}$$

$$\begin{Bmatrix} \tilde{q}_1 \\ \tilde{q}_2 \\ \tilde{q}_3 \\ \tilde{q}_4 \end{Bmatrix} = \left(I + \frac{\Delta t}{2} \begin{bmatrix} 0 & r & -q & p \\ -r & 0 & p & q \\ q & -p & 0 & r \\ -p & -q & -r & 0 \end{bmatrix} \right) \begin{Bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{Bmatrix} \tag{39}$$

Equation 39 is the new pseudo-measurement vector $\vec{v}_{pB}$. The weights (28b) are selected to adjust the amount of smoothing from the pseudo-vector. If the weight on $\vec{v}_{pB}$ is zero the solution reduces to the original estimator. As the weights on the true measurements approach zero the estimator reduces to numerical integration of the angular velocities, which would quickly diverge due to biases in the gyros. The pseudo-measurement vector and true measurements form complimentary filters. Integration of the angular velocity smoothes the measurements without adding delay, the other measurements eliminate divergence from integration of the gyros.

B. Gyro-Compensated Tilt Sensor and Compass

A drawback of the previous methods is $\vec{m}^I$ the magnetic vector in the inertial frame I is required. Due to magnetic variation and declination the magnetic vector varies as a function of latitude, longitude, altitude, and time. This requires an initial estimation unlike the gravity vector. Unlike the TRIAD and QUEST algorithms, the deterministic method presented in Eqs. (9, 10) and (15) does not require $\vec{m}^I$ but only uses the measurement as a compass. As with the previous proposed algorithm the angular velocity $\vec{\omega}_{B/I}$ can be used to form complimentary filtering. Notice by inspection of (6), (8), and (21) that $\vec{g}^B$ is a direct measurement of the l_{13}, l_{23}, l_{33} direction cosines. Instead of estimating φ and θ from (8) and (9) the gyro-compensated tilt sensor will estimate l_{13}, l_{23}, l_{33}. Using the properties of direction cosines that the dot product of any row or column with itself is 1 the raw gravity measurements can be normalized.

$$\begin{Bmatrix} \tilde{l}_{13} \\ \tilde{l}_{23} \\ \tilde{l}_{33} \end{Bmatrix} = \frac{1}{\sqrt{g_{XB}{}^2 + g_{YB}{}^2 + g_{ZB}{}^2}} \begin{Bmatrix} g_{XB} \\ g_{YB} \\ g_{ZB} \end{Bmatrix} \tag{40}$$

The estimated direction cosines at the next measurement are found by propagating the direction cosine kinematics in Eq. (36).

$$\begin{Bmatrix} \hat{l}_{13} \\ \hat{l}_{23} \\ \hat{l}_{33} \end{Bmatrix} = \left(I - \Delta t \begin{bmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{bmatrix} \right) \begin{Bmatrix} l_{13}(k) \\ l_{23}(k) \\ l_{33}(k) \end{Bmatrix} \tag{41}$$

Finally, the updated direction cosines are calculated by taking a combination of propagated and measured values and the updated Euler angles φ and θ can be found.

$$\begin{Bmatrix} l_{13}[k+1] \\ l_{23}[k+1] \\ l_{33}[k+1] \end{Bmatrix} = \alpha \begin{Bmatrix} \hat{l}_{13} \\ \hat{l}_{23} \\ \hat{l}_{33} \end{Bmatrix} + (1-\alpha) \left(I - \Delta t \begin{bmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{bmatrix} \right) \begin{Bmatrix} l_{13}[k] \\ l_{23}[k] \\ l_{33}[k] \end{Bmatrix} \tag{42}$$

$$\phi[k+1] = \sin^{-1}\left(-l_{13}[k+1]\right) \tag{43}$$

$$\theta[k+1] = \text{atan2}\left(l_{23}[k+1], l_{33}[k+1]\right) \tag{44}$$

Similarly, for ψ Eqs. (13), (43), and (44) can be used to calculate a measured $\tilde{\psi}$ in while an estimated $\hat{\psi}$ is found by propagating the kinematics in (34).

$$\tilde{\psi} = \text{atan2}\left(-m_{YM}, m_{XM}\right) \tag{45}$$

$$\hat{\psi} = \psi[k] + \Delta t \left(\frac{s_{\theta_k}}{c_{\theta_k}} q + \frac{c_{\theta_k}}{c_{\theta_k}} r \right) \tag{46}$$

Finally, an updated ψ is found by taking a combination of propagated and measured values.

$$\psi[k+1] = \alpha\tilde{\psi} + (1-\alpha)\hat{\psi} \tag{47}$$

C. Comparison of Improved Algorithms

To demonstrate the performance of both the improved algorithms, measurement data was generated by adding Gaussian noise with a standard deviation of 0.25 deg/s and a bias of 0.25 deg/s to the angular velocity data shown in Fig. 1. Results from improved algorithms are shown in Figs. 7 and 8 with errors in Figs. 9 and 10. In both cases, the angular velocity measurements have been pre-filtered with a first-order filter having a time constant of 0.075 seconds. For Quest, the weights appearing in Eq. (28b) were selected as 0.01, 0.07, and 1.97 for gravity, magnetic vector, and the pseudo-vector, respectively. For the gyro-compensated algorithm, α was selected as 0.02. It is clear from both Figs. 7 and 8 that the additional information from the gyros has reduced the errors from 10 deg to only a few degrees. Both improved algorithms have similar error bounds. However, QUEST with a pseudo-vector has higher frequency content. Either algorithm could be used, however, the gyro-compensated algorithm is quicker to implement and is the algorithm implement in flight tests discussed later.

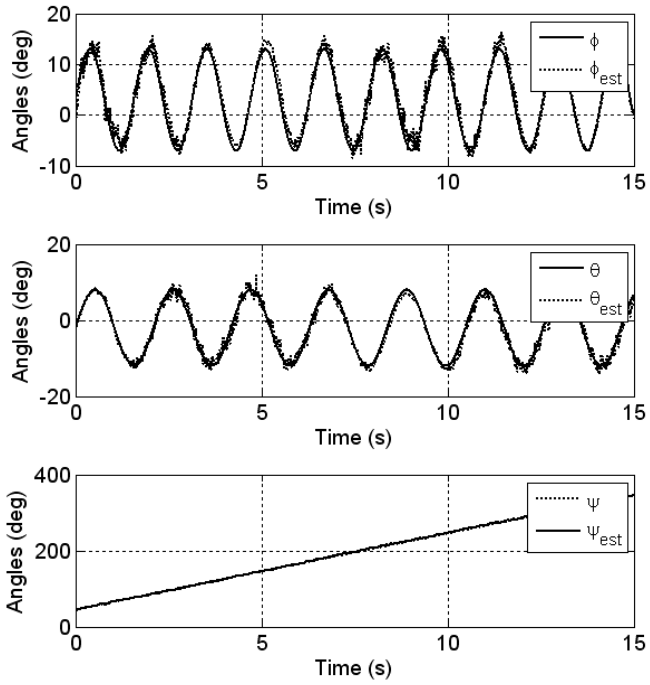


Figure 7. QUEST with Pseudo-Vector Estimates

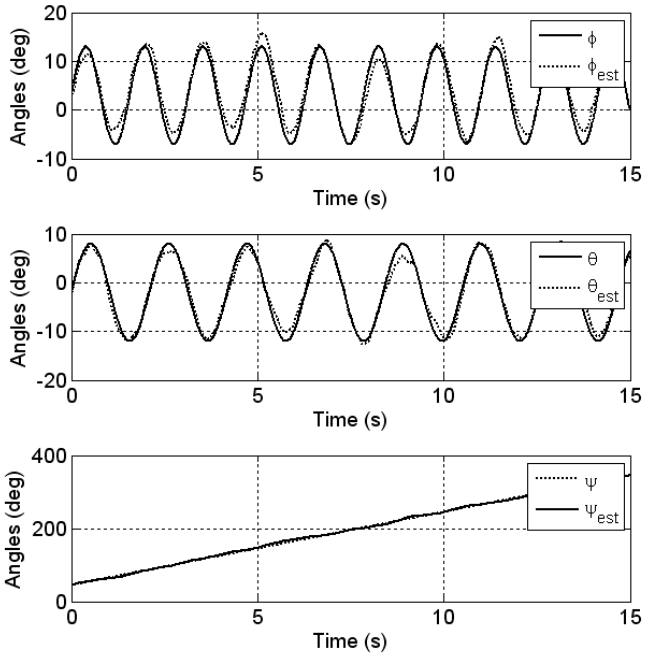


Figure 8. Gyro Compensated Euler angles

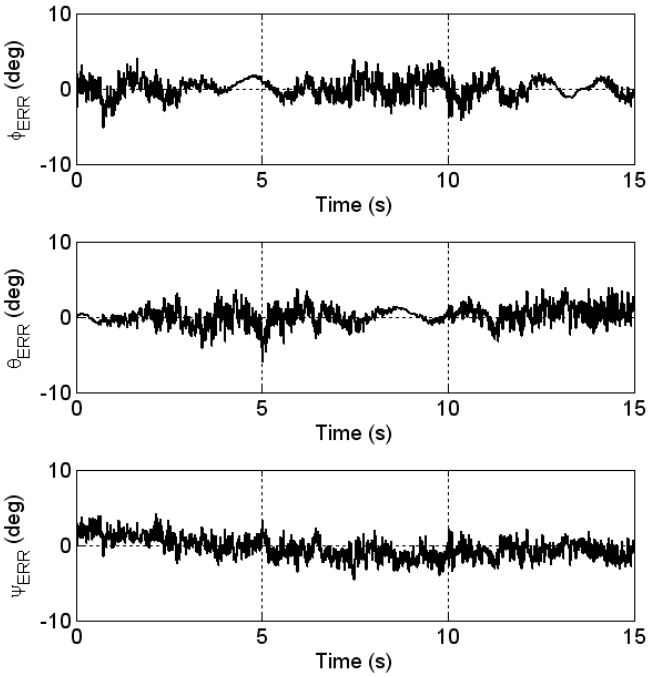


Figure 9. QUEST with Pseudo-Vector Errors

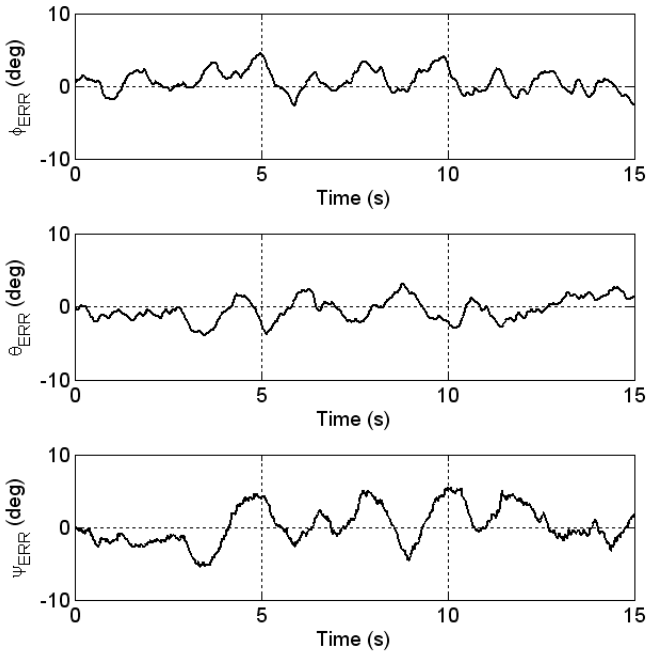


Figure 10. Gyro-Compensated Euler Angles Errors

V. TEST PLATFORM

The T-Rex 600 CF helicopter was chosen as the helicopter platform. This platform is popular among competition remote control pilots for its acrobatic capability. The T-Rex 600 CF is a full size radio controlled helicopter consisting of Carbon Fiber (CF) and CNC aluminum parts. The helicopter measure almost 4 feet in length The T-Rex 600 uses full size 600mm rotor blades and is powered by a ballistic combination of both a large brushless motor and lithium packs(in series). The T-Rex 600 allows for the ability to perform 3D maneuvers. The following sections are a few design features of the T-Rex 600.

Figure 1: T-Rex 600 Test Helicopter

The T-Rex rotor head is made from a combination of plastic moldings and machined aluminum parts. The head uses mixing in the transmitter to drive three servos linked to the three stationary swash plate balls at 120 degrees intervals. This moves the swash plate up and down for collective pitch and tilts it for cyclic control by moving the servos in combination. Because these rotors produce small damping moments in comparison to larger helicopters, the design features stabilizer bars for ease of handling.

The T-Rex 600 is equipped with a mechanical Bell-Hiller stabilizer bar that affectively applies lagged rate feedback to the two cyclic control channels. This system may be regarded as a secondary rotor attached to the shaft below the main rotor by an unrestrained teetering hinge. Aerodynamic paddles are attached to the ends of a rod. Cyclic pitch and roll are inputs transmitted to the stabilizer bar. But unlike the main rotor, the stabilizer bar has no collective.

VI. CONTROL ALGORITHMS

The small rotorcraft control law transforms the task of controlling the rotorcraft over the flight into one of controlling the servo applications for roll, pitch, yaw, and altitude. Information about the desired hover point is used to determine when to correctly adjust the swash plate and the heading. In hover, the desired altitude simply becomes the desired height above the ground. Desired roll and pitch are related to the position of the helicopter. Figure 12 shows a top view of a helicopter with a desired hover location. A vector $\vec{r}$ describing the position error in the helicopter frame is defined in Eq. (48). Equations (49) and (50) relate the desired roll and pitch to the desired location through terms proportional to the position and velocity errors.

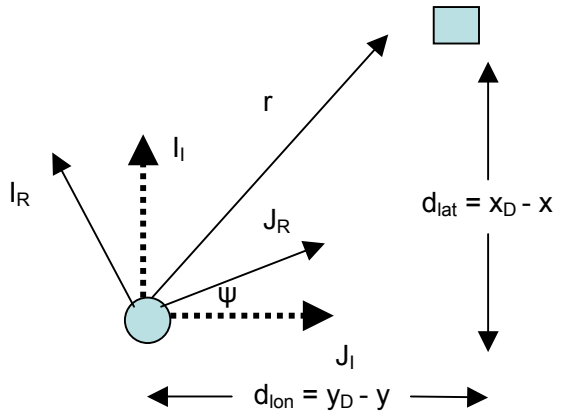


Figure 12: Desired Roll and Pitch Geometry

$$\vec{r} = \begin{Bmatrix} r_x \\ r_y \\ 0 \end{Bmatrix} = \begin{bmatrix} c_\psi & s_\psi & 0 \\ -s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} d_{lat} \\ d_{lon} \\ 0 \end{Bmatrix} \tag{48}$$

$$\phi_{DES} = -K_p r_y + K_D \dot{r}_y + K_{\phi SS} \tag{49}$$

$$\theta_{DES} = K_p r_x - K_D \dot{r}_x \tag{50}$$

The small rotorcraft control algorithm includes three components: Position/Velocity Filter, Swash Plate Control, and Heading Lock. Figure 13 illustrates the combined flight system with descriptions and block diagrams of the individual modules provided below. Figure 14 demonstrates the T-Rex 600 helicopter's servo location with respect to the swash plate and tail rotor. This representation clarifies the output channels with their appropriate servo.

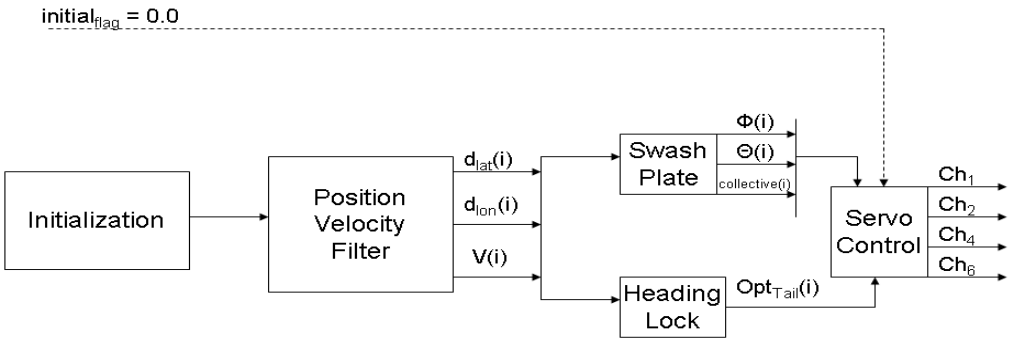


Figure 13. Autonomous Small Rotorcraft Flight Strategy

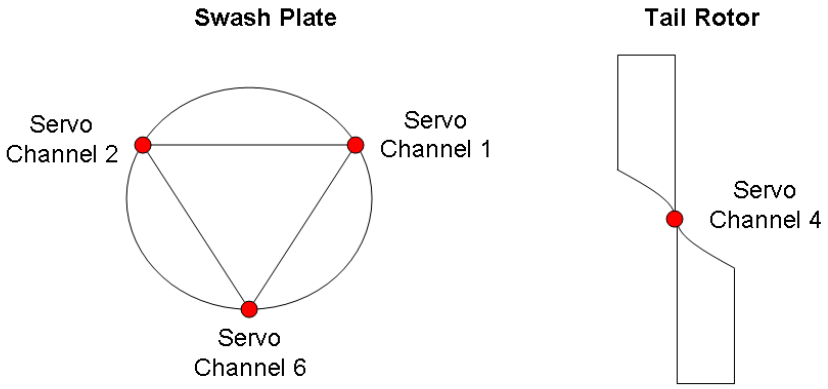


Figure 14. Servo Representation and Location on Swash Plate and Tail Rotor

A. Position/Velocity Filter

The position/velocity control filter illustrated in Figure 15 transforms the current GPS latitude (lat_{point}) and longitude (lon_{point}) points of the rotorcraft into feet and subtracts these points from the desired hover position to calculate the current lateral and longitudinal position error vectors. During the saturation process, a scale factor is used with the distance vectors to preserve the magnitude in the occurrence of large position errors. The derivatives of the distance vectors are used to calculate the vehicle's current velocity. The velocity vector is saturated with respect to V_{MAX}. Finally, the error vectors are integrated with respect to the previous integral error and featured with a forgetting factor. The magnitude of this integration is determined to provide the current position integral error. The outputs of the position/velocity filter are the current distance vector, the current vehicle velocity, and the current

integral error. The control system is updated four times a second. (See Appendix for description of position/velocity filter block elements.)

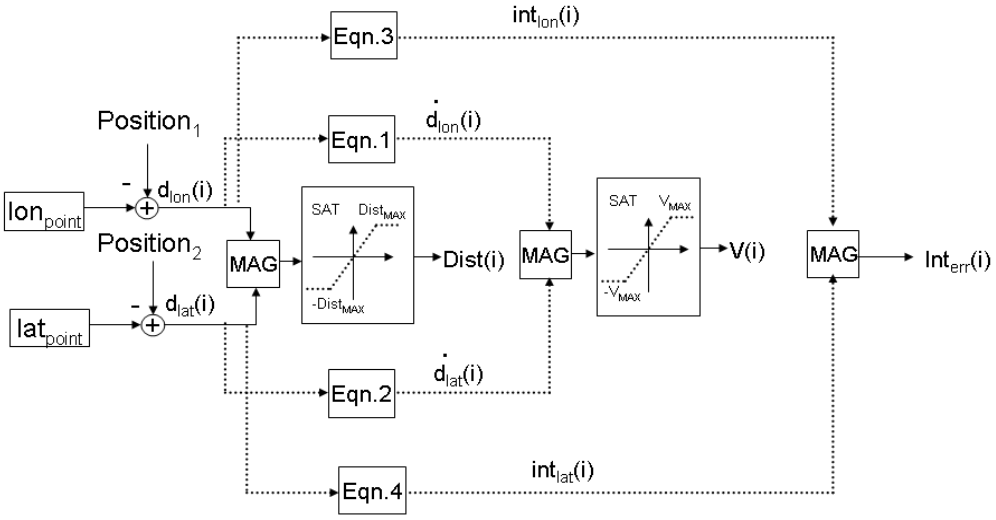


Figure 15. Position/Velocity Control Filter Block Diagram

B. Swash Plate Control

The output of position/velocity filter is transferred to the swash plate control system along with the current vehicle heading and the desired altitude. The distance, velocity, and integral errors from the position/velocity filter are transformed into the vehicle's body frame by the heading angle, yaw. The desired roll and pitch for the flight path has been previously described in Figure 12. The transformed vectors are factored with their appropriate gain: proportional, integral, or derivative value. The values in the x direction are combined together to provide the true roll value, φ_{True}. This is combined with the sensor roll and roll rate of the helicopter plus a bias and their appropriate gains. The values in the y direction are combined together to provide the true roll value, θ_{True}. This is combined with the sensor pitch and pitch rate of the helicopter plus a bias and their appropriate gains. The outputs then become the current roll and pitch control of the helicopter which is converted to servo controls: Channels 1 and 2.

For the altitude control, the current altitude is inputted using a barometric altimeter and subtracted with the desired altitude for the flight. The current altitude error is saturated with respect to Alt_{MAX}. The current altitude error is combined with current velocity and current altitude integral error in a simple proportion-integral-derivative

control fashion to calculate the current collective pitch of the vehicle in flight. The collective output is then converted to servo control: Channel 6. The combined roll, pitch, and altitude control systems seen in Figure 16 move the swash plate of the vehicle to reach the desired hover height and location. The control system is updated twelve times per second. See Appendix for block element descriptions.

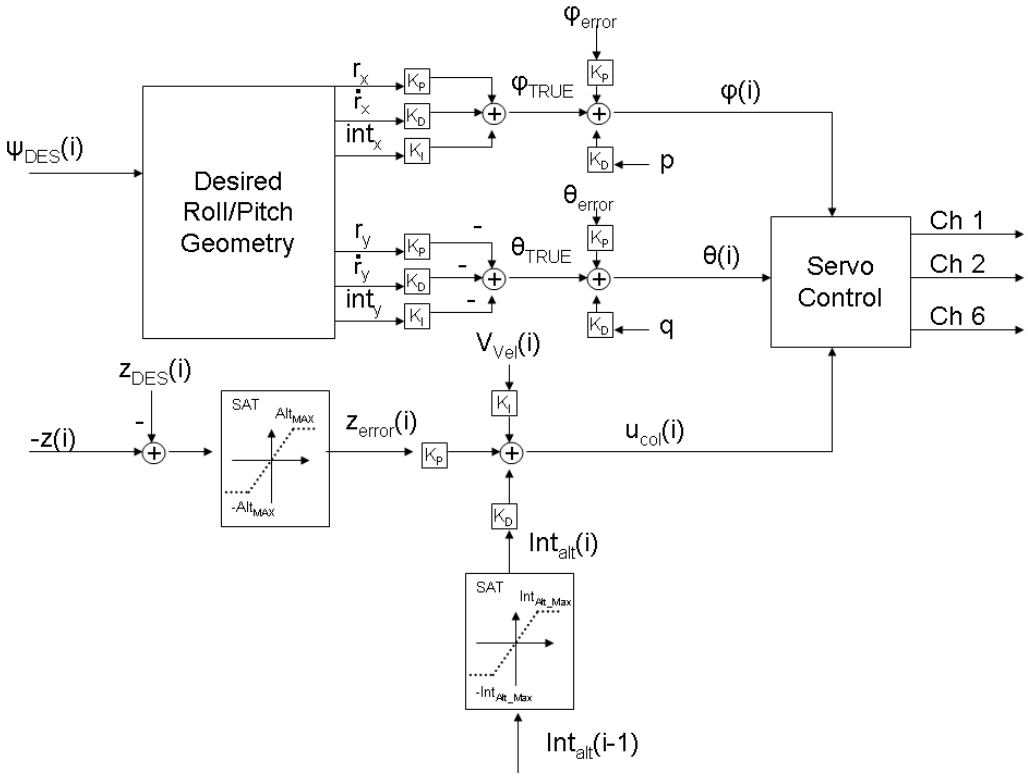


Figure 16: Swash Plate Control Block Diagram

C. Heading Lock Control

For the heading lock control system illustrated in Figure 17, the input is the current yaw angle which is subtracted from the desired heading. First, the yaw error is saturated with pi to determine the correct direction of the heading angle. Then, the yaw error is saturated with respect to Yaw_{error_MAX}. The current yaw error is combined with the current integral error and the sensor yaw rate in a proportional-integral-derivative control fashion to determine the optimum tail position. The optimum tail position is saturated with respect to Opt_{Tail_MAX}. The optimum tail position is converted to servo control: Channel 4. The heading lock control system is updated twelve times per second. See Appendix for block element descriptions.

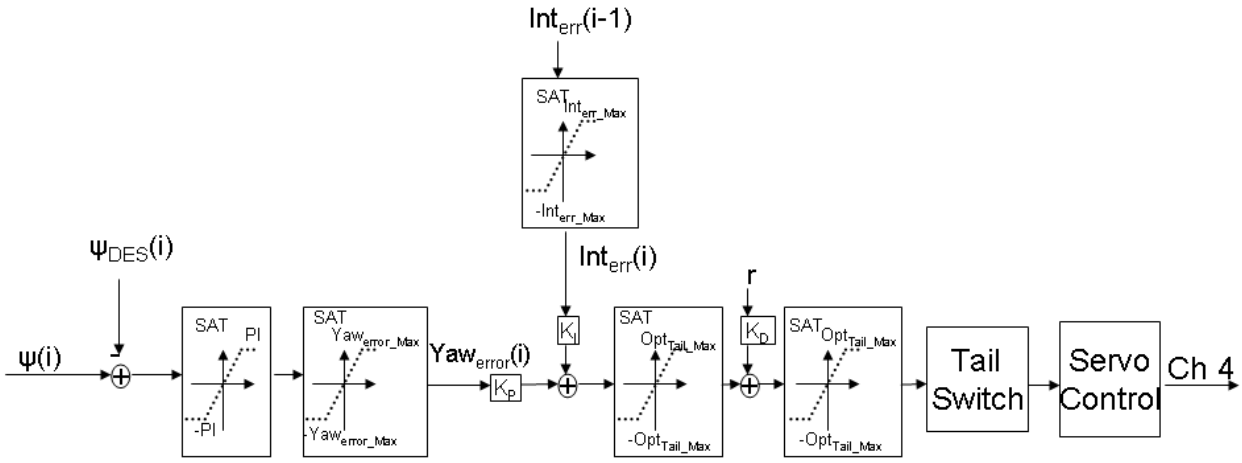


Figure 17: Heading Lock Control Block Diagram

D. Autopilot

The flight control system hardware include: Multiple microprocessors, GPS, three axis accelerometers, three axis magnetometer, three axis gyroscope, electronic variometer, airspeed sensor and barometric altimeter. The autopilot is illustrated in Figure 18. Figure 19 also shows the autopilot mounted and attached to the T-Rex 600 helicopter. The gyro-compensated tilt sensor and compass was selected as the orientation algorithm due to its comparable performance with the optimal QUEST solution and its simplicity.

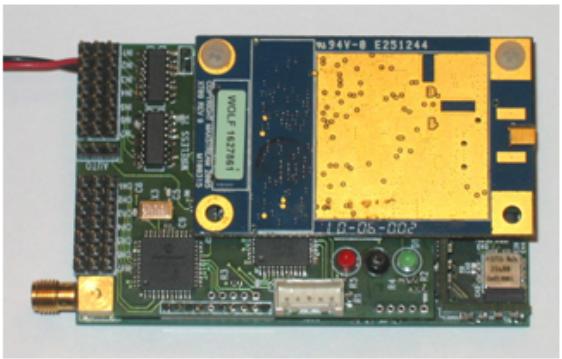


Figure 18: Miniature Autonomous Rotorcraft Autopilot

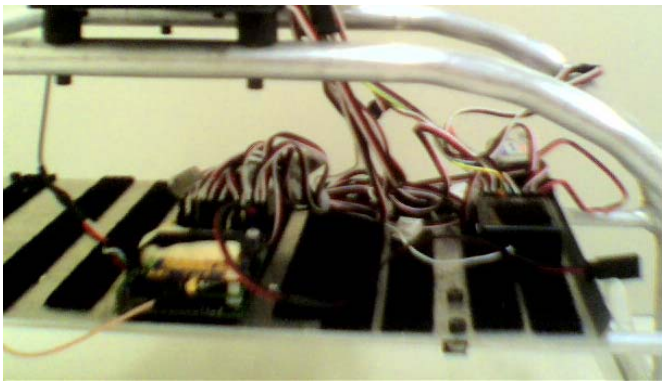


Figure 19: Current Autopilot connected to T-Rex 600 Servos and Receiver

E. Graphical User Interface

The ability to communicate wirelessly with the autopilot is a critical component in the autonomous flight development. This achievement is made possible by the graphic user interface (GUI) seen in Figure 20. The GUI allows the user to view real time sensor data, change desired wave points in current flight, store proportional-integral-derivative gains, calibrate onboard sensors and servo parameters, and view selected variables of the control algorithm.

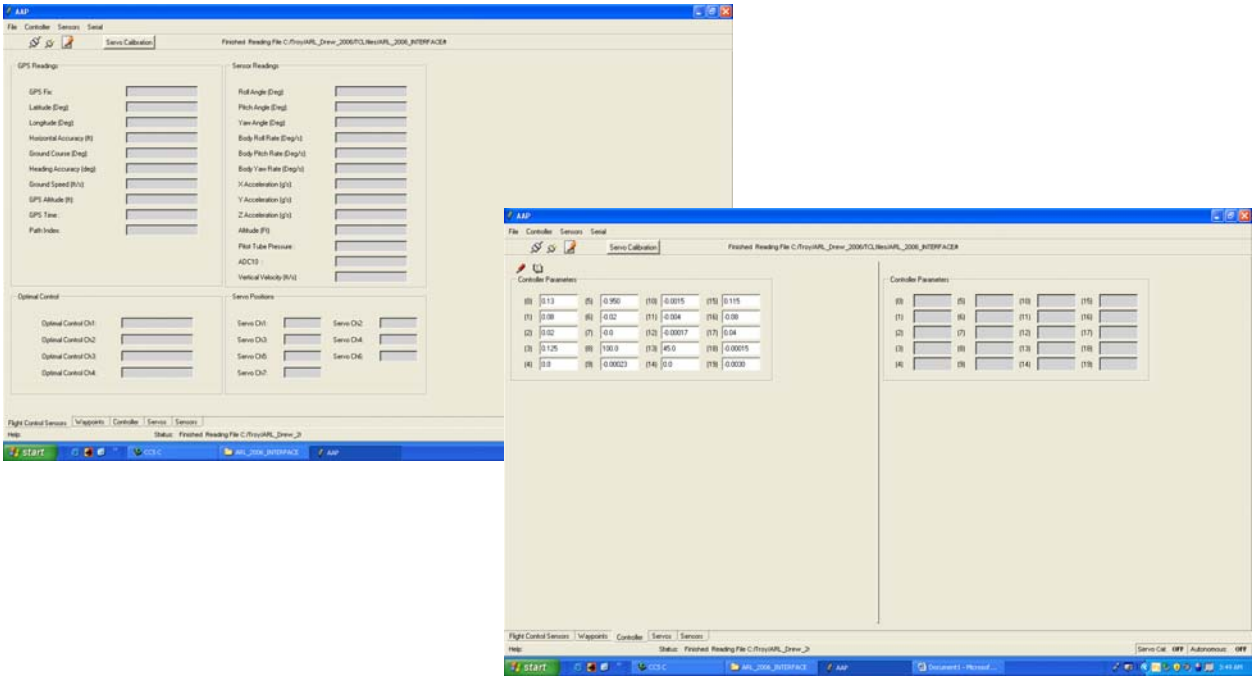


Figure 20: Graphical User Interface

F. Flight Test Results

For the flight test, the T-Rex 600 implemented with the autopilot was taken to a field at Dublin Park, in Madison, Alabama. The helicopter was placed stationary on the field until a reasonable GPS accuracy was reached. While the helicopter was stationary, all sensors were checked and calibrated appropriately. The helicopter was then placed in the center of the soccer field and that specific GPS point was entered as the desired wave point into the graphical interface. With the complete control algorithm programmed and all servos attached to the autopilot, the helicopter was taken to an altitude of 5ft manually and switched to autonomous mode. A camera was mounted to the bottom of the landing gear to track the desired point in the center of the soccer field, which was marked by a 2ft x 2ft x 2ft black case. Gains for desired roll and pitch K_P, K_D, K_I were selected as 0.00017, 0.00033, and 0.0006 deg, respectively. The T-Rex 600 was initialized at x = 8ft west and y = 60ft north, z = 0m, φ = 0.6 deg, θ = 2deg and ψ = -142 deg. The desired position is x = y =0ft, z = 30ft and the desired orientation is $\theta = \varphi = 0$ deg and ψ = -142 deg. Figure 21 shows the real time flight data for altitude. Sensor noise that is amplified by the PID gains causes the helicopter to oscillate about the desired height of 30ft by ±10ft. Figure 22 displays the helicopter speed which remains about 3ft/s for the duration of the flight. In Fig. 23, the orientation of the rotorcraft is shown to track the desired angles appropriately. Figure 24 provides the real time flight trajectory for the entire flight**.** The red circle provided in the graph displays 20ft diameter in which the helicopter is trying to hover. Finally, Figure 25 is a real time shot of the helicopter in flight tracking the desired location marked by the 2ft x 2ft x 2ft black case.

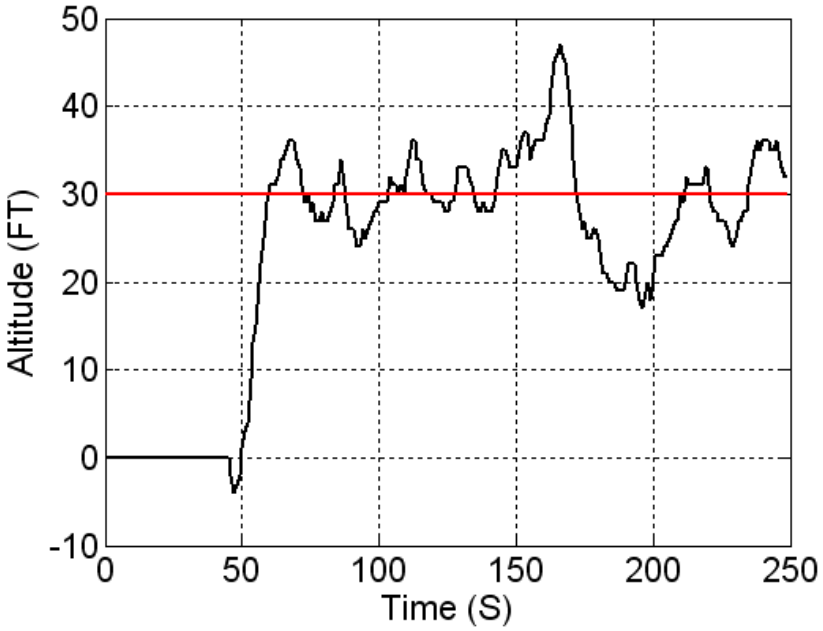


Figure 21. Real Time Altitude Data Tracking Desired 30ft

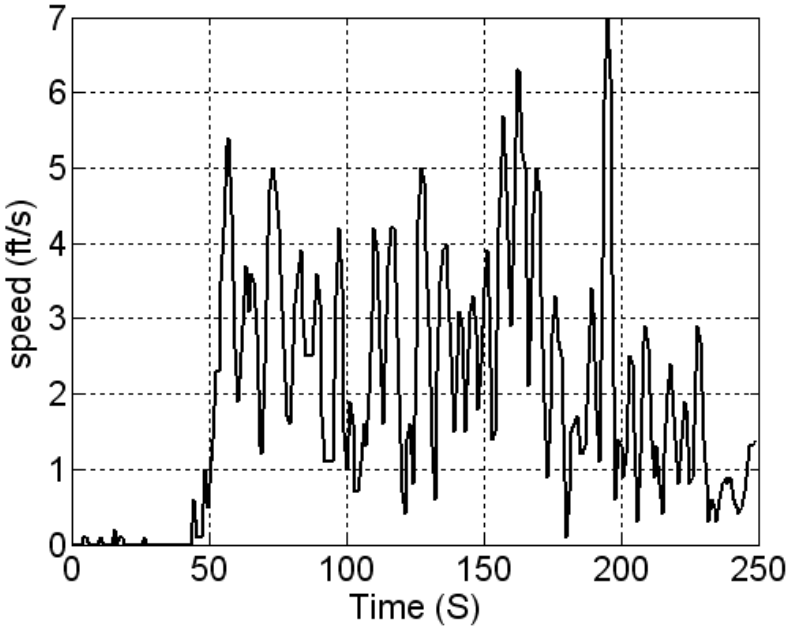


Figure 22. Helicopter Velocity for Real Time Test Case

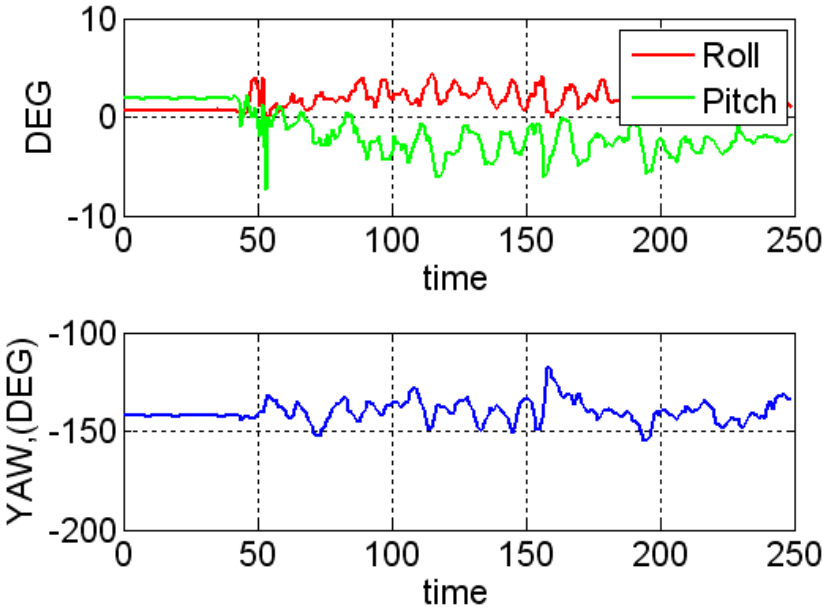


Figure 23. Real Time Flight Orientation

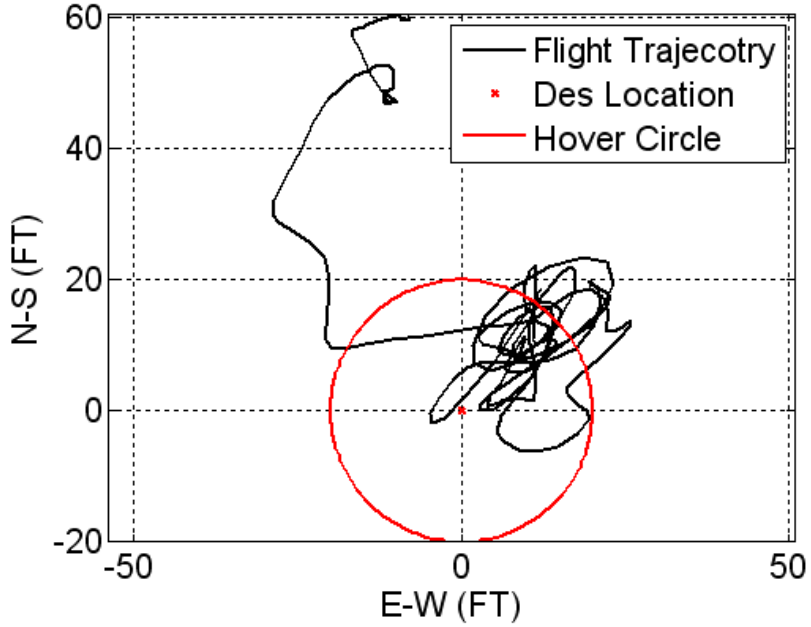


Figure 24. Real Time Flight Trajectory within 20ft Circle

Figure 25. Real Time Flight Video Containing Bottom View of Tracking Desired Hover Location

REFERENCES

[1] Kuipers, J. B., *Quaternions and Rotation Sequences*, Princeton University Press, NJ, 1999, pp. 155-171.

[2] Black, H. D., "A Passive System for Determining the Attitude of a Satellite," *AIAA Journal*, Vol. 2, July 1964, pp. 1350-1351.

[3] Lerner, G. M., "Three-Axis Attitude Determination," *Spacecraft Attitude Determination and Control*, edited by J. R. Wertz, D. Reidel Publishing Co., Netherlands, 1978, pp. 420-428.

[4] Shuster, M. D., and OH, S. D., "Three-Axis Attitude Determination from Vector Observations," *Journal of Guidance and Control*, Vol 4, No 1, 1981, pp 70-77.

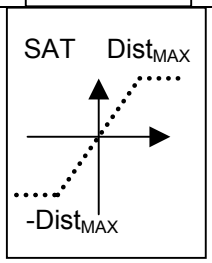
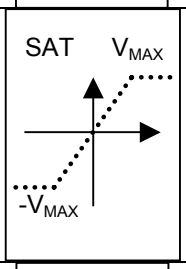
[5] Wahba, G., "Problem 65-1, (Solution)" SIAM Review, Vol. 8, 1966, pp. 384-386.

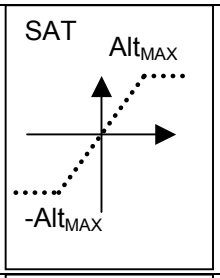
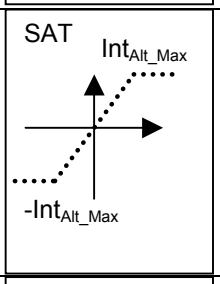
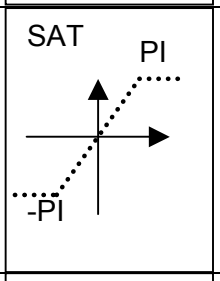
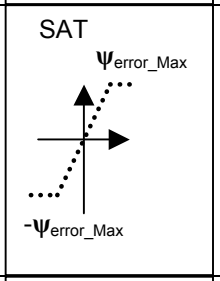
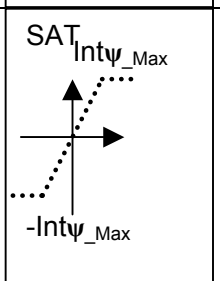
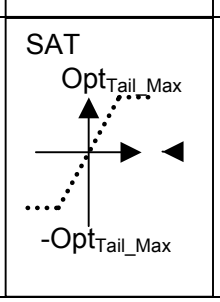
[6] Wahba, G., et al, "Problem 65-1, A Least Squares Estimate of Satellite Attitude" SIAM Review, Vol. 7, 1965, p. 409.

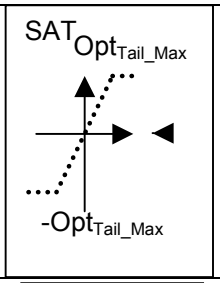
[7] Keat, J., "Analysis of Least Squares Attitude Determination Routine, DOAOP," Computer Science Corp., CSC/TM-77/6034, 1977.

[8] Shuster, M. D., "Approximate Algorithms for Fast Optimal Attitude Computation," Proceedings AIAA Guidance and Control Conference, Palo Alto, CA, Aug. 7-9, 1978.

Appendix

Block Element	Description
MAG	Dist(i) = sqrt($d_{lon}(i)*d_{lon}(i) + d_{lat}(i)*d_{lat}(i)$)
	$S_F = 125/Dist(i)$ $d_{lon}(i) = d_{lon}(i) * S_F$, **if Dist(i) > 125 ft** $d_{lat}(i) = d_{lat}(i) * S_F$ Dist(i) = 125 $d_{lon}(i) = 0.0$, **if Dist(i) < 1.0 ft** $d_{lat}(i) = 0.0$ $S_F = 1.0 – 1.0/Dist(i)$ $d_{lon}(i) = d_{lon}(i) * S_F$, **if Dist(i) <125 ft** $d_{lat}(i) = d_{lat}(i) * S_F$ $S_F = 60/Dist(i)$, **if Dist(i) > 60.0 ft**
Eqn.1	$\dot{d}_{lon}(i) = 2.0*(d_{lon}(i) - d_{lon}(i-1) + 0.5* \dot{d}_{lon}(i)$;
Eqn.2	$\dot{d}_{lat}(i) = 2.0*(d_{lat}(i) - d_{lat}(i-1) + 0.5* \dot{d}_{lat}(i)$;
MAG	V(i) = sqrt($\dot{d}_{lon}(i) * \dot{d}_{lon}(i) + \dot{d}_{lat}(i) * \dot{d}_{lat}(i)$);
	$\dot{d}_{lon}(i) = \dot{d}_{lon}(i)$ *27.0/V(i) $\dot{d}_{lat}(i) = \dot{d}_{lat}(i)$ *27.0/V(i) , **if V(i) > 27.0 ft/s**
Eqn.3	$int_{lon}(i) = int_{lon}(i) *F_F + dt* int_{lon}(i) *S_F$;
Eqn.4	$int_{lat}(i) = int_{lat}(i) *F_F + dt* int_{lat}(i) *S_F$;
MAG	$int_{err}(i) = sqrt(int_{lon}(i) * int_{lon}(i) + int_{lat}(i) * int_{lat}(i))$;
Desired Roll/Pitch Geometry	**sinyaw = sin(ψ(i));** **cosyaw = cos(ψ (i));** **r_x = cosyaw*** $d_{lat}(i)$ + **sinyaw*** $d_{lon}(i)$**;** **r_y = -sinyaw*** $d_{lat}(i)$ + **cosyaw*** $d_{lon}(i)$**;** $\dot{r}_x$ = **cosyaw*** $\dot{d}_{lat}(i)$ + **sinyaw*** $\dot{d}_{lon}(i)$ **;** $\dot{r}_y$ = **-sinyaw*** $\dot{d}_{lat}(i)$ + **cosyaw*** $\dot{d}_{lon}(i)$ **;** **int_x = cosyaw*** $int_{lat}(i)$ + **sinyaw*** $int_{lon}(i)$**;** **int_y = -sinyaw*** $int_{lat}(i)$ + **cosyaw*** $int_{lon}(i)$**;**

			$Alt_{error}(i) = 30.0$, **if $Alt_{error}(i) > 30.0$** $Alt_{error}(i) = 30.0$, **if $Alt_{error}(i) = 30.0$** $Alt_{error}(i) = -30.0$, **if $Alt_{error}(i) < -30.0$**
			$Int_{Alt}(i) = 75.0$, **if $Int_{Alt}(i) > 75.0$** $Int_{Alt}(i) = 75.0$, **if $Int_{Alt}(i) = 75.0$** $Int_{Alt}(i) = -75.0$, **if $Int_{Alt}(i) < -75.0$**
			$\psi_{error}(i) = \psi_{error}(i) - 6.28319$, **if $\psi_{error}(i) > PI$** $\psi_{error}(i) = \psi_{error}(i)$, **if $\psi_{error}(i) = PI$** $\psi_{error}(i) = \psi_{error}(i) + 6.28319$, **if $\psi_{error}(i) < -PI$**
			$\psi_{error}(i) = 1.57$, **if $\psi_{error}(i) > 1.57$** $\psi_{error}(i) = 1.57$, **if $\psi_{error}(i) = 1.57$** $\psi_{error}(i) = -1.57$, **if $\psi_{error}(i) < -1.57$**
			$Int_{err}(i) = 3.0$, **if $Int_{err}(i) > 3.0$** $Int_{err}(i) = 3.0$, **if $Int_{err}(i) = 3.0$** $Int_{err}(i) = -3.0$, **if $Int_{err}(i) < -3.0$**
			$Opt_{Tail}(i) = 0.125$, **if $Int_{err}(i) > 0.125$** $Opt_{Tail}(i) = 0.125$, **if $Int_{err}(i) = 0.125$** $Opt_{Tail}(i) = -0.125$, **if $Int_{err}(i) < -0.125$**

		$Opt_{Tail}(i) = 0.125$, **if $Int_{err}(i) > 0.125$** $Opt_{Tail}(i) = 0.125$, **if $Int_{err}(i) = 0.125$** $Opt_{Tail}(i) = -0.125$, **if $Int_{err}(i) < -0.125$**

	Tail Switch	$tail_{switch} = 1.75 * 0.125;$
		if($abs(Opt_{Tail}(i) - Opt_{Tail}(i-1)) > tail_{switch}$) {
		$Opt_{Tail}(i) = Opt_{Tail}(i-1);$
		}