FLYING QUALITIES BUILT-IN-TEST FOR UNMANNED AERIAL SYSTEMS

A thesis presented to the faculty of
California Polytechnic State University, San Luis Obispo

In partial fulfilment of
the requirements for the degree
Master of Science in Aerospace Engineering

by
Alton P. Chiu
June 2012

COMMITTEE MEMBERSHIP

TITLE: Flying Qualities Built-In-Test for Unmanned Aerial Systems

AUTHOR: Alton P. Chiu

DATE SUBMITTED: June 2012

COMMITTEE CHAIR: Eric A. Mehiel, Ph.D., Associate Professor, Aerospace Department

COMMITTEE MEMBER: Daniel J. Biezad, Ph.D., Professor, Aerospace Department

COMMITTEE MEMBER: Jangho Yoon, Ph.D., Lecturer, Aerospace Department

COMMITTEE MEMBER: Olga Dekhtyar, Lecturer, Statistics Department

ABSTRACT

Flying Qualities Built-In-Test for Unmanned Aerial Systems

Alton P. Chiu

This paper presents a flying qualities built-in-test for UAS application. A doublet input waveform excites the AV and both α and q are used by EUDKF to estimate the A and B matrices which are short period approximations of the system. ζ, ω, GM, PM, observability, and controllability are calculated to determine flying qualities with the results displayed to the AVO in a color-coded, easy to interpret display.

While SID algorithms have been flying in vehicles with adaptive control schemes, vehicles with other schemes (such as classical feedback) lack this built-in self assessment tool. In addition, adaptive control SID results are not analyzed and displayed but instead used internally. This work intends to extend this self-assessment option to all UASs regardless of control scheme as a "plug-and-play" add-on by building a reliable and robust tool that requires little tuning.

TABLE OF CONTENTS

LIST OF FIGURES

I. EXECUTIVE SUMMARY

B ATTLE damage or malfunctions can unpredictably compromise the airworthiness of an air vehicle. Furthermore, failure modes may be difficult to accurately model, analyze, and provide for. Given such challenges, the Flying Qualities Built-In-Test presented in this paper is invaluable in providing real-time qualitative metrics so that the air vehicle operator can decide whether to terminate the air vehicle or attempt a recovery.

Unmanned aerial systems face additional challenges compared to manned platforms in an off-nominal state. Pilots of manned platforms enjoy high-rate quantitative cues (instrument readouts) in addition to having qualitative (seat-of-the-pants) feedback. Unmanned aerial system operators suffer from a limited quantitative dataset updated at low rates (due to comm.-link bandwidth limitations and time-delay). The lack of qualitative feedback also contributes to the degradation of situational awareness, leaving the operator with serious handicaps.

The Flying Qualities Built-In-Test addresses these shortcomings by reliably providing the operator with accurate real-time airworthiness assessment "at the click of a button". Using system identification algorithms to estimate the state ($\hat{A}$) and input ($\hat{B}$) matrix, robustness (gain and phase margin) and stability (damping ratio and frequency) are evaluated. Controllability and observability tests establish test validity. The results are concisely displayed to the operator with unequivocal recommendations.

Air vehicles with control schemes without an explicit system identification component lack the ability to diagnose flying qualities online. This work extends this self-assessment to all large unmanned systems (such as RQ-4) regardless of control schemes as a "plug-and-play" add-on.

For a copy of the MATLAB files accompanying this work, please contact the aerospace department at California Polytechnic State University, San Luis Obispo.

II. INTRODUCTION

The use of Unmanned Aerial Systems (UAS) has risen dramatically in recent years. While their robotic nature has them uniquely suited to missions with long-endurance requirements, the reduced risk to human operators has also thrust the UASs into an armed role. With the rise of responsibilities and strategic importance come mounting intellectual and monetary investments that must be conserved. The corresponding shift from their earlier disposable nature gives rise to the efforts to increase survivability. This section briefly discusses the history of UAS, previous efforts to increase survivability, and the solution that is the topic of this thesis.

A. HISTORY

The history of UAS is intertwined with manned aviation from the very start. UASs were used as technology demonstrators; their flights were used to analyze theory and ideas before implementation on manned versions. The Montgolfier Brothers flew the first living beings in a balloon to assess the effects of flight in 1783. A sheep was used as it was believed to reasonably approximate human physiology; a duck was expected to be unharmed and included as a control for the effects of the craft; a rooster was also included as a control as it was a bird that does not fly at high altitudes. Samuel Langley's unmanned Aerodrome No.5, powered by a miniature steam engine, flew under its own power seven years before the Wright Flyer's historic flight at Kitty Hawk. However, these early machines lacked the stability and guidance mechanisms found on modern UASs.

By the end of the Great War, legends like the Sopwith Camel and Fokker DR.I had indisputably established the manned aircraft as a vital implement of war. Less known was the development of the UAS as cruise missiles. Only a decade after Kitty Hawk, Elmer Sperry had

Figure 1. *Kettering Bug.*

developed a gyro-based autopilot which provided the basis for the Hewitt-Sperry Automatic Airplane. Possessing a barometer to regulate altitude, gyroscope for heading tracking, gyroscopic stabilizers for stability, servo-motors for control surface actuation, and engine RPM based distance measuring equipment, the Automatic Airplane possessed most of the same functional blocks as a modern UAS. A similarly equipped Kettering Bug (Figure 1) was designed for deployment at the Western Front. After flying at a predetermined height, the wings were detached at a predetermined distance and the explosive payload detonated upon impact. However, teething troubles and the armistice prevented its operational use. These early drones were characterized by their disposable nature and the pre-programmed-only method of guidance.

The interwar years saw advancement of technology in autopilot, television cameras, and radio remote-control. The autopilot system was refined to the point where the Norden bombsight could perform bomb-runs by issuing guidance commands to be executed by the autopilot. The pilot did not fly the bomber during the run, the bombardier only placed the crosshair on the target and fed information such as wind into the Norden bombsight; it performs the calculation of flight path and release point, and the required adjustments were issued to the autopilot. Television cameras and radio remote-control equipment allowed aircraft to be piloted remotely with a higher degree of feedback to the human operator. The Second World War sent these technologies into action in the form of self-guided (V-1 Flying Bomb) and remotely-piloted (Operation Aphrodite) UAS.

After the failure of the Luftwaffe to subdue the British Isles, Germany sought alternative methods of bringing war to the British population. One such solution was the Fieseler Fi 103, also known as the V-1 Flying Bomb (illustrated in Figure 2). Like the Kettering Bug, the V-1 possessed mechanisms that regulated course, altitude, speed and distance to detonation. While the mechanisms were more refined, the underlying concept remained the same. The V-1 executed a pre-programmed sequence-of-events, which could not be modified mid-flight by either the vehicle itself or human operators.

Figure 2. *Fi 103.*

On the other hand, the advent of television and improvements in radio-control allowed a UAS to attain closed-loop guidance. Operation Aphrodite was an Allied attempt at destroying the German V-weapons site with remotely-piloted B-17/24s laden with explosives. The takeoff and climb portions were performed by an on-board pilot. Once at cruise, the pilot bailed out and control was handed over to the remote pilot onboard a chase aircraft. A television camera aimed outside the windows gave the remote pilot situational awareness while another camera aimed at the instrument panel provided telemetry (hence the etymology of the word). The explosive aircraft was flown into the target by the remote pilot while he loitered in the chase aircraft outside the range of the defenses. While not very successful (one operation cost the life of Joseph Kennedy, older brother of John F. Kennedy), this early attempt at RPV provided the human-machine interaction that was vital to modern UAS. It is noteworthy that these drones of WWII were disposable by nature and did not warrant the consideration of survivability.

The Cold War (and the various localized hot wars) saw the deployment of UAS as reconnaissance platforms. The D-21 (pictured in Figure 3) is a Mach-3 capable drone designed for a one-way trip utilizing much of the same technology as the SR-71. After flying a pre-programmed route, the reconnaissance payload (camera and film) is ejected for recover, the drone then self-destructs[12].

Figure 3. *D-21 atop an SR-71.*

Up to this point, all the UASs discussed are disposable in nature; the Ryan Model 147 (Figure 4) breaks new ground in being recoverable. Instead of ejecting the reconnaissance payload like the D-21, the entire vehicle parachutes to the ground at the recovery site[18]. The Ryan 147 is also notable in that it began replacing manned reconnaissance flights. After the loss of a U-2 during the Cuban Missile Crisis, Ryan 147s were almost used for reconnaissance until a decision to resume U-2 flights was made just prior to the launching of the drones. In April 1969, an EC-121 was shot down during a SIGINT mission, killing all 31 onboard. The incident led to the 147T which flew under both internal guidance and controller instruction aboard the DC-130 launch aircraft through a communications datalink. This last element brought the UAS evolution from disposable "fire-and-forget" drones to the reusable, human-in-the-loop vehicles as we have
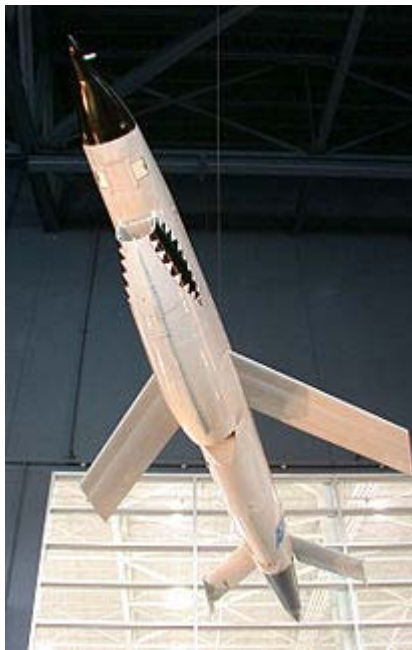
Figure 4. *Ryan 147.*

come to associate with modern UAS such as the MQ-1 Predator.

Figure 5. *Armed Ryan 147.*

The Ryan 147 represented another first by adding weapons delivery to the repertoire of the UAS. Throughout the Vietnam War, SA-2 surface to air missiles presented a major threat to the US warplanes. Suppression of Enemy Air Defenses missions was undertaken by modified F-105 and F-4s; losses among such units were especially high. The disposable nature of UASs lent itself well to undertaking these highly dangerous missions. In 1972, a Rayn 147 (depicted in this role in Figure 5) successfully delivered an AGM-65 Maverick electro-optical guided missile into a radar van mockup during test[14]. Although the system was never deployed operationally, the potential was recognized for armed UAS to soften up targets in the initial wave before the manned platforms strike[40].

After the Vietnam drawdown, the UAS development lapsed until the RQ-2 Pioneer brought UAS to the limelight during the 1991 Gulf War. A DOD report on Desert Shield and Storm credited the UAS with "direct and indirect gunfire support, day and night surveillance, target acquisition, route and area reconnaissance and BDA."[18] Such services led to Dick Cheney commenting that the RQ-2 "appears to have validated the operational employment of UAVs in combat."[7] The shortcomings discovered during the Gulf War and the promise shown by UASs led to the development of the RQ-1/MQ-1/MQ-9 Predator/Reaper in the medium altitude long endurance category and the RQ-4 Global Hawk in the high altitude long endurance area.

Figure 6. *MQ-1 Predator.*

The RQ-1 Predator (pictured in Figure 6) provided intelligence in the forward observation role. In marked contrast to previous UASs, RQ-1 was capable of executing pre-programmed operations as well as being flown "stick-and-rudder" style by a remote pilot. In addition, it had been modified to carry munitions. In 2002, the designation was changed to MQ-1 ("M" for multi-role) reflecting its growing use as an armed strike platform. The high demand of this vehicle is illustrated in Figure 7. This partly reflects the persistence requirements of the War on Terror, and partly the desire to reduce casualties. The long endurance capability of UASs not only lies in the airframe, but also with the human operators. Because operators can hand off control to one another in order to take a rest, human endurance no longer contributes to the equation. Because of the satellite link capability, the deployment cost is also lower in that human operators and their
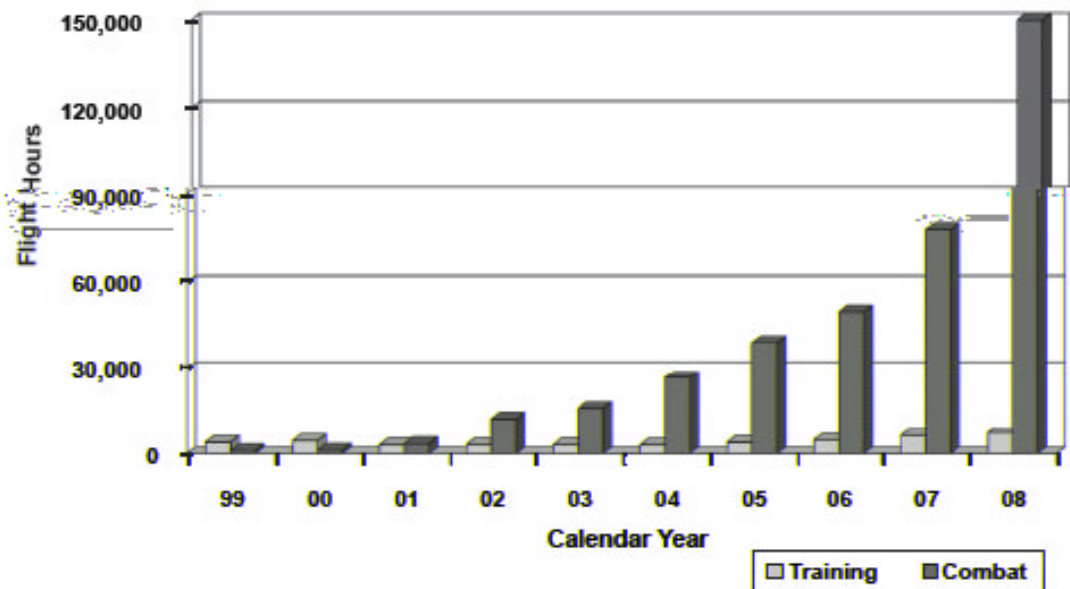
Figure 7. *MQ-1 usage[6].*

attendant supply train need not be physically at the operational area.

The RQ-4 (depicted in Figure 8) is similar in role and operational design as the U-2 by providing a theatre commander with broad overview and

Figure 8. *RQ-4 Global Hawk.*

surveillance. While sharing much of the architectural design and advantages as the RQ-1, the guidance of the RQ-4 Global Hawk differs by dispensing with the "stick-and-rudder" capability. Instead, the human operator only issues outer loop commands to the autopilot, specifying desired altitude, airspeed, heading, etc.

Both RQ-4 and RQ-1 have two major components to the system: the air vehicle and the ground station. The communications between the two are provided by either a satellite communications or a line-of-sight link. The line-of-sight link, as the name implies, requires an unimpeded straight line between the air vehicle and the ground station. The range is limited by the power of the transmitter and the curvature of the Earth to about a hundred nautical miles.
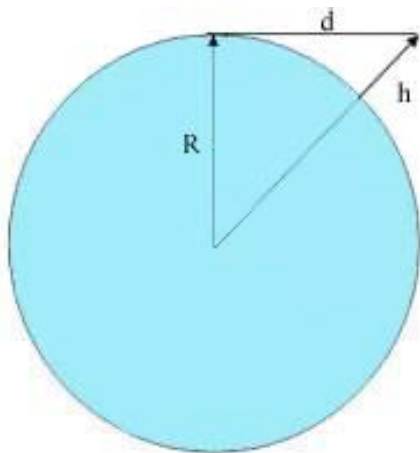
Figure 9. *Earth horizon.*

Referencing Figure 9, *R* is the radius of the Earth, *h* is the height of the air vehicle, and *d* is the geometric distance between the ground station (at sea level) and the air vehicle. Pythagorean Theorem gives the following relationship,

$$d^2 = (R+h)^2 - R^2 = 2Rh + h^2$$

Since h is much less than R, d can be approximated as,

$$d \approx \sqrt{2Rh}$$

However, radio signals propagate in a curved line due to atmospheric diffractions. A factor k is used to adjust for this; under normal weather conditions, k of 4/3 is used. If h is given in m and d is in km, distance can further be simplified into

$$d \approx 4.12\sqrt{h}$$

If the air vehicle is flying at 7000ft, the range of the link is 100nmi. With buildings and terrain in sight, the air vehicle must be flying at medium altitude to maintain line-of-sight. The assumption of 100nmi is carried through for the rest of the latency analysis.

The limited distance involved does have an upside in that the lag between a command input at the ground station and its execution at the air vehicle is markedly less than that achieved through a satellite link. The satellite link provides operational flexibility by allowing the air vehicle to be operated from almost anywhere in the world. However, this comes at the price of increased electrical power and physical space requirements for larger transmitters to reach the satellites. Both RQ-1 and RQ-4 have line-of-sight and satellite communications (K_U band) capabilities.

An idealized calculation can be made to show the difference in the latency. By neglecting atmospheric effects, which among other things, makes the propagation speed lower, the speed of the radio signal is idealized to be the speed of light (299,792,458 m/s). The range of a line-of-sight link is assumed to be a hundred nautical miles. The satellite is assumed to be in geosynchronous orbit (35,786 km above the mean sea level) like most other communications satellites. Note that the command sent has to travel from the ground station to the air vehicle, and then the results have to travel back to the ground station; in other words, the distance concerned is

Table 1. *UAS unit cost adjusted for FY2011.*

	Adjusted per Unit Price ($ millions)
RQ-4[8]	218
MQ-1[3]	56.1
Ryan 147[11]	12.7
D-21[11]	53.1

twice the physical distance between the human operator and the air vehicle. The calculations show a travel time of 1.2 milliseconds for a line-of-sight link and 0.24 seconds for a satellite link. While this does not account for other latency factors such as processing time, the numbers do illustrate that the satellite latency can be as much as three-orders of magnitude higher. The implications are immense for the "stick-and-rudder" controls of the RQ-1. The latency is in effect adding phase lag into the system which can easily lead to pilot induced oscillation during closed-loop high gain tasks like landings as shown by the operational history of the RQ-1 (discussed in the next section).

Both RQ-1 and RQ-4 represent major monetary investments. Table 1 shows the cost of a selected few UASs described above. In addition to the dramatically higher cost, the technologies utilized
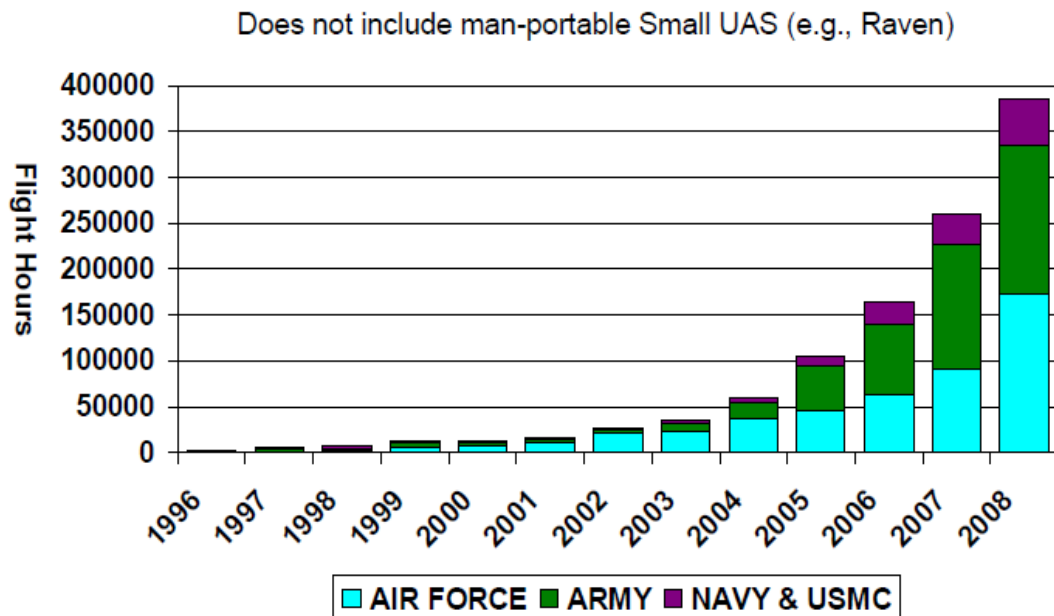
Figure 10. *UAS flight hours*[43].

are also more sensitive. While a lost D-21 may allow opponents insight into the hypersonic engine or heat distribution, no vital electronics equipment or encryptions are endangered. In contrast, the low observable techniques applied on the RQ-170 lost near Iran in December 2011, negatively impacted a key component in the technological dominance of the US. The stakes in improving UAS survivability have increased along with rising cost and ever-increasing sensitivity of onboard technology.

Meanwhile, the operations tempo has dramatically increased in the last decade. The exponential trend seen in the MQ-1 (Figure 7) is not limited to one system or service. Figure 10 shows the exponential trend across all service. The rise of UAS in comparison with manned missions can be seen in Figure 11. While flight hours for manned assets have grown linearly, the hours for Predator/Reaper grow exponentially. The increased flight hours naturally lead to higher attrition rates, which is discussed in the next section.

As the UAS matured during the last century, it developed from disposable, one-use vehicles to
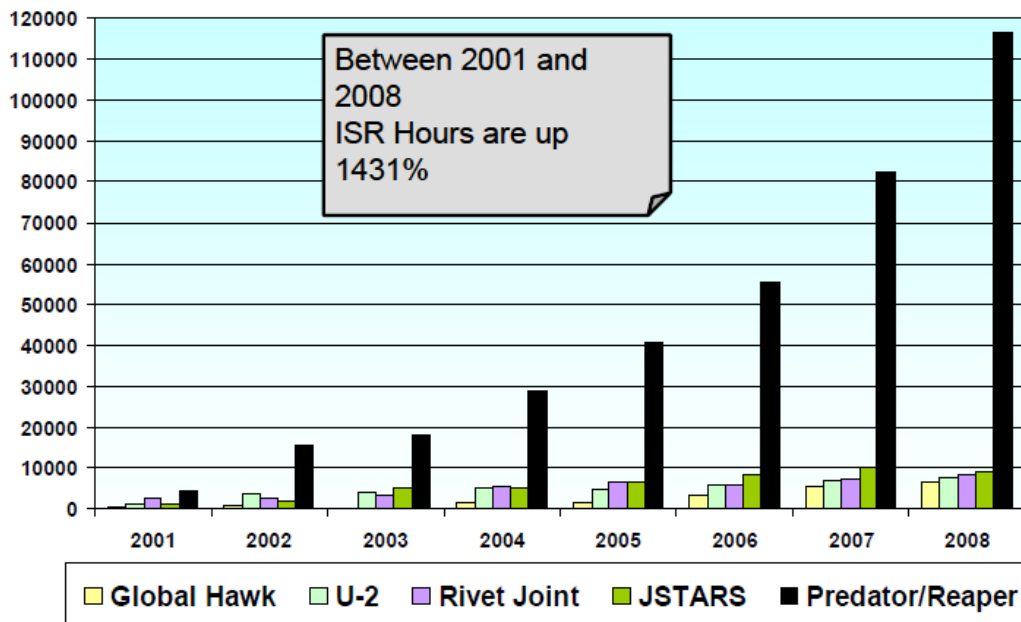
Figure 11. *CENTCOM ISR flight hours[19].*

intricate systems as expensive and strategically valuable as manned platforms. During the last decade, UASs were thrust into the frontlines of the War on Terror and saw their use increase exponentially. Given the high monetary and technology investments as well as the political sensitivity of their missions, UAS survivability has risen in importance from negligible to the front page of priorities.

B. **PROBLEM STATEMENT**

In face of unknown anomalies or battle damage, UAS pilots face additional challenges in diagnosing and correcting problems in comparison to their counterparts on manned platforms. Pilots of manned platforms enjoy high-rate quantitative cues in the form of instrument readouts. They also have the
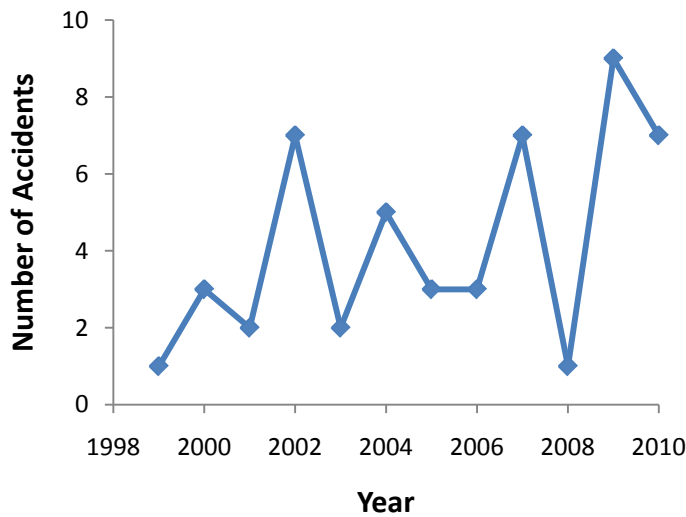
Figure 12. *UAS mishap rate.*

qualitative "seat-of-the-pants" feel. Since the data is displayed within the air vehicle, there is little to no lag in displaying system statuses; and with the modern electronic flight information systems, almost any parameter can be displayed to assist in troubleshooting. Even if the qualitative cues fail to provide sufficient information, the "seat-of-the-pants" feel might give clues that can help recover the air vehicle. In contrast, UAS operators are burdened with a limited data set with noticeable time delay. The communications link between the air vehicle and the ground station has limited bandwidth which restricts the number of parameters displayed to the

operator. In addition, the satellite link entails latency in the tenth-second order. The negative implications upon the flying qualities of the system are illustrated in the next paragraph.

The United States Air Force classifies mishaps according to severity into A, B, and C. Class A is the most severe category where the damage is more than $1,000,000, or the aircraft is damaged beyond repair, or fatal injury / permanent disability is incurred. A summary of all class A mishaps involving UASs is summarized in section A of the appendix and is plotted in Figure 12. The upward trend in Figure 12 matches the increased flight hours shown in Figure 11. Of these mishaps, there are five cases of pilot induced oscillations; all happened aboard the Predator family with the "stick-and-rudder" style control. One of the accident reports explicitly points to the fact that "the Predator's lack of sensory cues contributed to the MP's [mishap pilot] decision to continue to go-around attempt after the MRPA [mishap remotely piloted aircraft] had become unflyable."[1]

The restricted situational awareness of UAS not only causes trouble at closed loop high gain tasks, it also affects the troubleshooting ability of the operator. Because the operator was "confused over the flight characteristics of the MPRA [mishap remotely piloted aircraft], the MP [mishap pilot] erroneously concluded that there was a control problem…"[2] During the troubleshooting process, the operator turned off the stability augmentation system which resulted in "the aircraft banking excessively in a nose down attitude, thereby losing its satellite communication link. The MP was unable to regain control of the aircraft, which subsequently impacted the ground."[2]

In order to increase survivability of the UAS, some function must exist onboard the vehicle to either adapt to the anomaly, or explicitly compare the current performance against the expected values in order to check for anomaly. These functions must occur on the vehicle due to the

bandwidth limitation and latency of the communications link between the air vehicle and the ground station.

C. **ADAPTIVE CONTROL IS INSUFFICIENT**
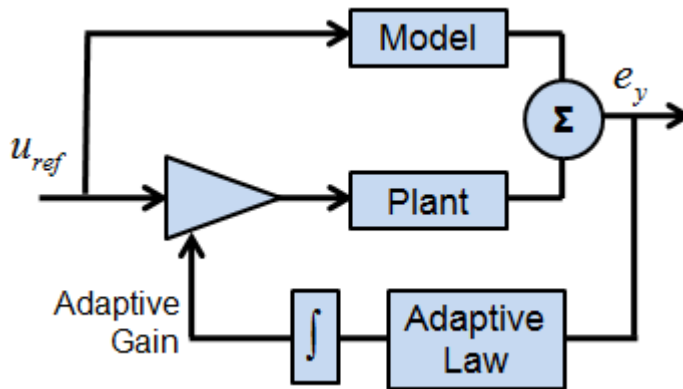
Figure 13. *Explicit model following architecture.*

Much work has been done in the field of adaptive control to account for anomalies. The advantage of adaptive control is that any reasonable anomaly will be washed-out and become transparent to the human operator. However, with the exception schemes with an embedded SID algorithm, none of the adaptive control schemes can easily lead insights into the flying qualities.

Adaptive control schemes attempt to overcome plant or actuator uncertainties by adjusting the gains on-line. An example of adaptive control (explicit model following) is displayed in Figure 13. One adaptive scheme that does not guarantee stability is mechanized as the Honeywell MH-96 controller and is used on the third vehicle of the X-15 (the other two vehicles are fitted with conventional gain-scheduled controllers). The MH-96 increases the gain until limit cycle oscillations are observed, at which point the gain is lowered. As the gain approaches the critical level and the rudder chatter amplitude at the limit cycle frequency increases, the gain is lowered[9]. Theoretically, such a controller gives the highest performance possible by keeping the gain at critical level, while maintaining stability by lowering the gain when oscillation is observed.

However, this scheme does not guarantee stability; this fact is reflected in the 1967 crash of the X-15-3 during which the elevator entered into a limit cycle oscillation. There is no warning that the controller will fail; and the degradation is not graceful.

Another adaptive scheme is the explicit model following scheme. Usually, the closed-loop behavior of the plant is used as the reference model. The control inputs are given to the reference model and the error between the model and the plant is used to drive the adaptive law. The law calculates the appropriate plant input so that the system behaves like the reference model. Such a scheme allows the control system to overcome anomalies such as loss of control effectiveness or nonlinearity. The stability of such a scheme is proven in a dissertation.[20] Note that while stability and performance are maintained, there is no indication of how hard the controller is working to keep the air vehicle flying or how many margins there are; in other words, there is no way to characterize system robustness.

Plant inversion is a modern control technique that, depending on implementation, can contain an adaptive flavor. Instead of driving the error between the model and the plant to zero, this scheme puts an inverted plant in-line with the plant to achieve pole-zero cancellation. After the system dynamics are taken out, pole placement is used to achieve desired dynamics. If the plant is non-minimum phase, the right-half plane zero becomes a right-half plane pole in the inverted plant. If the model of the plant is exactly correct, pole-zero cancellation occurs; otherwise, there exists a pole in the right-half plane resulting in an unstable closed-loop system. While accurate modeling can prevent such a disastrous occurrence, anomalies can unexpectedly change that. By adding an adaptive component in the form of a system identification algorithm, pole-zero cancellation can be reliably achieved[42]. Metrics on system robustness are calculable only in the case of adaptive plant inversion.

With the exception of adaptive schemes having an embedded SID algorithm, none of the control strategies can easily estimate the margins. While the performance of the closed loop system can be established by examining the reference model in explicit model following and pole placement techniques (even though the latter is not an adaptive scheme), the robustness cannot be easily established. For systems using other control schemes (such as traditional PID controllers), there is no easy way to obtain a flying qualities estimate on-line. FQ-BIT addresses this deficiency.

D. **FQ-BIT**

FQ-BIT remedies the deficiency by delivering a plug-and-play software package that is applicable to air vehicles with any control scheme: classical or modern, fixed-gain or adaptive. Once damage or anomalies occur, the control system is either capable of returning the air vehicle (AV) to straight and level flight, or it becomes uncontrollable. In the uncontrollable case, the AV either crashes despite best efforts, or the operator chooses to terminate without consulting FQ-BIT due to time constraints. In either case, FQ-BIT is not involved in the decision process.

If the AV is able to maintain straight and level flight after sustaining damage, the AVO can execute the FQ-BIT. If the AV is observable, controllable, stable, and robust, the AVO is advised to return-to-base (RTB). If the BIT is invalid, the AVO is advised to re-run the BIT. If the AV is not observable, controllable, stable, or robust, the AVO is advised to terminate the AV at the earliest opportunity. As the air vehicle approaches the landing phase, the AVO is advised to progressively step down in altitude and to execute FQ-BIT at each altitude step. This is done as a build up to the culminating event where the air vehicle simulates a landing approach at a safe flight condition in order to evaluate the flying qualities required for recovery. If the flying qualities are within acceptable limits, the AVO is advised to recover the air vehicle.

The decision tree presented above is graphically summarized in Figure 14.

16
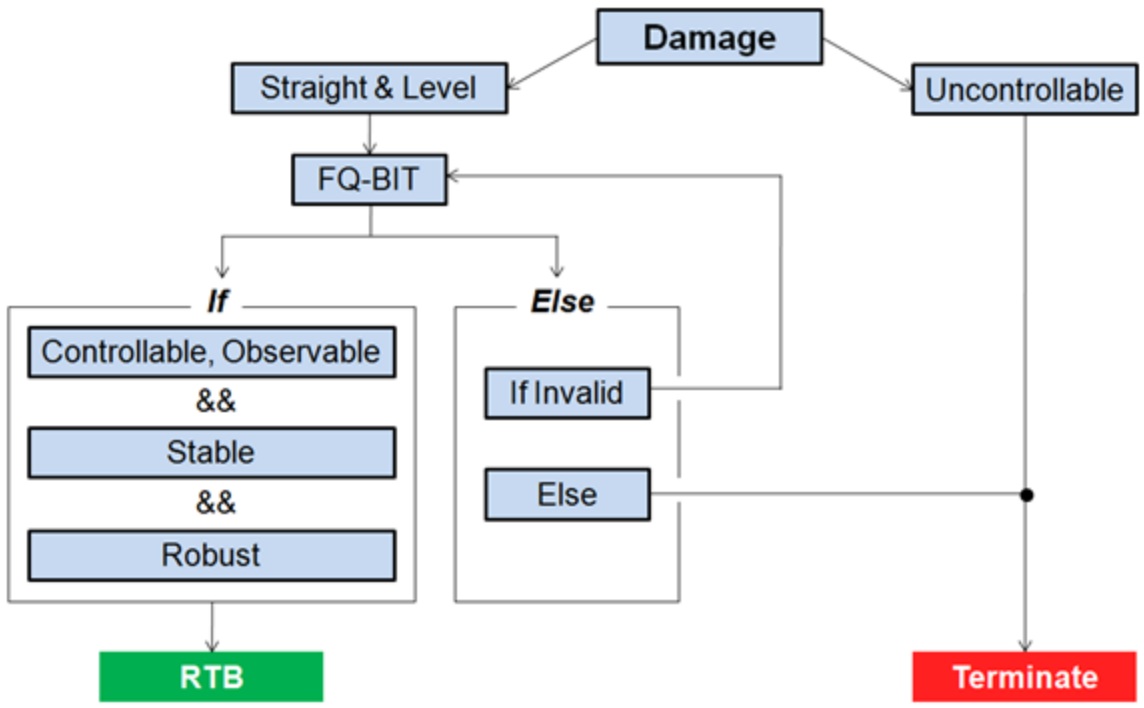
Figure 14. *ConOps.*

III. SYSTEM DESCRIPTION

The architecture of the FQ-BIT is outlined in Figure 15. A known input waveform (u) is injected into the air vehicle and the pertinent states recorded (albeit corrupted by sensor error into $y+\varepsilon$). The SID algorithm outputs a linear estimation of the air vehicle dynamics ($\hat{A}, \hat{B}$), which are then evaluated by the flying qualities analysis block, and the metrics are displayed to
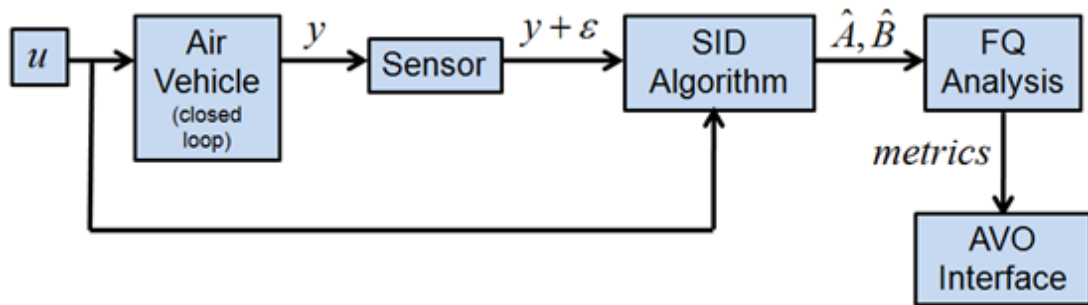
Figure 15. *FQ-BIT architecture.*

the air vehicle operator (AVO). FQ-BIT assumes that the longitudinal and lateral modes of the UAS are decoupled, and estimates them separately. Because the roll and yaw axis are coupled, they are estimated together in the lateral axis.

Given the limited communications bandwidth available between a UAV and the ground station, FQ-BIT process data onboard the AV and only transmit the results to the ground station. As such, FQ-BIT must be computationally simple so as not to burden the flight computers. This requirement not only influences the SID algorithm selection, but also the order of the estimation model. This implication is further discussed in SID algorithm subsection below.

The following subsections detail the theory and mechanization of each block illustrated in Figure 15; the process of arriving at these choices is discussed in the results and analysis sections.

A. **INPUT WAVEFORM**

The purpose of the input waveform (*u*) is to excite the plant so that output states can be measured for use by the system identification algorithm. An optimal waveform choice maximizes the informational content in all the output states. At the same time, the waveform should avoid destabilizing the plant with too much input energy.

Most modern air vehicles, manned or unmanned, exhibit open-loop relaxed stability or even instability; this in turn requires the control system to be operating at all times. The dynamic instability stems from a wide variety of requirements, from maneuverability (high rates can be achieved with an unstable system, as exemplified with the Lockheed F-22), to low-observability (shape of the airframe as determined by radar signature may not be aerodynamically stable, such as the case for the Northrop B-2), and even aerodynamic efficiency (lower trim-drag can be achieved with low static-stability margin, as was done with the Airbus A-320). With this in mind, the input waveform must excite the closed-loop dynamics of both the airframe and control system.
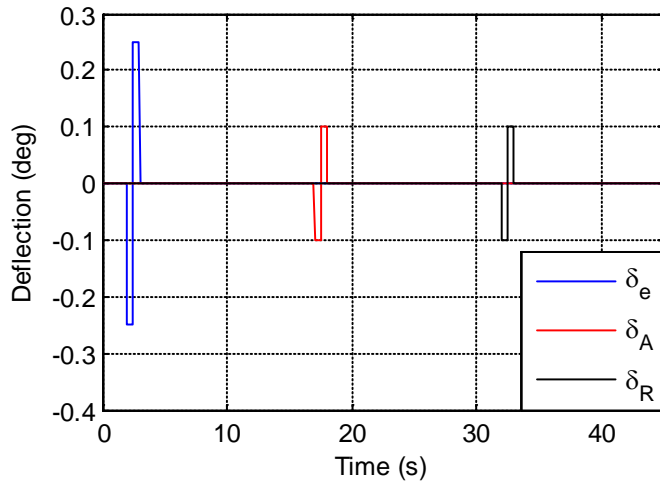
Figure 16. *Doublet Input*

Three artificial waveform candidates (pulse, 3-2-1-1, doublet) and one natural waveform candidate (turbulence) were considered for use by FQ-BIT. After a selection

process discussed in the appropriate subsections of section IV, V, and VI, the Doublet waveform (depicted in Figure 16) was chosen for its balance of stability and accuracy. The amplitude is tuned as a compromise between exciting the system enough to produce good estimates, and avoiding destabilizing the system. For the longitudinal axis, the doublet is injected only to the elevator. For the lateral axis, the doublet is injected into the both aileron and rudder. The aileron is first perturbed, then fifteen seconds later the rudder is perturbed. The separation between the aileron and rudder perturbation allows for the transient motions to dampen out so as not to corrupt the results of the rudder estimation results.

B. **AIR VEHICLE PLANT**

Large UASs serve strategically important roles in intelligence-surveillance-reconnaissance and strike missions are reflected in the large monetary and intellectual investments (the reader is referred to the history subsection of the introduction for statistical information). In addition,

Figure 17. *RQ-4A.*

their large size is more likely to cause casualties in the event of a mishap. Given this, FQ-BIT is focused on serving large UASs; the comparative size of a large UAS is illustrated by an RQ-4A in company with a thirteen passenger Beechcraft C-12 in Figure 17.

C. **SENSOR**

The air vehicle must have sensors to measure the following longitudinal states $[\alpha, q]$ and lateral states $[\beta, \phi, p, r]$ for use by the SID algorithm. In addition, all perturbed control surface positions $[\delta_e, \delta_a, \delta_r]$ must also be measured and recorded; this requirement should be easily

fulfilled as most actuators already incorporate a position measurement for use in their internal command tracking control system. Likewise, the Euler angle rates $[p,q,r]$ and roll angle $[\phi]$ should easily be obtainable from the onboard navigational unit(s). The angles between the body and stability axis $[\alpha, \beta]$ are more difficult to obtain accurately.

In comparison to the other measurement requirements, $[\alpha, \beta]$ are not necessarily required by the basic control architecture. While one can safely assume the navigational and actuator needs will provided $[p,q,r,\phi]$ and $[\delta_e, \delta_a, \delta_r]$ to FQ-BIT at no additional cost, $[\alpha, \beta]$ are not required by any function. The control system may require them for stability augmentation, but that necessity cannot assuredly exist on all UAS of concern. As such, FQ-BIT levies additional requirements on the UAS to either possess α and β sensors, or to have a method of synthesizing the measurements with a state estimator. Note that the following equations

$$\alpha = \theta - \gamma$$

$$\beta = \psi - ground\ track$$

are unsuitable for FQ-BIT use. The α equation is only valid for wings-level straight and level flight, and the transients during a pitch doublet make that calculation invalid. The β equation is invalid because it assumes that no wind exists (in other words, the air mass is not moving). While wind may be correct for post-flight with accurate weather information, it is impractical to do so in an operational environment real-time.

D. **SID ALGORITHM**

The system identification (SID) component estimates the plant which is then analyzed for flying qualities. SID algorithm assumes the lateral and longitudinal dynamics of the air vehicle are decoupled and processes them separately. For the longitudinal mode, the states $[\alpha,q]$ and input δ_e are used for SID. For the lateral mode, the states $[\beta,\phi,p,\psi,r]$ and the input $[\delta_a,\delta_r]$ are used.

Three candidates (equation error in time domain, equation error in frequency domain, and Extended UD Factorized Kalman Filter) are evaluated in the appropriate subsections of section IV, V, and VI with EUDKF being chosen for use. The following paragraphs describe the implementation of EUDKF in detail.

1. *Extended UD Factorized Kalman Filter*

The system identification (SID) algorithm of choice is the Extended Kalman Filter in UD factorized form (hereafter referred to as EUDKF), which is a member of the Kalman Filter family. Kalman Filter uses statistical probability to meld the measurements and the model together. It is a recursive process that assumes a Gaussian white noise distribution. The Kalman Filter only operates upon the data one time step prior and hence has low memory access requirement; in contrast, batch filters require the entire measurement and hence have high memory access requirement. Although generally applied towards state estimation, the Kalman Filter family can also be used for system identification by augmenting the state matrix with the elements of *A* and *B* matrices.
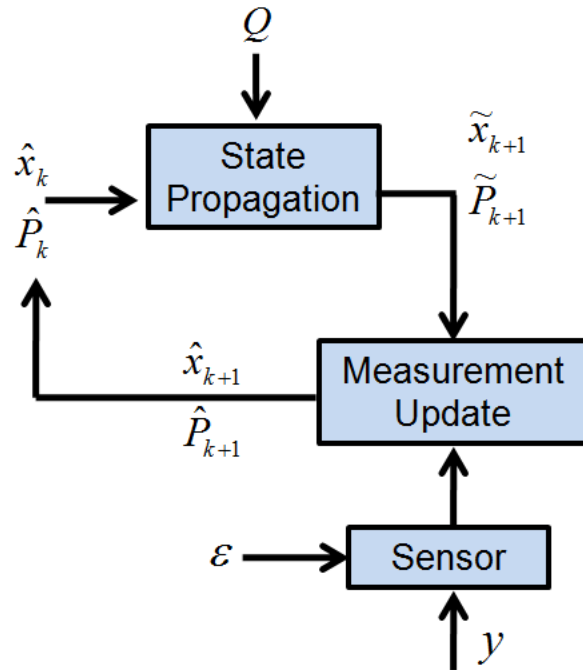
Figure 18. *Kalman filter architecture.*

The Kalman Filter architecture is displayed in Figure 18. From the last state estimate, the state is propagated forward in time using the state transition matrix ϕ such that $\tilde{x}_{k+1} = \phi \tilde{x}_k$, where ϕ is given as the matrix exponential of the state matrix *A* ($\phi = e^{At}$) . To save computational resources, a truncated Taylor series expansion of the exponential can be used such that $e^{At} = I + At$.

Alternatively, any numerical technique (such as the fourth-order Runge-Kutta scheme) can be used to propagate the states as $\dot{x} = Ax$. Since any nth-order ordinary differential equation can be written as n-number of 1st order differential equations, the solution to any nth-order ordinary differential equation can be written as a matrix exponential problem. While the states are propagated, the covariance matrix P is also extrapolated. This P is used in the measurement update step. As implemented in FQ-BIT, the initial augmented state matrix guess is that of the initial states augmented by the nominal A and B matrices, and the initial covariance P is guessed to be 50%.

Q is the covariance matrix of process noise. It is mathematically represented as a square matrix of size nxn if the state vector is of size nx1. Tuning of the Kalman Filter can be performed by adjusting the magnitude of each Q element. Each noise injection must be proportional to the magnitude of each individual state. Assuming a state vector of [10; 1000], the diagonal of the Q matrix could be a scalar multiple of [1; 100]. If improper noise magnitudes are specified, the Kalman Filter could track the largest magnitude state(s) and ignore others. Too little excitation could lead to filter "smugness" in which the filter erroneously trusts the model too much and does not pay attention to the measurement updates. Alternatively, too much excitation could drive the system unstable.

Once the state is propagated to the next time step, and the state measurements are available through the sensor, the next step is to combine the two. Using statistical probability, the Kalman gain K is calculated using a combination of the covariance matrix P, predefined measurement noise matrix R, and sensor matrix C such that $K = \tilde{P} - C^T[C\tilde{P}C^T + R]^{-1}$. If the covariance is high, this indicates a high confidence in the model or the filter considers the measurement is an outlier; as such, K approaches zero and the measurement is ignored. The states are updated using the Kalman gain as such, $\hat{x} = \hat{x} + K[y_m - C\tilde{x}]$, where y_m is the measurements. After this, the

covariance matrix is updated as well using the equation $\hat{P} = [I - KC]\tilde{P}$ where I is the identity matrix.

The "Extended" flavor of the Kalman Filter differs by having an A matrix linearization step at each iteration, whereas the plain Kalman Filter has a predefined, fixed A matrix. The linearization step allows the filter to account for plant non-linearity. In the FQ-BIT implementation, the linearized A matrix is found by numerically taking the Jacobian of the non-linear function around the current states (first-order accurate in the Taylor series expansion sense); the sensor matrix could be likewise linearized if necessary. An Extended Kalman Filter is a non-optimal solution for nonlinear systems because it uses a linear approximation of a nonlinear system. In contrast, plain Kalman Filter is the optimal solution for a linear system as no system approximation is utilized. The Extended Kalman Filter also incurs penalties in the form of computational requirements associated with the linearization step. As implemented in FQ-BIT, each element of the Jacobian matrix is calculated by the central difference method where $\dot{f} = \frac{f(x+\Delta x) - x(x-\Delta x)}{2\Delta x}$. The states are propagated forward using the matrix exponential method.

Rounding errors can cause the Kalman Filter to diverge. If the covariance matrix P has a small positive eigenvalue to be computed as zero or a negative number, it thereby turns the covariance matrix P indefinite (it should be positive-definite). The UD factorized flavor of the Kalman Filter addresses this problem by operating on P in a unit-triangular and diagonal (UD) decomposed form. Because P is a positive-definite matrix, the square-root can be calculated. By performing operations on the square-root and then recombining, P is guaranteed to be positive-definite. However, square-root operations are slightly more expensive than the divisions used by the UD-decomposition[37]. UD-decomposition retains the numerical properties of the square-root while requiring less computational resources. P can be decomposed using the following equation $P=UDU^T$ where U is the unit-triangular component and D is the diagonal component. The bonus

of such factorization comes in storage space requirements: UD-decomposition requires only half the word length compared to explicitly calculating P.

Divergence could occur due to things like incorrect initial guesses, nonlinearity, and rounding error; EUDKF and tuning strategies are chosen to avoid such problems. To handle incorrect initial guesses, the output from other SID algorithms can be used to initialize the KF; but this negates the advantage of a recursive process. As implemented in FQ-BIT, EUDKF makes an initial guess with the nominal values and 50% covariance. While the large initial covariance can increase convergence time, the increased robustness is worth the trade-off. The mechanization of the EUDKF is summarized below.

State and Covariance Propagation

The State Vector propagates through time according to the transition matrix ϕ where $\hat{x}$ is the current state estimate, $\tilde{x}$ is the extrapolated state at the next time step, and w_k is the process noise.

$$\tilde{x}(k+1) = \phi(k, k+1)\hat{x}(k) + w_k \tag{1}$$

Using the definition of covariance matrix $P(k) = E\{x(k)^T x(k)\}$ and the assumption that state error and process noise (Q) are uncorrelated, the error covariance matrix can be written as

$$\tilde{P}(k+1) = \phi \hat{P}(k)\phi^T + GQG^T \tag{2}$$

However, explicitly calculating P could lead to divergence, so the upper triangle term U and diagonal term D are introduced such that

$$\hat{P} = \hat{U}\hat{D}\hat{U}^T \tag{3}$$

With the following definitions:

$$V = [\phi\hat{U} \mid G]$$

$$V^T = [v_1, v_2,..., v_n]$$

$$\overline{D} = diag[\hat{D}, Q]$$

The extrapolated P can be written as

$$\tilde{P} = \tilde{V}\tilde{D}\tilde{V}^T \tag{4}$$

25

where U and D are recursively calculated with the following algorithm.

```
for (j=1...number of states) {
```

$$\tilde{D}_j = v_j^T \overline{D} v_j$$

```
    for (i=1...j-1) {
```

$$\tilde{U}_{ij} = \left(\tilde{D}_j^{-1}\right) v_i^T \overline{D} v_j$$

$$v_i = v_i - \tilde{U}_{ij} v_j$$

```
    }
```

```
}
```

Linearization

Note that for a non linear system described by the following state equations where Θ is the vector of unknown parameters,

$$\dot{x}(t) = f[x(t), u(t), \Theta] \tag{5}$$

$$y(t) = h[x(t), u(t), \Theta] \tag{6}$$

EKF linearizes the system around the current estimate such that,

$$A(k) = \left. \frac{\delta f_a}{\delta x_a} \right|_{x_a = \hat{x}_a(k), u = u(k)} \tag{7}$$

$$C(k) = \left. \frac{\delta h_a}{\delta x_a} \right|_{x_a = \hat{x}_a(k), u = u(k)} \tag{8}$$

And the transition matrix is given as,

$$\phi(k) = \exp[-A(k)(t_{k+1} - t_k)] \tag{9}$$

Hence the extrapolated state vector can be found using equation 1 using ϕ from the equation 9.

Measurement Update

After a new measurement is available, K the Kalman gain is calculated as such,

$$K = \tilde{P} - C^T \left[C\tilde{P}C^T + R \right]^{-1}$$

where R is the measurement noise covariance matrix. The state and error covariance matrix is updated with the following equations. y_m is the measurement. The sensor matrix C used is from equation 8.

$$\hat{P} = [I - KC]\tilde{P} \qquad (10)$$

$$\hat{x} = \hat{x} + K[y_m - C\tilde{x}] \qquad (11)$$

2. $\hat{A}, \hat{B}$ Matrix Size

A lower-order estimation is used for FQ-BIT in the longitudinal states in order to reduce computational requirements. Textbooks typically linearize air vehicle plants into four states: $[VTAS, \alpha, \theta, q]$ in the longitudinal mode and five states $[\beta, \phi, p, \psi, r]$ in the lateral mode, and inputs are linearized to two states $[\delta_e, \delta_{th}]$ in the longitudinal mode and two states $[\delta_a, \delta_r]$ in the lateral mode[25, 35]. However, the physics of large UAS allow an order reduction to just $[\alpha, q]$ in the plant and $[\delta_e]$ for the input.
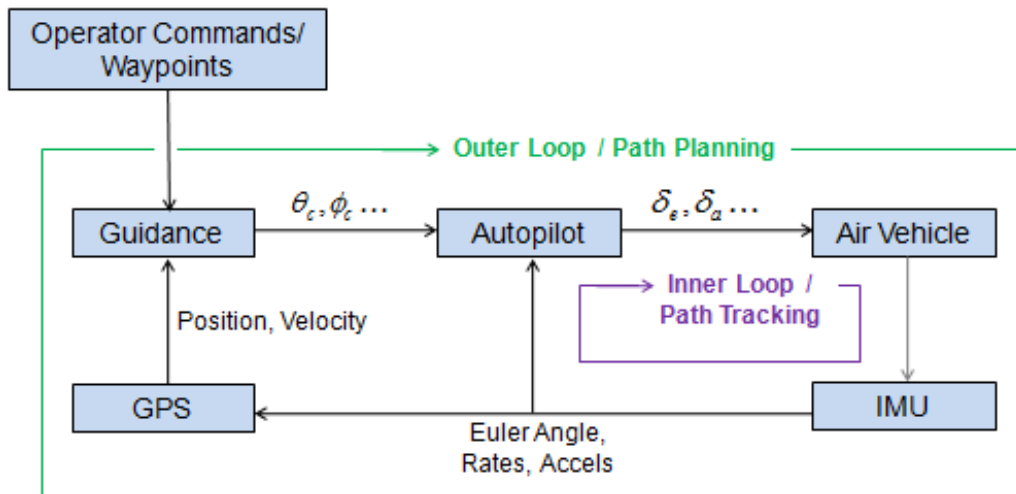
Figure 19. *Typical UAS control system.*

A typical control system for UAS is illustrated in Figure 19. In the longitudinal mode, the autopilot forms a tight inner loop with altitude and airspeed tracking. This augmentation effectively suppresses the phugoid mode. This is illustrated in Figure 20 where the open-loop dynamics is that of a Cessna 172 (shown in the appendix) with an initial VTAS disturbance of 5kt; the augmentation is achieved through pitch rate feedback architecture with a gain of 20 in the feedback path.
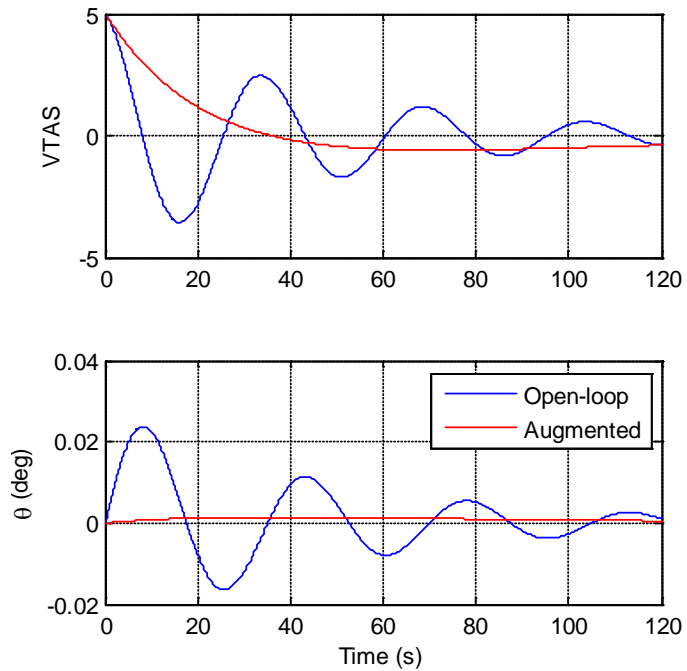
Figure 20. *Phugoid suppression.*

The Phugoid is a "slow" mode in comparison to the short period mode with the dynamics primarily showing up in the VTAS and θ states. Similarly, the throttle input has a comparatively higher lag time than the elevator and only minimally affects the short period. The long period (roughly forty seconds in Figure 20) in comparison to the maneuver length (fifteen seconds), as well as the fact that the autopilot / inner-loop controller is never disengaged on a UAS; the phugoid mode can be removed from the longitudinal estimation. Hence, the state vector is reduced to $[\alpha, q]$ and input vector reduced to $[\delta_e]$. The five lateral states cannot be reduced because all states are necessary to capture the coupled rolling and yawing motions.

Because of the tight inner-loop controller, the phugoid mode can be ignored. The relevant longitudinal states and input are $[\alpha,\ q]$ and $[\delta_e]$; the relative lateral states and inputs are $[\beta,\phi,p,\psi,r]$ and $[\delta_a,\delta_r]$.

E. **FQ ANALYSIS**

To access the flying qualities (FQ) of the air vehicle, the stability / performance, robustness, controllability, and observability of the system must be determined. Since the SID algorithm outputs the linear closed-loop dynamics of the air vehicle in the state-space form, all analysis will be performed in the state-space form as well.

1. *Stability / Performance*

The pole locations (in the Laplace domain) or eigenvalues of A (in state-space representation) determine the stability of the system. Considering the system with no external input or disturbances so that $\dot{x} = Ax$, the system is stable if and only if all eigenvalues of A are inside the left half of the complex plane. In other words, the real parts of the eigenvalues of A are all negative. Assuming A is diagonalizable with all the eigenvalues in the left half plane, the following relationship exists:

$$\|e^{At}\| \leq k_0 e^{\sigma t}$$

where k_0 is the condition number and σ is the largest real eigenvalue of A. Because all the eigenvalues of A are in the left half plane, $\sigma < 1$. Multiplying both sides of the inequality with the magnitude of the initial state x_0, and applying the fact that the solution $\dot{x} = Ax$ to is $x = e^{At}x_0$, the following can be seen:

$$\|e^{At}x_0\| \leq k_0 e^{\sigma t}\|x_0\|$$
$$\|x(t)\| \leq k_0 e^{\sigma t}\|x_0\|$$

If the system is exponentially stable the steady-state norm of *x(t)* should be zero as time approaches infinity. Observing the right hand side of the equation above, one can see that the quantity approaches zero as time approaches infinity because of the negative exponential (σ is a negative number). Thus, the following relationship exists:

$$\lim_{t \to \infty} \|x(t)\| \leq \lim_{t \to \infty} k_0 e^{\sigma t} \|x_0\|$$

$$\lim_{t \to \infty} \|x(t)\| \leq 0$$

However, the definition of a vector norm states that $\|x(t)\| \geq 0$. For both the relationship derived above and the vector norm definition to be true, the following must be true,

$$\lim_{t \to \infty} \|x(t)\| = 0$$

Hence, given the system with dynamics such that $\dot{x} = Ax$, the system is stable if and only if the real part of all the eigenvalues of A are negative.

Armed with the knowledge above, the stability of the system is easily established. As implemented in FQ-BIT, the eigenvalues of the A matrix is calculated and the real and imaginary parts are determined. If all the real parts of the eigenvalues are negative, then the system is stable. However, the FQ-BIT does not explicitly check for stability as a Boolean. Instead, this is implicitly established along with performance section.

The flying qualities of an air vehicle are specified in damping frequency (ω_d) and damping ratio (ζ) for second order responses, time constants (τ) for first order responses in MILSPEC 8785. Since the desirable metrics derive from stable systems, metrics from unstable systems will lie outside the desirable region along with stable
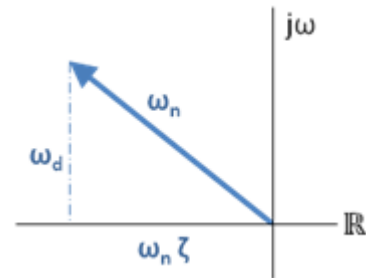
Figure 21. *FQ stability metrics.*

systems that do not perform up to par. Hence, by just analyzing the flying qualities metrics, the stability and performance of the air vehicle can both be established at the same time.

For a second order response, the damping frequency and ratio can be calculated based upon the pole locations. The relationship is illustrated in Figure 21. The imaginary part of the eigenvalue is the damping frequency. The modulus of the eigenvalue gives the natural frequency. The real part of the eigenvalue is the product of the damping ratio and natural frequency. The short-period, Dutch-Roll, and Phugoid modes are second order responses and can be analyzed with this method.

For first order response, approximations are needed to convert the pole locations to time constants. Note that poles for first order responses lie on the real axis of the root locus plot; thus, the damping frequency is zero and damping ratio is one. Observe that the magnitude of the eigenvalue is the natural frequency of the first order response. The following approximations[25] between time constant, rise time, and the natural frequency, and eigenvalue (λ) are used to calculate the metric,

$$t_r \cong \frac{1 + 1.1\zeta + 1.4\lambda^2}{\omega_n} = \frac{3.5}{|\lambda|}$$

$$\tau \cong \left(1 - \frac{1}{e}\right) t_r \frac{2.2}{|\lambda|}$$

The desirable flying qualities for manned platforms are outlined in MILSPEC 8785. However, these requirements are written for pilots having direct "stick-and-rudder" inputs to the UAS; while having relevance to a remotely-piloted-capable platform such as the RQ-1/MQ-9, it is not pertinent to the RQ-4 with "keyboard-and-mouse" inputs. The difference in input style is illustrated in Figure 22. With the RQ-4 style of inputs, the human operator is issuing high level commands like desirable heading and altitude while the inner-loop autopilot is performing the tracking task; in the RQ-1, the human operator both issues high level commands and performs tracking. Because of different styles of human-machine interface, the desirable flying qualities are left to the UAS designers to specify.

Because different turbulence conditions might impose different response times, the desirable performance metrics are split into three progressively stringent categories of light, medium, and heavy turbulence. The performance metrics calculated from the SID algorithm output are compared against the requirements and the result is a "go / no-go" for flight in different atmospheric conditions.
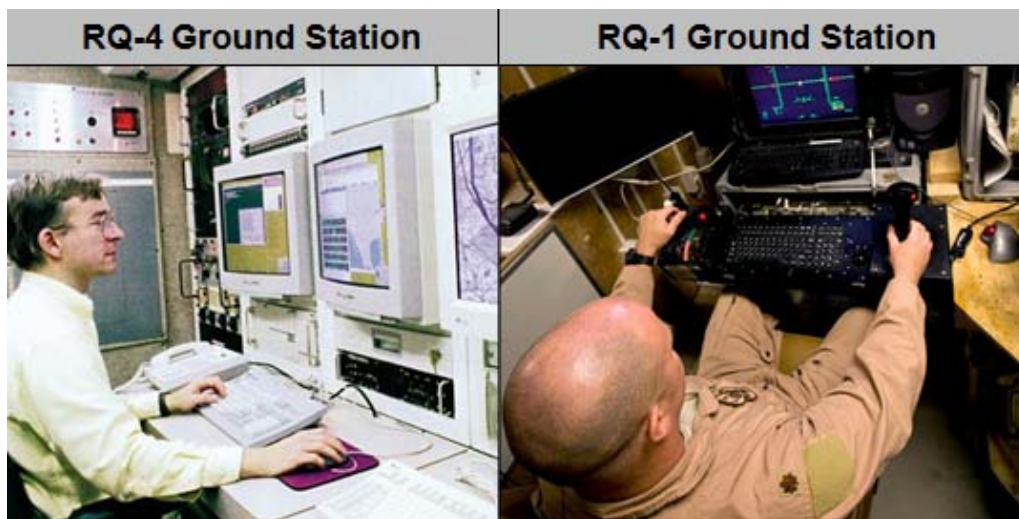
Figure 22. *Keyboard-and-Mouse vs. Stick-and-Rudder*

2. *Robustness*

Even if the system is found to be stable with adequate performance, robustness must still be established. Robustness assures stability even if the results from the SID algorithm have up to 6db and 45degs of errors in gain and phase. Referencing the
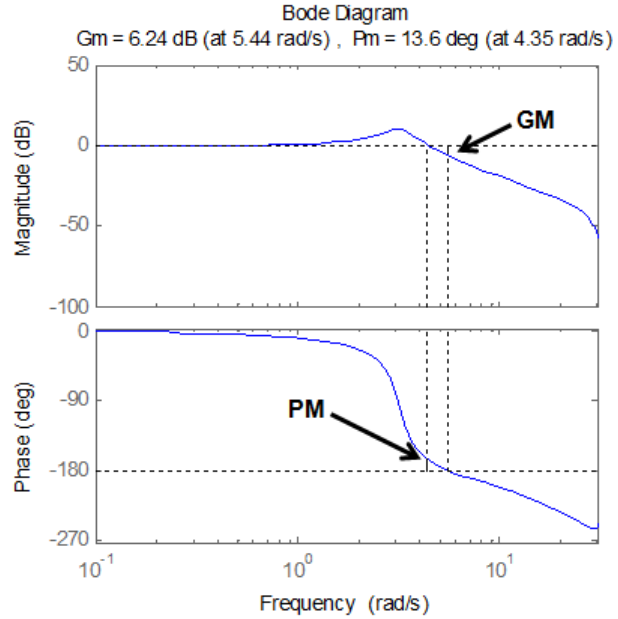
Figure 23. *Bode Plot*

system architectural diagram in Figure 15, error sources can be identified. Sensor error (ε) in form of bias, scaling, noise, or nonlinearity corrupts the state outputs. This has the cascading effect on the SID results. In addition, the SID algorithm introduces process noise by estimating a linear, reduced-order model since the "real-world" is non-linear with an infinite number of states. While the SID algorithm is tuned to give the best results despite these errors (described in the relevant test and analysis sections), margins must be built in.

The robustness of the system is characterized by the gain and phase margins. Gain margin indicates the amount of error in the modeled magnitude before system instability occurs. Assuming a linear representation at an input of 1 is *output = 3 x input*, and the system has 6dbs of gain margin, then it can be said that the model (amplification of 3) can have 6dbs of error (i.e., true amplification can be as high as 6) before the system becomes unstable. The instability is a result of the phase being 180degs apart where negative feedback becomes positive feedback. On the other hand, phase margin indicates the amount of error in the modeled phase before instability occurs. Assuming a modeled phase margin of 45degs, then the true phase lag can have an

additional 135degs before instability occurs due to the feedback switching sign. Gain and phase margins are determined from the Bode plot; their relationship is illustrated in Figure 23. The reader is referred to textbooks[25, 35] for further explanation.

MILSPEC 9490 defines the desirable gain and phase margins as 6db and 45degs respectively. Since there is no fundamental difference between manned and unmanned platforms regarding robustness, these criteria are adopted for determining robustness of the system. Once the robustness of the air vehicle is determined, a Boolean flag indicates to the human operator whether the system has the required 6dbs and 45degs robustness.

3. *Observability*

Observability is critical to the function of the control system. Given an observable system, the current states can be determined in finite time using the observed outputs. Alternatively, this means that it is possible to determine system behavior given the outputs available. If the system is unobservable, then some of the output values are not known to the controller and cannot fulfill the control specifications.

Since the SID algorithm output a linear time-invariant state-space representation of the closed-loop air vehicle, the observability matrix (O) is constructed as such where n is the number of states:

$$O = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

If the rank of the observability matrix is n (in other words, full rank), then each row is linearly independent and each state can be determined; thus the system is observable. The operator is

informed if the system is observable. If the system is unobservable and no sensor faults are indicated, then the FQ-BIT results are invalid and should be re-performed as soon as practical.

4. *Controllability*

Controllability is the mathematical dual of observability. Given a controllable system, then the states can be changed from one to another in a finite amount of time. Alternatively, this is a measure of whether all the states can be altered by a combination of inputs. If the system is uncontrollable, the control system cannot change all the relevant parameters of the flight path, which presents a danger to both the air vehicle and the surroundings.

Given a linear time-invariant state-space representation of a system, the controllability matrix (R) where n is the number of states can be written as such:

$$R = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix}$$

If R is full rank, then there are n numbers of linearly independent columns and all the states are reachable given a combination of inputs; thus the system is controllable. The operator is informed if the system is controllable.

F. **AVO INTERFACE**

The raw, numerical, results from FQ analysis in subsection E, while providing a wealth of engineering information, should not be directly displayed to the air vehicle operator (AVO). Instead of overwhelming a presumably stressed operator with a deluge of numbers, the results are presented as a series of "go / no-go", respectively green and red, flags so that the pilot can determine the flying-qualities in a glance. The interface is displayed in Figure 24.

Area I allows the AVO to select the axis to be tested; if both axes are selected, longitudinal will be tested first, then lateral. The execute button in area II is used to initiate the test(s) selected in area I. Area III is the detail status button which allows the AVO to thoroughly examine the results (further elaborated in the next paragraph). Area IV indicates whether the AV is safe to operate in differing levels of turbulence. If all three columns are green, then the AV is safe to operate in conditions up to heavy turbulence; if the light and medium columns are green but the heavy column is red, then the AV is unsafe to operate in the presence of heavy turbulence. If all three columns are red, the AVO is advised to terminate the AV as soon as practical. Area V
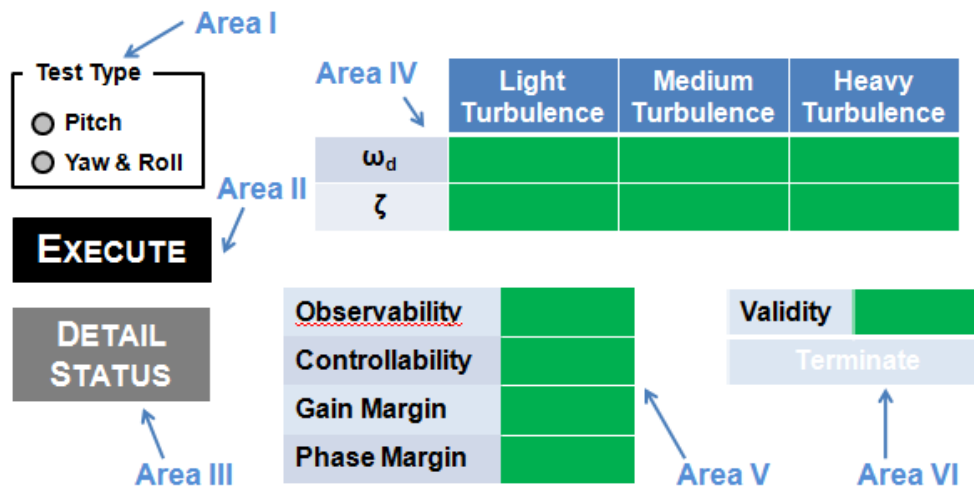
Figure 24. *AVO interface.*

displays other flying qualities metrics. Observability and controllability should be green for a safe recovery. If observability is red while no sensor faults are indicated, then the invalid flag will turn red in area VI. If either gain or phase margin flags turn red, the AVO is advised to terminate. Area VI displays the validity as well as termination recommendation. If the results are invalid, the operation should re-execute FQ-BIT at the earliest convenience. If any of the aforementioned criteria for termination are met, the terminate flag will light up and the AVO is advised to terminate. The FQ-BIT will not automatically terminate the AV.

Pitch

	Result	Light Turbulence	Medium Turbulence	Heavy Turbulence
ω_d	1.6	1.4-2.0	1.5-1.8	1.4-1.7
ζ	1	>0.5	>0.5	>0.8

	Result	Req.
GM	10	6
PM	90	45

Yaw

	Result	Light Turbulence	Medium Turbulence	Heavy Turbulence
ω_d	2.1	1.4-3.0	1.4-2.5	1.5-2.0
ζ	1	>0.5	>1.1	>1.1

	Result	Req.
GM	7	6
PM	54	45

Roll

	Result	Light Turbulence	Medium Turbulence	Heavy Turbulence
ω_d	1.6	1.4-1.7	1.5-1.8	1.4-2.0
ζ	1	>0.8	>0.5	>0.5

	Result	Req.
GM	5.5	6
PM	90	45

Figure 25. *Detail Status.*

The detail status display is illustrated in Figure 25. This corresponds to a scenario in which the AVO observes red items in area IV and V, and further investigates before deciding upon a course of action. Both Yaw and Roll are red boxed because at least one metric failed to meet the requirements. In the yaw axis, one observes that the damping frequency lies outside of the acceptable range for heavy turbulence and the damping ratio is inadequate for anything but light turbulence. In the roll axis, there is insufficient gain margin. Given this information, the operator can make an informed decision.

A few scenarios are presented for reference. Figure 26 displays a nominal scenario where no restrictions are implied.

Figure 27 shows a scenario in which it is inadvisable to operate in the presence of medium or heavy turbulence. Figure 28 shows a scenario in which the AV is no longer airworthy and should be terminated. Figure 29 illustrates a scenario where FQ-BIT returns an invalid result.

Figure 26. *Nominal.*
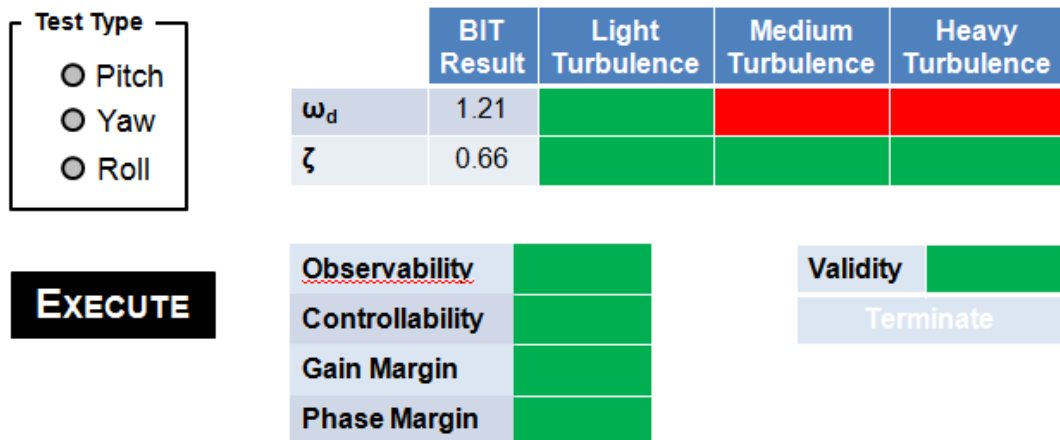
Figure 27. *Restricted.*

Test Type

- ○ Pitch
- ○ Yaw
- ○ Roll

EXECUTE

	BIT Result	Light Turbulence	Medium Turbulence	Heavy Turbulence
ω_d	1.21	🟩	🟥	🟥
ζ	0.66	🟩	🟩	🟩

Observability	🟩
Controllability	🟩
Gain Margin	🟥
Phase Margin	🟩

Validity	🟩
Terminate	

Figure 28. *Terminate.*

Test Type

- ○ Pitch
- ○ Yaw
- ○ Roll

EXECUTE

	BIT Result	Light Turbulence	Medium Turbulence	Heavy Turbulence
ω_d	1.21	✕	✕	✕
ζ	0.66	✕	✕	✕

Observability	✕
Controllability	✕
Gain Margin	✕
Phase Margin	✕

Validity	🟥
Terminate	

Figure 29. *Invalid.*

IV. Test Suites

A. Air Vehicle Plant (Characterization)

The air vehicle plant used for evaluating the candidates is split into two cases: linear and non-linear. In both cases, the longitudinal and lateral dynamics are assumed to be uncoupled. The plants chosen represented an air vehicle comparable in size to a large UAS; the flight conditions are selected to be at the heart of the envelope. In the linear case, a Cessna 172 at an altitude of 5000ft and airspeed of 120kts is used. For the nonlinear case, an F-16 at an altitude of 20000ft and airspeed of 400kts is modeled. The six degrees-of-freedom nonlinear model of the F-16 is based on the information in *Aircraft Control and Simulation*[35] and improved by the Software Enabled Control group at the University of Minnesota[34].

The linear model of the Cessna 172 is presented in the appendix section B. For the longitudinal case, four states are simulated [$VTAS,\alpha,\theta,q$]. Note that only two longitudinal states are estimated by the SID algorithm, so process noise is intentionally introduced to assess the supposition that Phugoid mode can be neglected. The lateral model simulates five states [β,ϕ,p,ψ,r], all of which are used by the SID algorithm.

B. Sensor (Characterization)

Sensor error corrupts the input to the SID algorithm and may affect the accuracy of its solution; hence, these errors must be modeled in order to accurately evaluate the candidates. Several types of errors are displayed in Figure 30. Noise, not graphed, corrupts the truth value with undesirable random perturbations. Bias error indicates a constant offset from the truth value; for example, the truth is $y=x$ while the sensor readout is $y+\varepsilon=x+2$. A scaling error is an error in
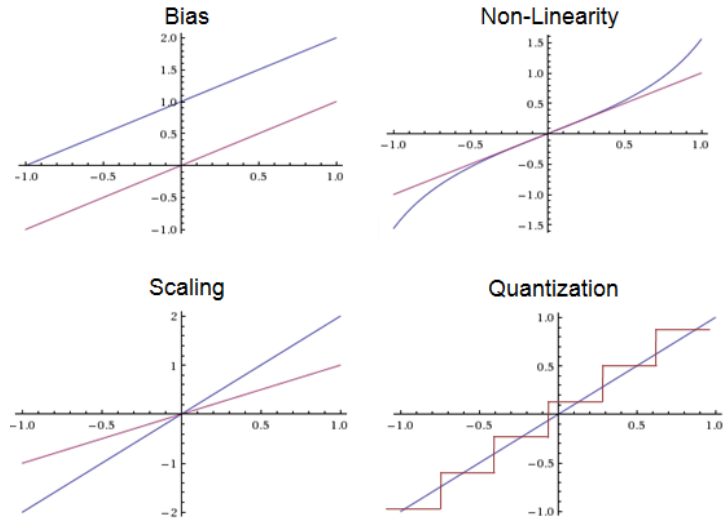
the slope; for example, the truth is $y=3x$ while the sensor readout is $y+\varepsilon=2x$. Nonlinearity error indicates that while the sensor readout is approximately equal to the truth over a range of values, the error nonlinearly expands outside said range; for example, the truth is $y=tan(x)$ while the

Figure 30. *Sensor error types.*

sensor readout is $y+\varepsilon=x$. Quantization error occurs during an analogue-to-digital conversion where the resolution is not high enough to fully describe the analogue function.

Scaling errors often happen when the output units (e.g. volts) are not correctly translated to engineering units (e.g. ft). This error is not inherent to any sensor and is likely to be discovered and rectified before the air vehicle is put into service. Therefore, scaling error is ignored for the purpose of these evaluations. The same argument goes for nonlinearity, and thus is left out of the analysis. With the advanced electronics, it is safe to assume that the sampling bins are fine enough to make quantization an inconsequential issue. Bias and noise are the two consequential components that are modeled for evaluation.

The states required by the SID algorithm can be gathered by two subsystems. $[VTAS,\alpha,\beta]$ can be gathered by the air data system (ADS), and $[\theta,q,\phi,q,\psi,r]$ can be gathered through the inertial navigation unit (INS), specifically the rate sensors. While $[\alpha,\beta]$ can be estimated using inertial navigation outputs, the results are only valid for steady-state flight in the case of α and zero-wind

in the case of β. As neither assumption is appropriate for FQ-BIT, the errors from both INS and ADS are investigated.

1. *INS*

In addition to the aforementioned errors, mechanical misalignment, stress deformation, and temperature are contributors to sensor errors. Mechanical misalignment errors stem from the accelerometers / rate-gyros not being perfectly orthogonal to each other. This error is disregarded because it is largely constant and can be filtered out by sensor-fusion algorithms (namely Kalman Filters) internal to the INS. Load factor from high-G maneuvers can cause deformation of the sensor which affects the readings. Since the FQ-BIT is performed in steady-state level flight, this error source is disregarded. Temperature fluctuations are disregarded with the assumption that environmental conditions will not exceed design limits.

INS determines angles by integrating the rotation rate over time, and sensor fusion algorithms mesh the integration results (which have unbounded errors) with other sensor readings (such as GPS) to keep the error within the acceptable bounds. Ignoring the sensor fusion part, the noise of the integration output can be specified in three formats: function of frequency using power spectral density (PSD) or Fast Fourier Transform (FFT), and function of time using angle random walk (ARW). ARW is usually used in specification of INS because it represents the root-
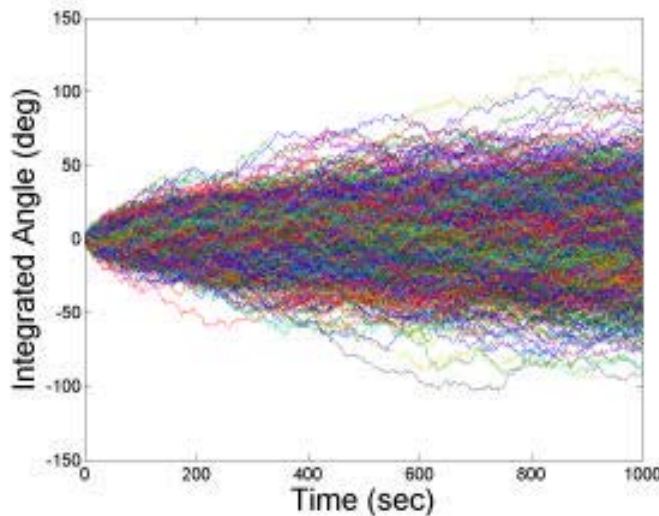
Figure 31. *Integrated angle noise. [ref Stockwell]*

mean-squares error. ARW is a noise specification is in units of angle per square-root of time that directly the average deviation that occurs with integration. Because the integration process involves time, ARW is also a function of time. The statistical distribution, described by the standard distribution, of noise is linearly scaled with the noise level (deg) and with the square root of time ($\sqrt{hr}$). ARW is assumed to be a Gaussian white noise process, and hence, has a normal distribution[13]. An integration of ARW is shown in Figure 31.

Bias results from error in the sensor fusion process (e.g. Kalman filter). Referencing the Kalman filter architecture in Figure 18, the state propagation portion represents the integration of angular rates to obtain angles. The measurement comes from another sensor such as a GPS. The measurement update process fuses the inertial and GPS measurements together. Being a statistical process, the error is never driven to zero as
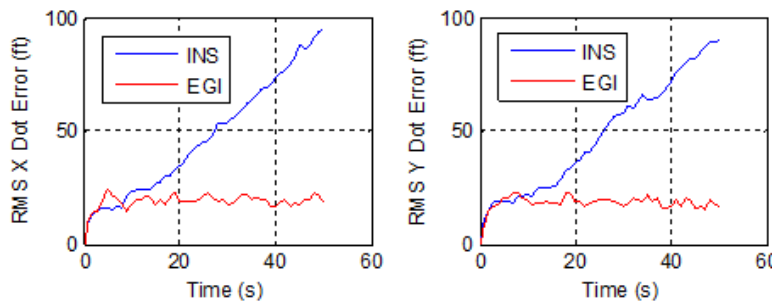
Figure 32. Sensor fusion.

nothing is absolutely certain, which results in a constant offset. Figure 32 compares the unbounded error of the pure integration against the bounded bias error of the embedded GPS/INS (EGI).

To characterize noise and bias, manufacturer spec-sheets[26] are consulted. Noise is assumed to a Gaussian white noise process with a 1-σ value of 0.15 deg/$\sqrt{hr}$ [24]. The bias error comes from the integration of noise and is provided in units of deg/hr. Fibre optic gyros have shown to display about 0.025 deg/hr of bias[26].

Figure 33. *Vane sensor.*

2. *ADS*

Angle of attack can be measured by the ADS through two methods: vane (illustrated in Figure 33) or pitot-static. The vane type sensor works by aligning the vane with the airflow and the difference between the vane and tare angle is the difference between the air-path axis and body axis respectively. This is the angle-of-attack. The pitot-static based sensor has two ports angled differently; the difference between the pressure-readings can be used to solve for the angle-of-attack. Due to the scarcity of open-sure literatures for error of pitot-static alpha sensors, the type is disregarded for the purposes of this paper.

ADS data is often processed before being used by the flight controls program. One such algorithm, a first-order complementary filter, is illustrated and simulated with the bias of 0.125 deg and noise of 0.05 deg (1-σ) in Figure 34 to demonstrate the functionality. Note the ability of the filter to suppress the noise but not the bias error. To remain conservative, no filtering is assumed in the SID algorithm selection process. Literature quotes the performance of a vane type sensor as having a total error less than ±0.25 deg[31]. Due to a shortage of information as most companies regard this information as proprietary, the bias is arbitrarily set at ±0.1 deg and noise at a 1-σ value of 0.05 deg with Gaussian white noise properties.
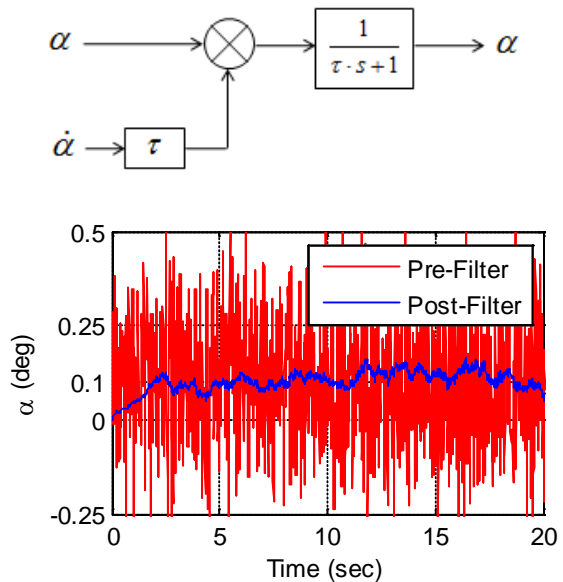
Figure 34. *Complementary Filter (τ=2)*

44

The bias error and noise properties as described above are synthesized for use in the SID algorithm evaluations. Three sensor test cases are used: noise only, bias only, and noise with bias. Each case allows the evaluation of the SID algorithm with each type of sensor error; the noise with bias case presents the worst-case scenario as a "stress-test" of the algorithm.

C. **Input Waveform (Evaluation)**

1. *Candidates*

Besides the doublet, three additional waveforms were considered: 3-2-1-1, pulse, and natural turbulence. NASA had investigated other waveforms such as the sinusoid for stability derivative (parameter) estimation, but found them wanting[32]. The lack of clearly defined edges leads to a subpar estimate. In addition, system identification makes the assumption that flight conditions (such as Mach number and altitude) remain constant; while other input waveforms allow the air vehicle to remained trimmed, sine waves (especially long sweeping inputs) tend to vary the flight conditions, further degrading their results. In Figure 35, note the poor estimate as well as large error bounds exhibited by the sine wave inputs.
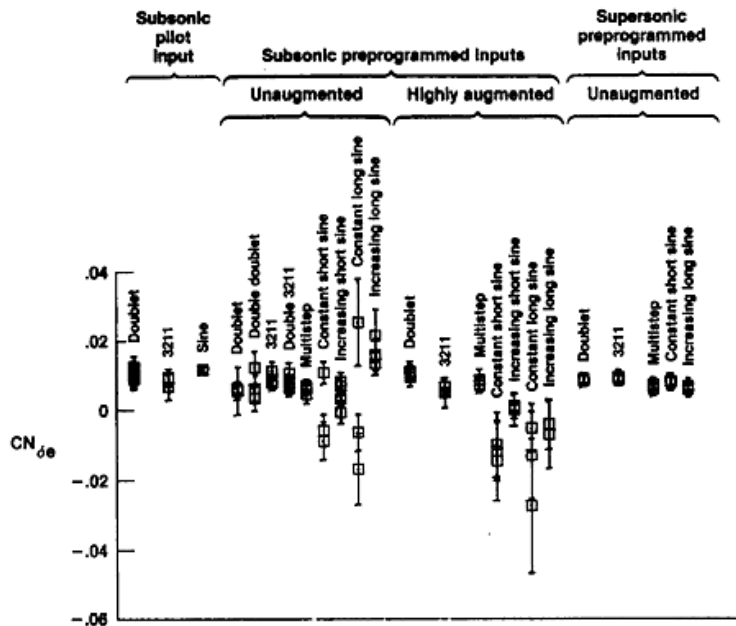

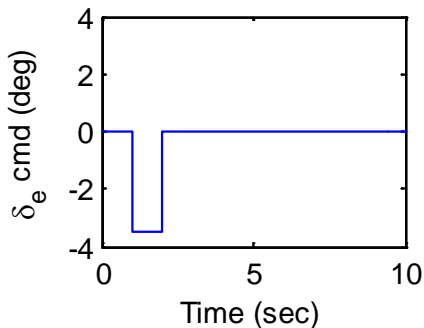**Figure 35.** *Inputs compared. Chart reproduced from NASA report[32].*

Since the FQ-BIT is aimed at serving large UAS that are likely to be subsonic and possess highly augmented (tight) control systems, the sinusoidal input is discounted and 3-2-1-1, doublet, pulse, and turbulence inputs investigated. For a description of the doublet input, refer to the system description section, subsection A.
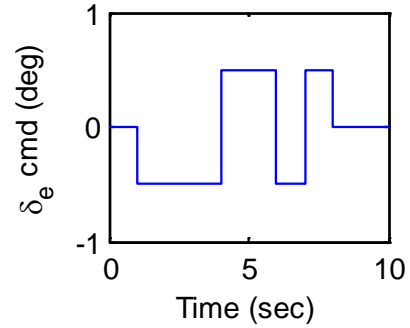
Figure 36. *3-2-1-1.*

A 3-2-1-1 waveform is illustrated in Figure 36. The maneuver is so named because it provides input in time duration of three, two, one, and one, with the direction of the input reversing at every new time bin. Input energy is spread out (lower amplitude) because of the long duration. On the positive, the low amplitude maintains the linearity assumption. Additionally, the different time duration acts like a pseudo-frequency sweep, which can benefit frequency-domain based SID algorithms. The drawback is that the low amplitude input yields less accurate results due to lower SNR and corner-rounding from the inner-loop autopilot. The long duration may also violate the assumption that flight conditions stay the same. Because of these issues, the 3-2-1-1 waveform is rejected.

A pulse waveform displayed in Figure 37 has a high magnitude that gives good signal-to-noise ratio. The input is designed to mimic the Dirac delta function, the length of the pulse is set to account for actuator response time. The short duration of the pulse means it is less likely to induce oscillations. However, the high magnitude input may violate the linearity assumption of SID algorithms. A series of tests are conducted to further characterize the pulse waveform.

Figure 37. *Pulse.*

A natural source of input is turbulence. The inner-loop controller issues control surface commands to maintain the desired flight path in the presence of turbulence. This creates a correlation between the control surface activity and the response of the air vehicle which an SID algorithm can exploit. With the minimal disturbance to flight path, turbulence has the benefit of being easily adaptable for a continuous FQ-BIT implementation. However, the frequency and magnitude of the turbulence encountered are uncertain so that it may be unreliable. In addition, the turbulence input is not exactly known or repeatable (in contrast to a pulse or doublet), which introduces process noise and can degrade the solution.

Turbulence is simulated by the Dryden discrete turbulence model as published in MILSPEC 8785, which is a stochastic process. Given the altitude and wingspan or the vehicle, wind velocities are generated along all three axes. The wind speed at ground level of 10ft/s is defined
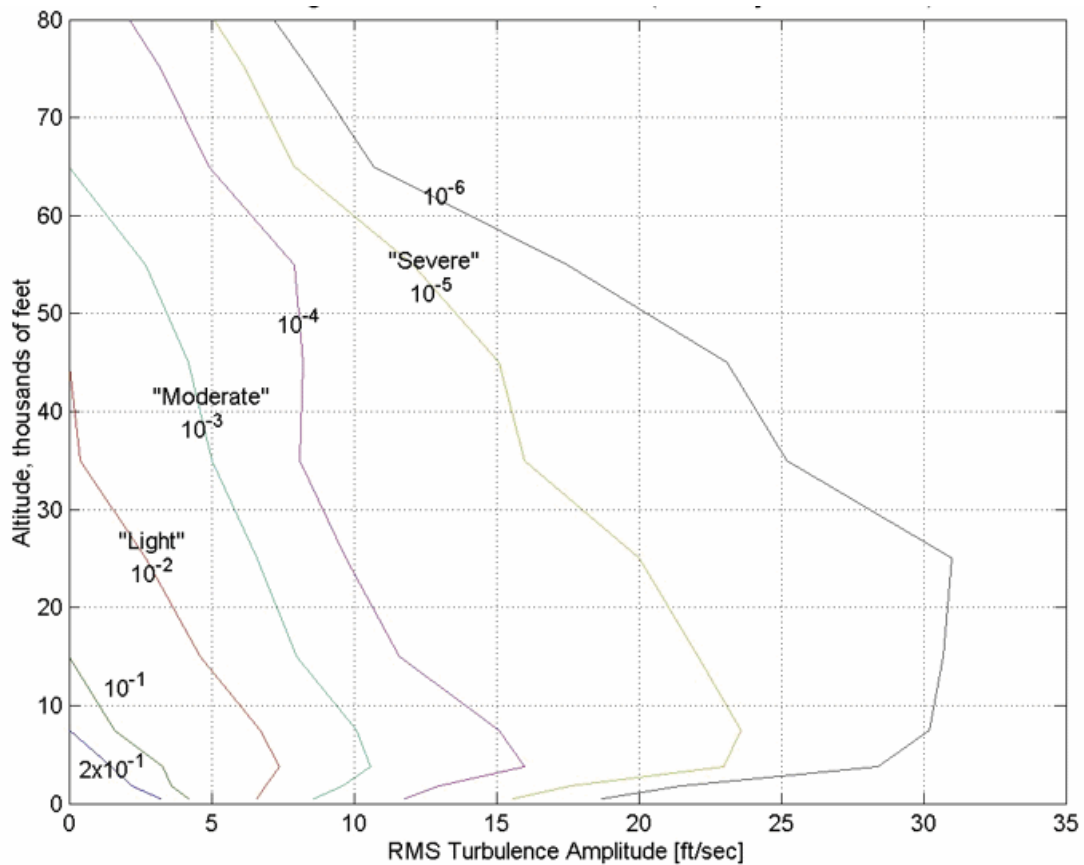

Figure 38. *Turbulence intensity.*

to set the moderate wind magnitude for the model. A "medium severity" probability of exceedance is selected. The turbulence intensity is determined from a lookup table (graphed in Figure 38) as a function of altitude, and the probability of exceedance gives the probability that the turbulence intensity exceeds the table lookup value. The turbulence acts as an input to which the autopilot of the air vehicle responds with control surface commands.

2. *Evaluation Methods*

To choose between the waveforms, the estimation results must be compared against the truth data using a metric for accuracy. The norm of $\hat{A}, \hat{B}$ can be compared against the norm of *A, B*. However, there are dizzying variety of norms available (max-norm, Frobenius-norm, Schatten-norm, etc) with no clear directions on which is best. In addition, norm is only a scalar value and provides no information on convergence or trends. In contrast, by simulating the states using $\hat{A}, \hat{B}$ and input *u* to get $\hat{y}$, a frame-by-frame comparison can be made against the recorded response (*y*). All candidate SID algorithms are used to process $\hat{y}$ in order to determine if any correlation exists between SID algorithm and input waveform. The sensors are assumed to be perfect (no noise, bias, etc) for the purpose of this test. The plant used is the longitudinal linear plant because it is the easiest estimation problem. Because EUDKF is a stochastic process, fifty Monte Carlo trials are run and the ensemble of data at each time step are processed by the root-mean-squares (RMS) error equation:

$$\sqrt{\sum_{N=1}^{k} \frac{1}{k}(\hat{y}_N - y_N)^2}$$

The results are evaluated based on transient behaviour, convergence rate, and steady-state error.

To test the effectiveness of atmospheric turbulence as an input waveform, a 15sec long simulation is built in SIMULINK for testing. The block "Dryden Wind Turbulence Model" is

used to generate the wind gusts with horizontal, vertical, and lateral components. The Dryden model generates gusts that smoothly ramp from zero velocity to the gust magnitude, stay at that magnitude for a period of them, then ramp back to zero. The time duration of the ramps and the interval at maximum magnitude, as well as the gust magnitude itself, are stochastic processes that follow the models and equations outlined in MILSPEC 8785. For the purposes of this evaluation, the air vehicle altitude is considered to be invariant at 5000ft; in addition, the roll and yaw angles are assumed to be invariant at zero as well. The aircraft plant is the linear time-invariant representation of the longitudinal dynamics of a Cessna 172 at an altitude of 5000ft and airspeed of 120kts. Because only the longitudinal dynamics are simulated, the lateral wind gust is not used. The plant incorporates the horizontal turbulence by adding it to the airspeed state. The vertical turbulence is incorporated by calculating the induced angle-of-attack, and then adding it to the corresponding state. An altitude hold autopilot is constructed with altitude rate feedback. The airspeed is multiplied by the pitch angle (small angle approximation is assumed to remove the sine function on θ) to represent altitude rate. A proportional gain feedback is used to generate the elevator command. The block diagram of this simulation setup is displayed in Figure 39. The power spectral density of the horizontal and vertical wind gusts from the Dryden model is
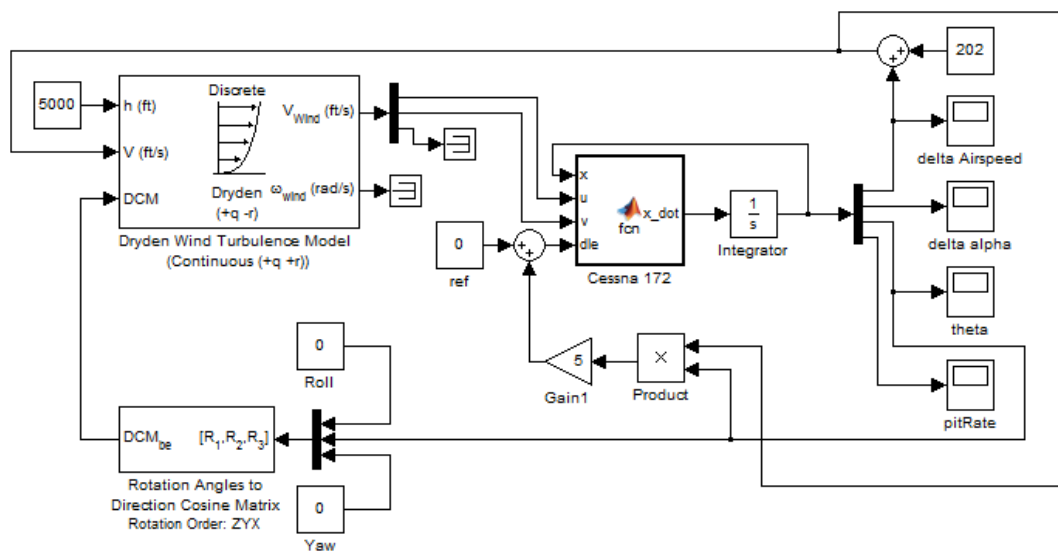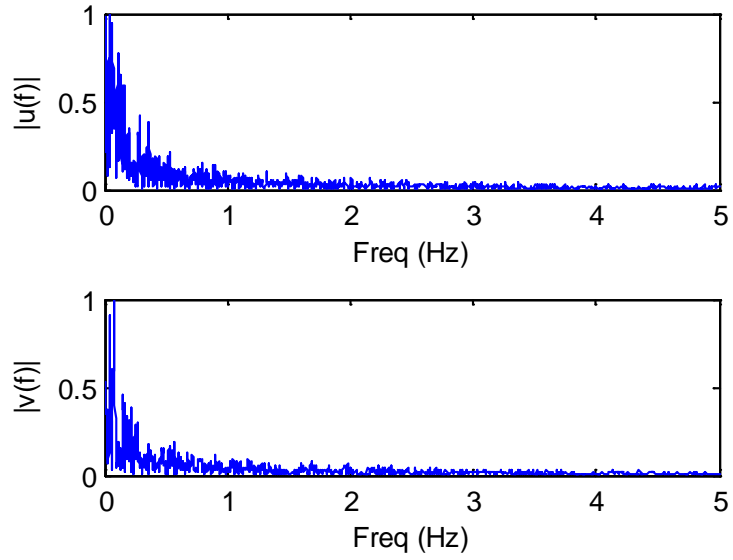

Figure 39. *Turbulence SIMULINK model setup.*

displayed in Figure 40. Note that turbulence is mostly concentrated at low frequencies.

Figure 40. *Turbulence power-signal-density.*

The resulting states and elevator movements from the turbulence are recorded, processed by the SID algorithm, and the estimated states generated. The RMS equation is again applied to evaluate the usefulness of turbulence as an input.

D. **SID Algorithm (Evaluation)**

In addition to Extended UD Factorized Kalman Filter (EUDFK), two other methods were considered: Equation Error (Time Domain) and Equation Error (Frequency Domain) methods. Output error method is not considered due to the long runtime (fifteen to thirty minutes) and occasional instability observed by the author in industry use. For a description of EUDKF, the reader is referred to the system description section, subsection A.

Unlike EUDKF, equation error is a deterministic batch filter which operates on all the measurements; it is also a "one-step" method that performs all the computation once whereas EUDKF performs the same steps recursively for each data measurement. Equation error was one of the first analytical techniques used to estimate aircraft dynamic model parameters from flight data; modern updates to the method had negated much of earlier problems[22]. In essence, equation error is the linear least squares problem in matrix form. It assumes the system is linear time-

invariant, the measurements are known without error, and that noise is Gaussian white noise. The method utilizes relatively elementary and undemanding mathematical operations. Being a batch filter however, it processes all the measurements at once, and the large number of measurements (large matrix sizes) makes matrix multiplications expensive propositions. In addition, the algorithm cannot process the data until all measurements are gathered. This concentrates all the computational requirements in one short time period, which may affect the other programs being run.

The estimation error is a linear function of the measurement error. If no measurement noise is present, then the estimated value is a deterministic quantity exactly equal to the plant (assuming no process noise). In the presence of noise, the estimated value becomes stochastic and possibly biased. For an unbiased estimation, the residuals have a zero mean; this property can be used to detect sensor bias.

1. *Candidates*

Two variations of the equation error method are considered: one operating in the time domain, the other in the frequency domain.

Equation Error (Time Domain)

The central idea of the least squares method revolves around minimizing the errors expressed in a cost function. Starting with the state space formulation of the problem, the error, cost function, and solution are derived below. The state equation can be written as,

$$\dot{x} = Ax + Bu \tag{12}$$

With the subscript m denoting measurement, the error (ε) can be written as,

$$\varepsilon = \dot{x}_m - \dot{x} = \dot{x}_m - Ax_m - Bu_m \tag{13}$$

If the *A* and *B* matrices along with x_m and u_m vectors are collected together, their augmented forms can be written as,

$$A_a = \begin{bmatrix} A & B \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} & \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \end{bmatrix}$$

$$x_{am} = \begin{bmatrix} x_m \\ u_m \end{bmatrix} = \begin{bmatrix} \alpha \ldots \alpha(t) \\ q \ldots q(t) \\ \delta_e \ldots \delta_e(t) \end{bmatrix}$$

(14)

Then equation 13 can be rewritten,

$$\varepsilon = \dot{x}_m - A_a x_{am}$$

(15)

Using the deterministic least squared method, the cost function to be minimized can be written as,

$$J = \frac{1}{2}\varepsilon^T \varepsilon$$

(16)

To minimize J, the first derivative is taken with respect to x_{am} and set to zero. This results in the estimated augmented matrix.

$$\hat{A}_a = \begin{bmatrix} \hat{A} & \hat{B} \end{bmatrix} = \dot{x}_m \left(x_{am}^T \right) \left(x_{am} \cdot x_{am}^T \right)^{-1}$$

(17)

The residuals are,

$$\varepsilon = \dot{x}_m - \hat{A}_a x_{am}$$

(18)

Equation Error (Freq Domain)

Time domain methods have the advantage that measurements are sampled in discrete time domain form and ready for processing. Frequency domain methods, despite the additional computational requirements for the transformation, can prove advantageous due to the sampling window acting like a band-pass filter so as to focus only on the area(s) of interest. The lower frequency bound is set higher than the sensor drift rate; the high frequency bound is set lower

than the structural resonance to exclude undesirable dynamics. As a bonus, this frequency domain technique is less dependent on sharp input corners than its time domain cousin[29].

The window size is chosen between half of the sampling rate and 1/8Hz. Of all the air vehicle modes both longitudinal and lateral, the short period mode is expected to be the fastest with a nominal period no more than four seconds (or 1/4Hz). With this assumption, the minimum window frequency is set at 1/8Hz for added margins. The maximum window frequency is set by the critical frequency of the Nyquist-Shannon sampling theorem. For example, if the sampling rate is 50Hz, then the highest discernable (and maximum window) frequency is 25Hz.

The number of frequency bins is chosen as the next higher multiple of two of the sample size; e.g. 500 samples would have 512 frequency bins. The number of bins is a multiple of two so as to increase Fast Fourier Transform (FFT) efficiency. In addition, the frequency resolution is refined to be at least 0.04Hz or better; e.g. the frequency resolution of 512 bins is 0.5Hz, then the bins are increased until the resolution is at least 0.04Hz ($512\text{bins} \times 0.5\text{Hz} \div 0.04\text{Hz} = 6400\text{bins}$) which is then raised to the next multiple of two
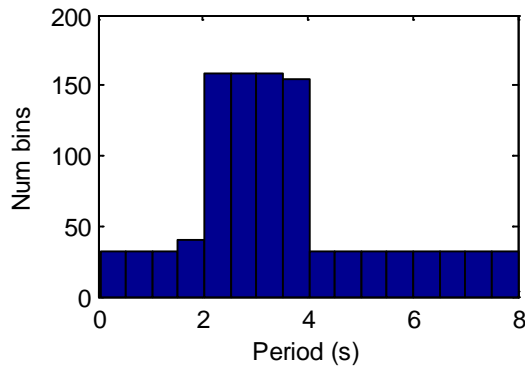
Figure 41. *Histogram of frequency range (displayed in period for clarity).*

(8192bins). The algorithm thus far has spread the bins evenly across the frequency spectrum, but additional bins are desirable in the area of interest. In the longitudinal case, additional bins are added between 1/2Hz and 1/4Hz for enhanced resolution near the expected short period frequency with the resulting histogram displayed in Figure 41 (note that the period is displayed instead of frequency for clarity). The nominal periods of the lateral modes are more difficult to characterize and the frequency bins remain un-augmented.

The only difference between the mechanization of equation error in time domain and frequency domain is that the later is comprised of both real and complex numbers. The derivation process follows the same broad strokes. The discrete Fourier transform can be written as,

$$\tilde{x} \approx \Delta t \cdot \sum_{0}^{N-1} x_i e^{-j\omega t_i} \tag{19}$$

Rewriting the state equation in the frequency domain, the following results,

$$j\omega\,\tilde{x}(\omega) = A\,\tilde{x}(\omega) + B\,\tilde{u}(\omega) \tag{20}$$

Following the same logic as the time domain method, the cost function can be written as,

$$J_k = \frac{1}{2}\sum_{n=1}^{m}\left| j\omega_n\,\tilde{x}_k(n) - A\,\tilde{x}_k(n) - B\,\tilde{x}_k(n)\right|^2 \tag{21}$$

where m is the number of bins used in the Fourier transform, $\tilde{x}_k(n)$ denotes the k-th element of $\tilde{x}$ for ω_n, and A_k and B_k denote the k-th row of the matrix.

By defining the X and Y the following way,

$$
\begin{aligned}
Y &\equiv \begin{bmatrix} j\omega_1\,\tilde{x}_k(1) \\ j\omega_2\,\tilde{x}_k(2) \\ \vdots \\ j\omega_m\,\tilde{x}_k(m) \end{bmatrix} = \begin{bmatrix} j\omega_1\,\tilde{\alpha}(1) & j\omega_1\,\tilde{q}(1) \\ j\omega_2\,\tilde{\alpha}(2) & j\omega_2\,\tilde{q}(2) \\ \vdots & \vdots \\ j\omega_m\,\tilde{\alpha}(m) & j\omega_m\,\tilde{q}(m) \end{bmatrix} \\
X &\equiv \begin{bmatrix} \tilde{x}(1) & \tilde{u}(1) \\ \tilde{x}(2) & \tilde{u}(2) \\ \vdots & \vdots \\ \tilde{x}(m) & \tilde{u}(m) \end{bmatrix} = \begin{bmatrix} \tilde{\alpha}(1) & \tilde{q}(1) & \tilde{\delta}_e(1) \\ \tilde{\alpha}(2) & \tilde{q}(2) & \tilde{\delta}_e(2) \\ \vdots & \vdots & \vdots \\ \tilde{\alpha}(m) & \tilde{q}(m) & \tilde{\delta}_e(m) \end{bmatrix}
\end{aligned}
\tag{22}
$$

the cost function can be rewritten as,

$$J = \frac{1}{2}(Y - X\theta)^*(Y - X\theta) \tag{23}$$

where * denotes the complex conjugate transpose and θ is,

$$\hat{\theta} = \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \\ b_1 & b_2 \end{bmatrix} \tag{24}$$

The minimum of the cost function is met with the following criteria,

$$\hat{\theta} = \left[\mathrm{Re}\left(X^T X\right)\right]^{-1} \mathrm{Re}\left(X^T Y\right) \tag{25}$$

where *Re( )* means the real part of the matrix. Note that equation 25 is the same as equation 18 if only real numbers are concerned.

2. *Evaluation Methods*

In order to evaluate the three SID candidates, each combination of air vehicle plant and sensor must be tested. Two plant types (linear and non-linear), four sensor error types (no error, noise
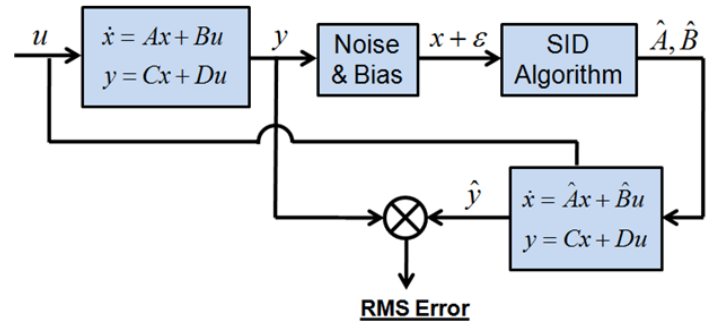
Figure 42. *Evaluation Process.*

corruption, bias offset, noise and bias together), and three candidate SID algorithms (equation error in time domain, equation error in frequency domain, and EUDKF) combine to form twenty-four test conditions for both longitudinal and lateral dynamics. This means forty eight total tests are performed.

The test process outlined in Figure 42 uses the selected input waveform (doublet) to excite the plant (two flavors), and then the signal is injected with sensor noise (four choices), finally the signal is routed through the SID algorithm under test (three selections). The resulting $\hat{A}, \hat{B}$ are used to generate the estimated system response ($\hat{y}$), which is then compared against the true system response *y* using the root-mean-squares (RMS) error metric shown below,

$$\sqrt{\sum_{N=1}^{k} \frac{1}{k}(\hat{y}_N - y_N)^2}$$

In order to more easily interpret the results, they are separated into two categories: linear plant and non-linear plant. In each category, the four sensor-error types are presented separately. Finally, in each sensor-error type, the RMS errors of each state resulting from all three SID algorithms are plotted together so as to give an "apples-to-apples" comparison that allows one to draw conclusions as to whether a SID algorithm is universally superior, under certain circumstances, or otherwise.

V. Test Results

A. **Input Waveform**

<u>**Turbulence**</u>

Fifty Monte Carlo trials were performed to obtain the results in Figure 43 thru Figure 46. **Note that both equation error methods (time domain and frequency domain) diverged and were not plotted.** The red lines in the figures represent the RMS error from EUDKF. The significance of the results is discussed in chapter VI.
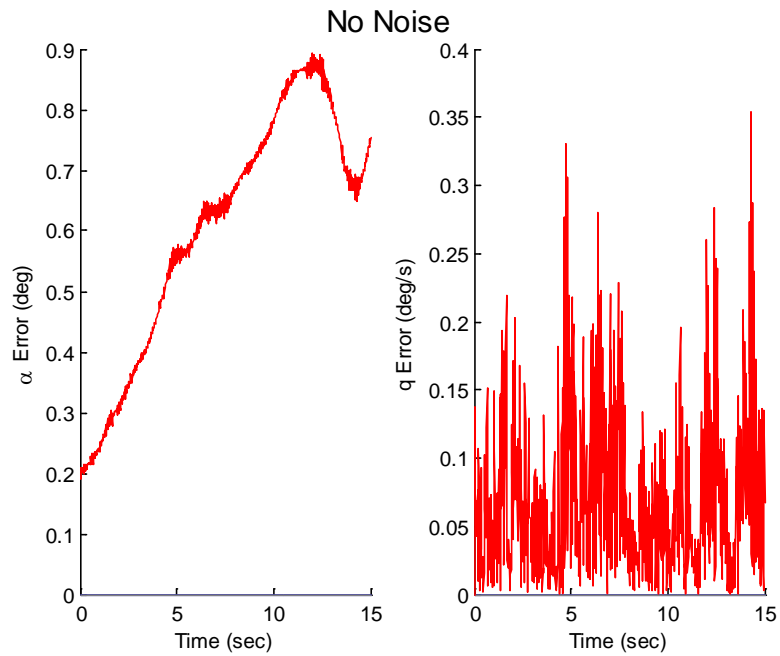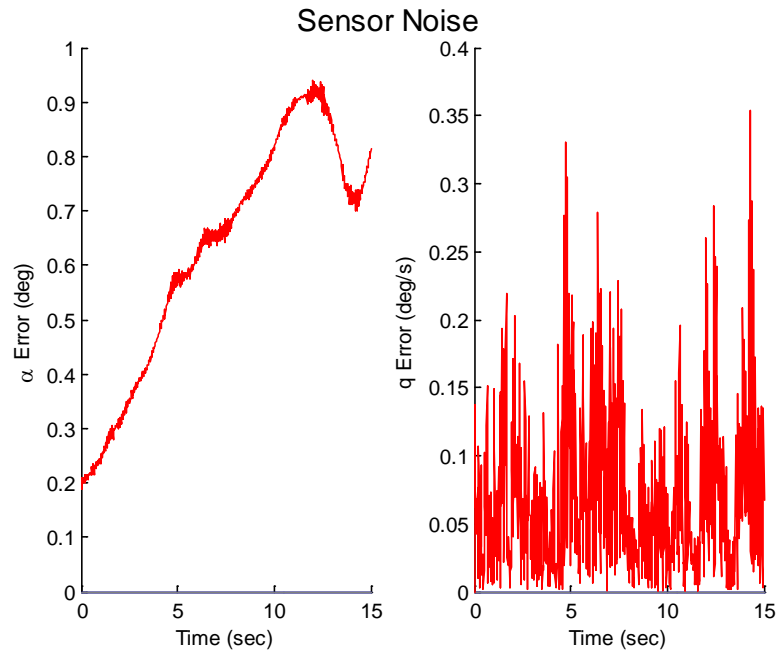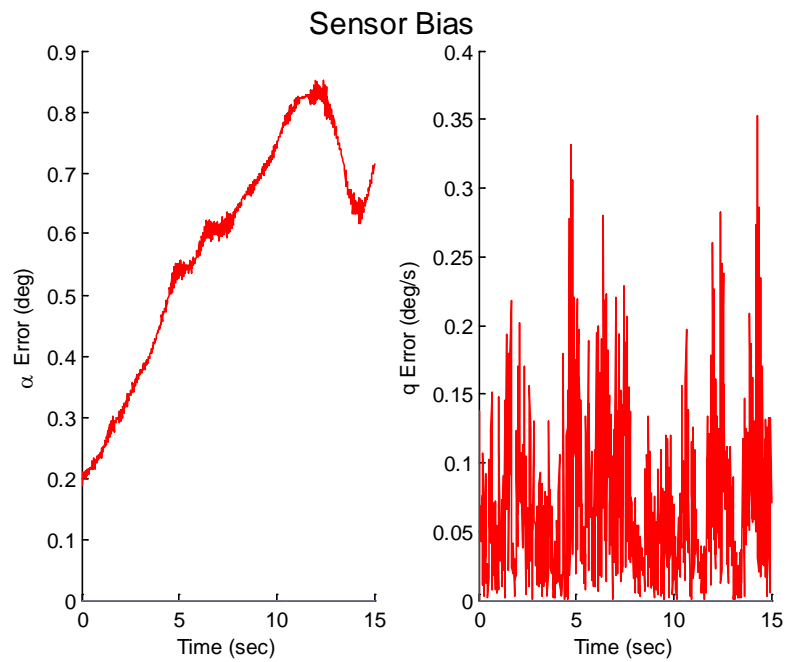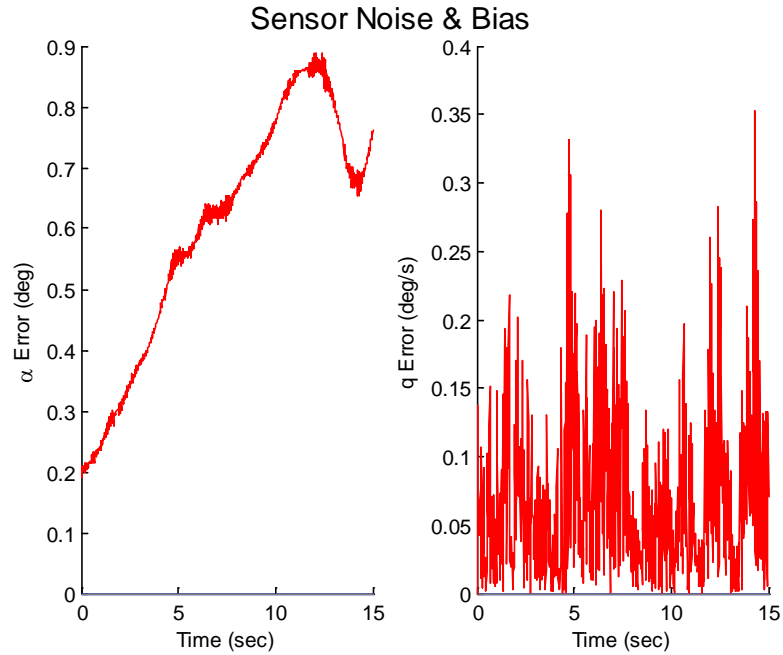

Figure 43. *Turbulence error, no sensor noise, linear Cessna 172 longitudinal plant.*

Figure 44. *Turbulence error, with sensor noise, linear Cessna 172 longitudinal plant.*

Figure 45. *Turbulence error, with sensor bias, linear Cessna 172 longitudinal plant.*

Figure 46. *Turbulence error, with sensor noise and bias, linear Cessna 172 longitudinal plant.*

Doublet vs. Pulse

Fifty Monte Carlo trials were performed using the linear Cessna 172 longitudinal plant. The RMS errors of the pulse input waveforms are shown in Figure 47 while the RMS errors of the doublet input waveform are shown in Figure 48. **Note that in Figure 47, equation error (frequency domain) method diverged and the results were not plotted.** The significance of the results is discussed in chapter VI.
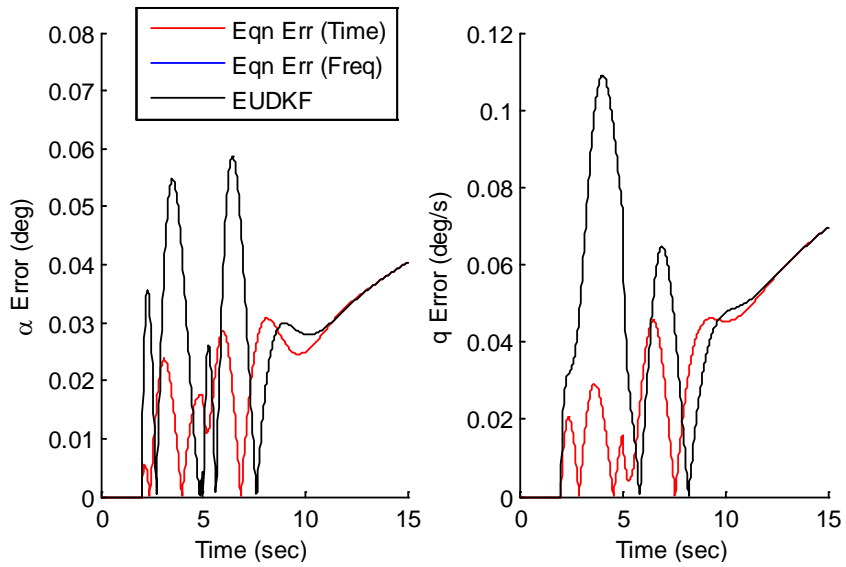
Figure 47. *Pulse error, no sensor noise, linear Cessna 172 longitudinal plant.*
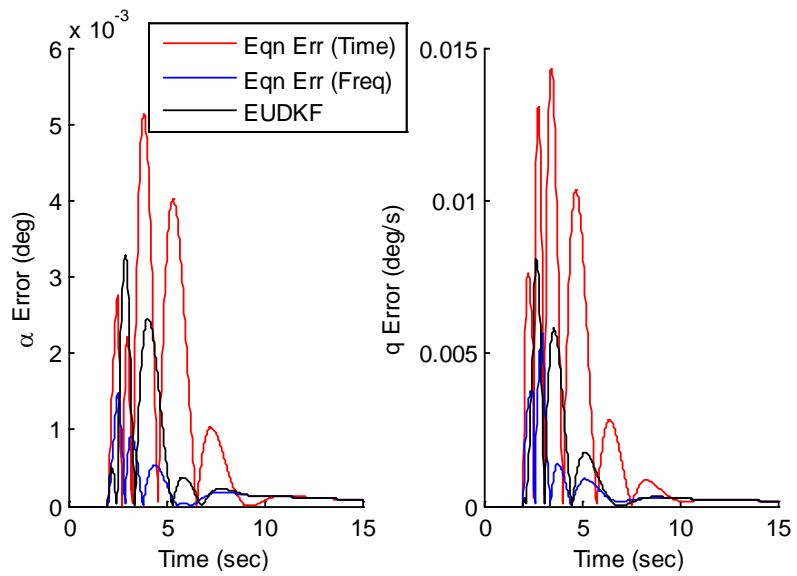

Figure 48. *Pulse error, no sensor noise, linear Cessna 172 longitudinal plant.*

B. **SID Algorithm**

Fifty Monte Carlo trials were performed for each of the following charts. The results with the longitudinal linear plant are presented in Figure 49 thru Figure 52. The results with the longitudinal nonlinear plant are presented in Figure 53 thru Figure 56. The results with the lateral linear plant are presented in Figure 57 thru Figure 60. The results with the lateral nonlinear plant are presented in Figure 61 thru Figure 64. **Note that equation error (frequency domain) method universally diverged and the results were not plotted.** The significance of the results is discussed in chapter VI.

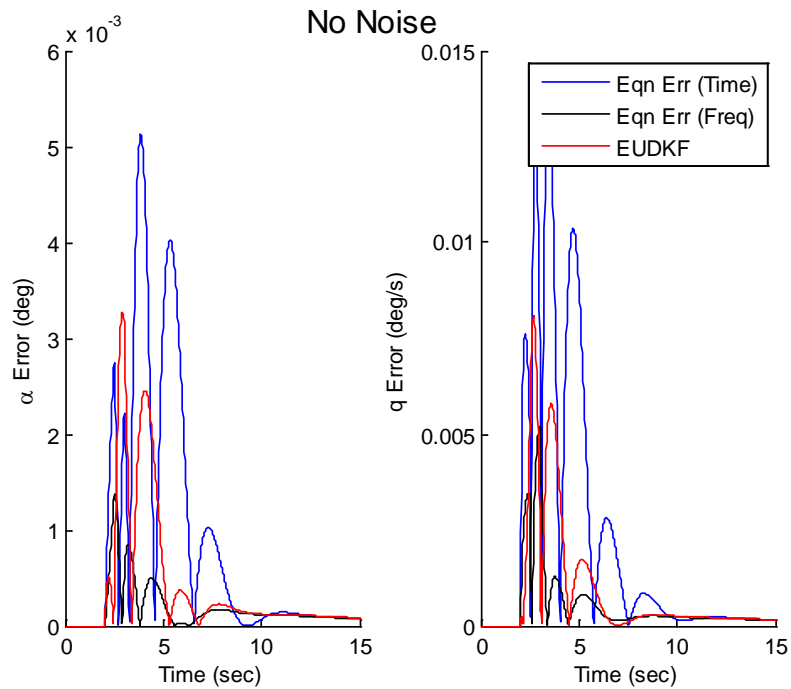**Figure 49.** *SID RMS error, no sensor error, linear longitudinal plant*

Figure 50. *SID RMS error, with sensor noise, linear longitudinal plant*
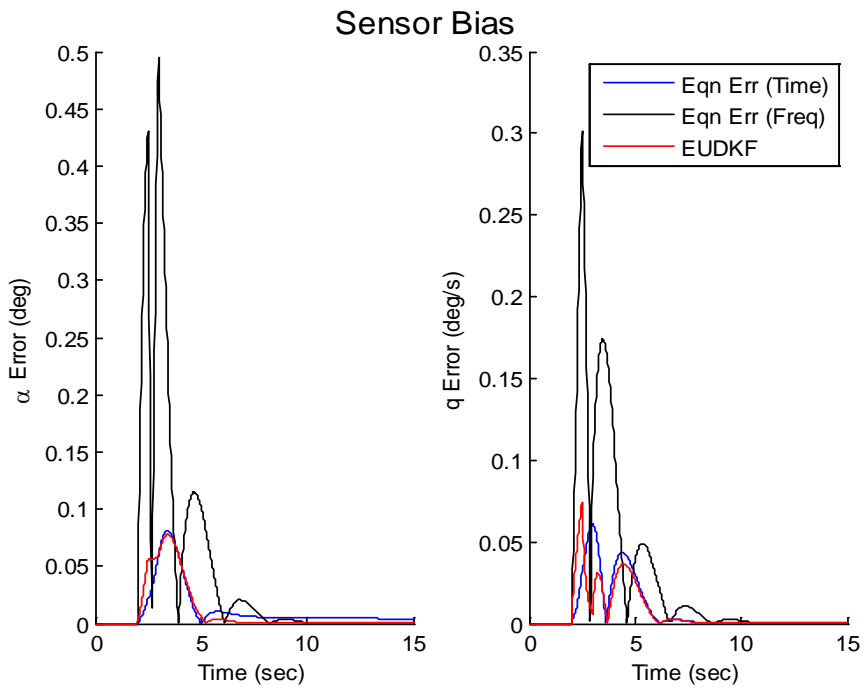
Figure 51. *SID RMS error, with sensor bias, linear longitudinal plant*

Figure 52. *SID RMS error, with sensor noise and bias, linear longitudinal plant*

Figure 53. *SID RMS error, no sensor error, nonlinear longitudinal plant*

Figure 54. *SID RMS error, with sensor noise, nonlinear longitudinal plant*

Figure 55. *SID RMS error, with sensor bias, nonlinear longitudinal plant*

Figure 56. *SID RMS error, with sensor noise and bias, nonlinear longitudinal plant*

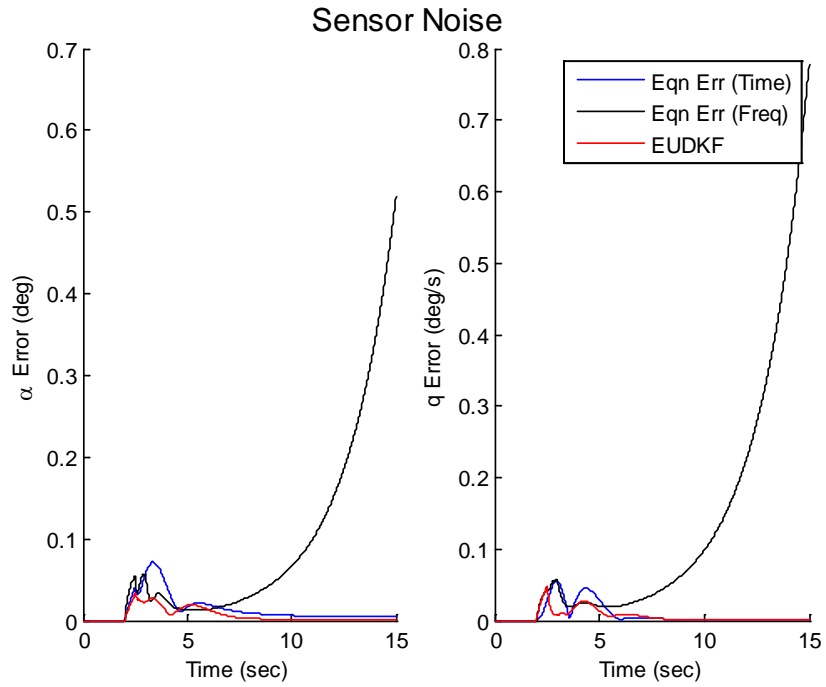**Figure 57.** *SID RMS error, no sensor error, linear lateral plant*

Figure 58. *SID RMS error, with sensor noise, linear lateral plant*

Figure 59. *SID RMS error, with sensor bias, linear lateral plant*
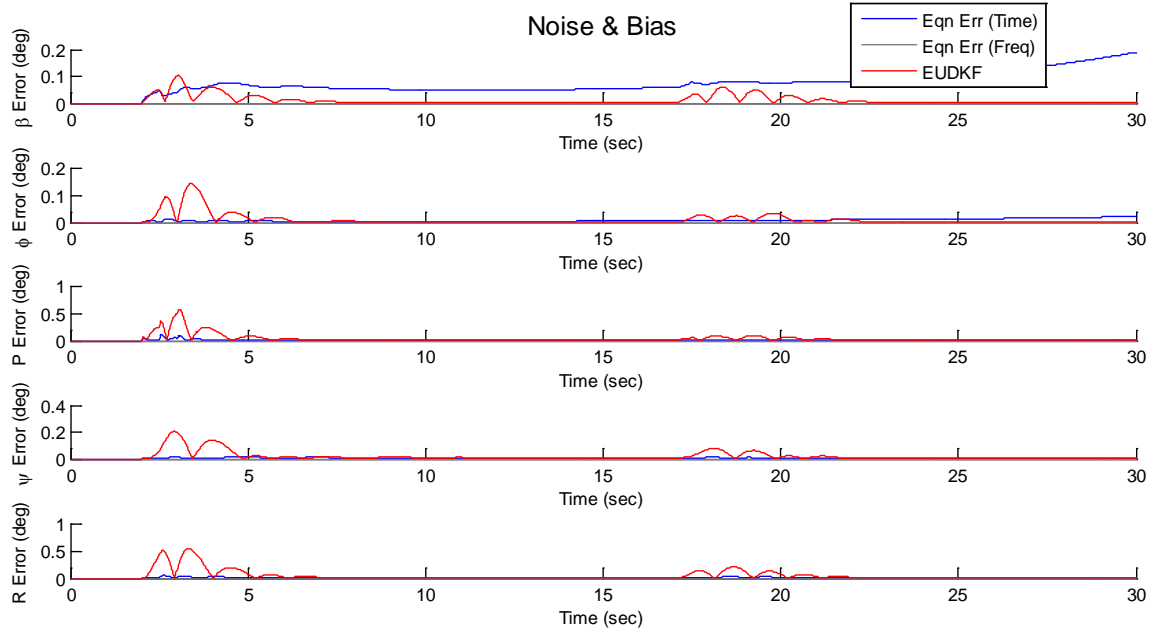
Figure 60. *SID RMS error, with sensor noise and bias, linear lateral plant*
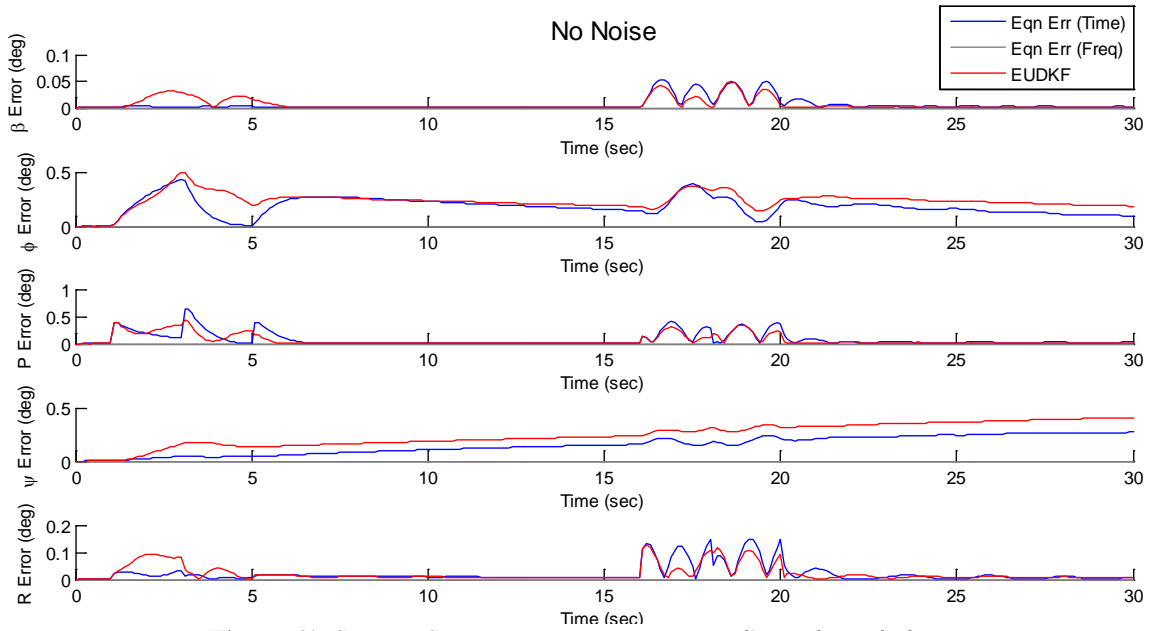
Figure 61. *SID RMS error, no sensor error, nonlinear lateral plant*
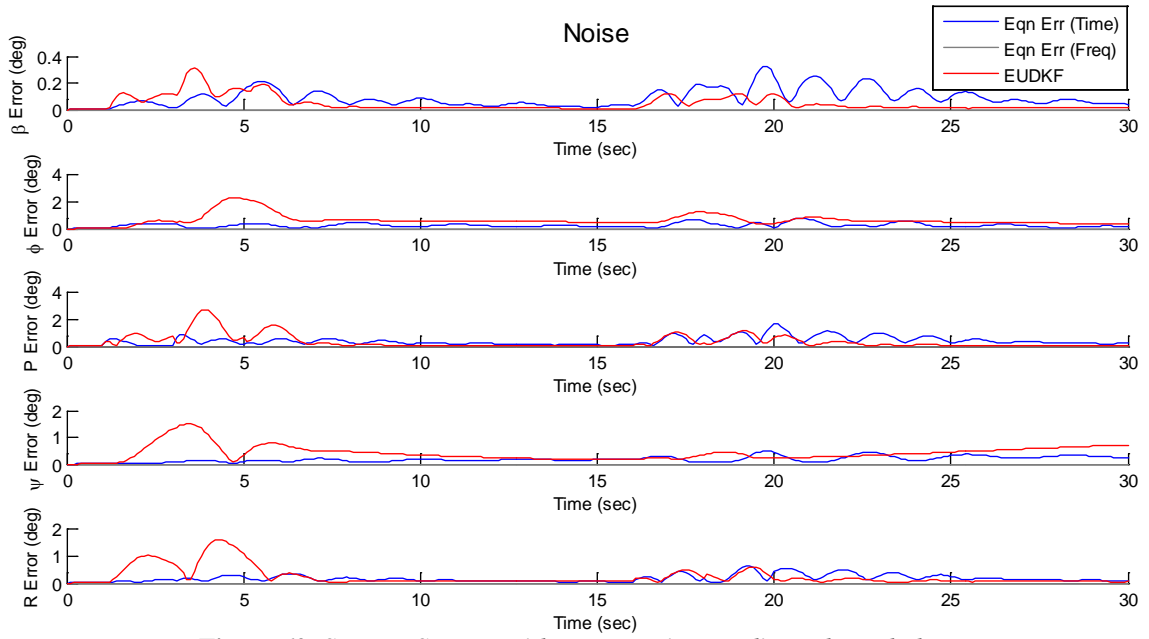
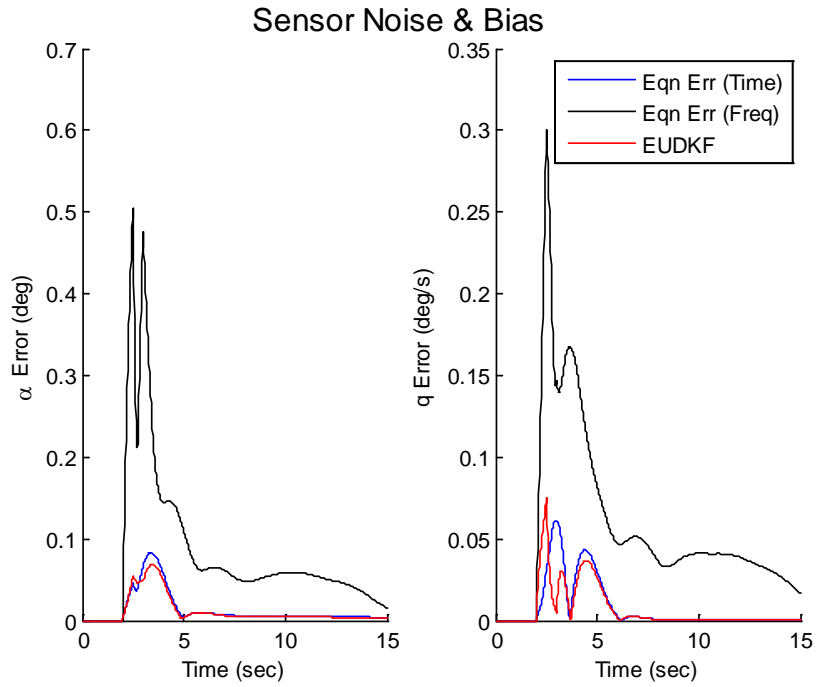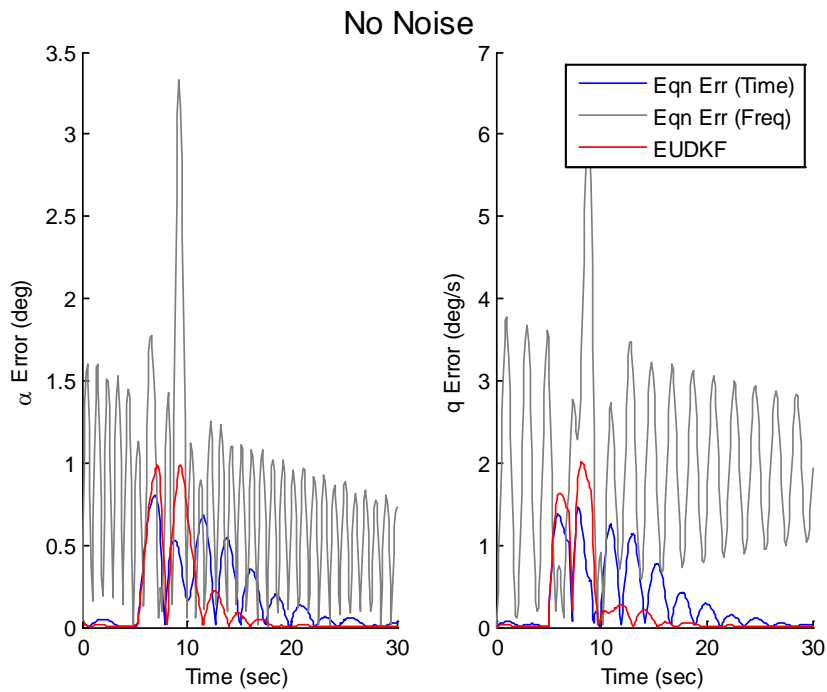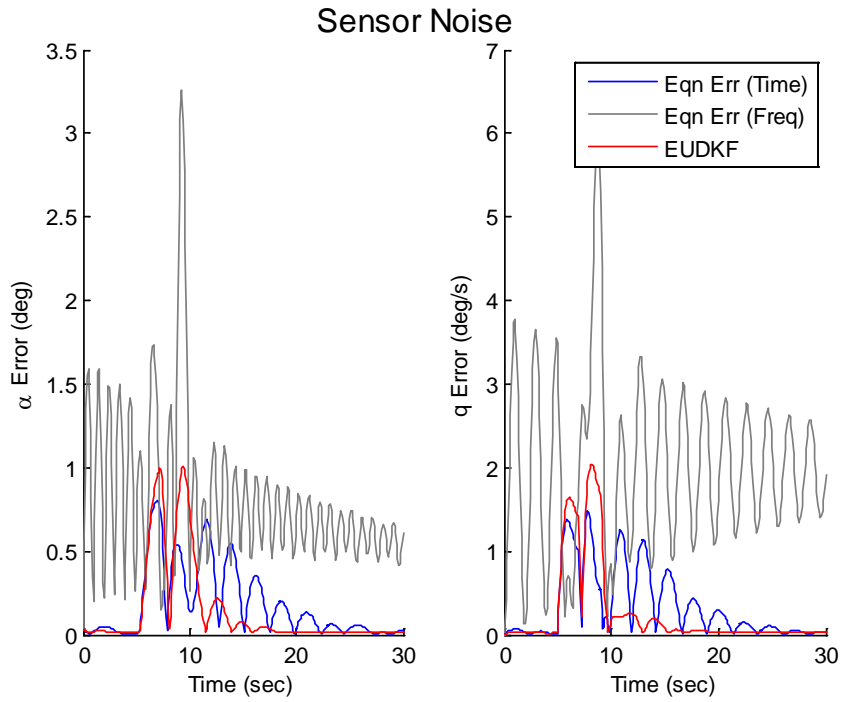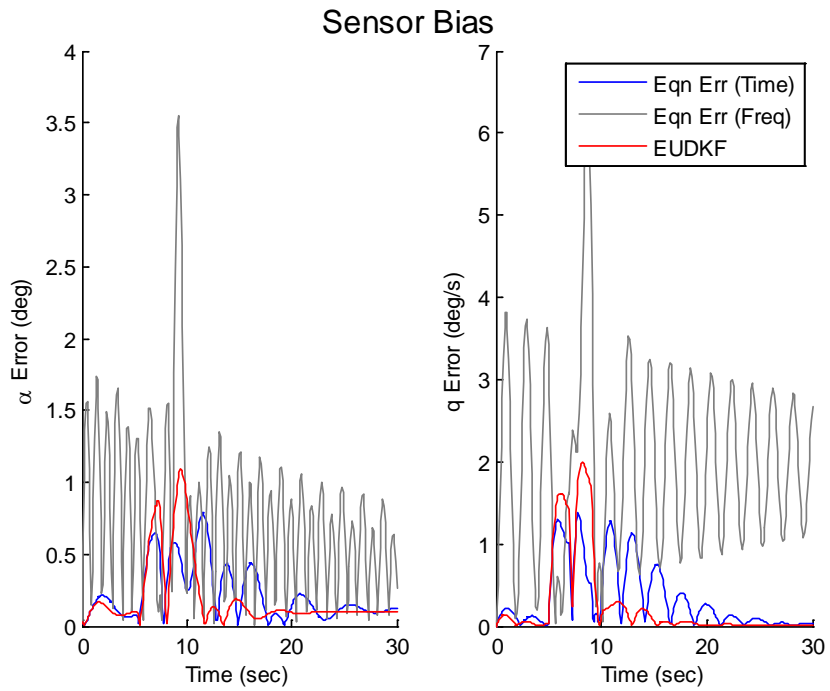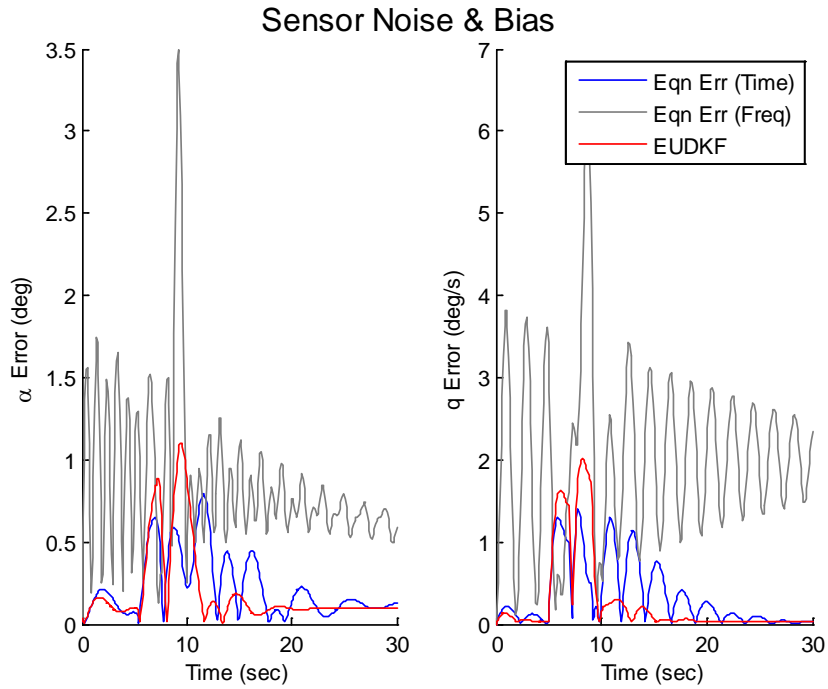**Figure 62.** *SID RMS error, with sensor noise, nonlinear lateral plant*

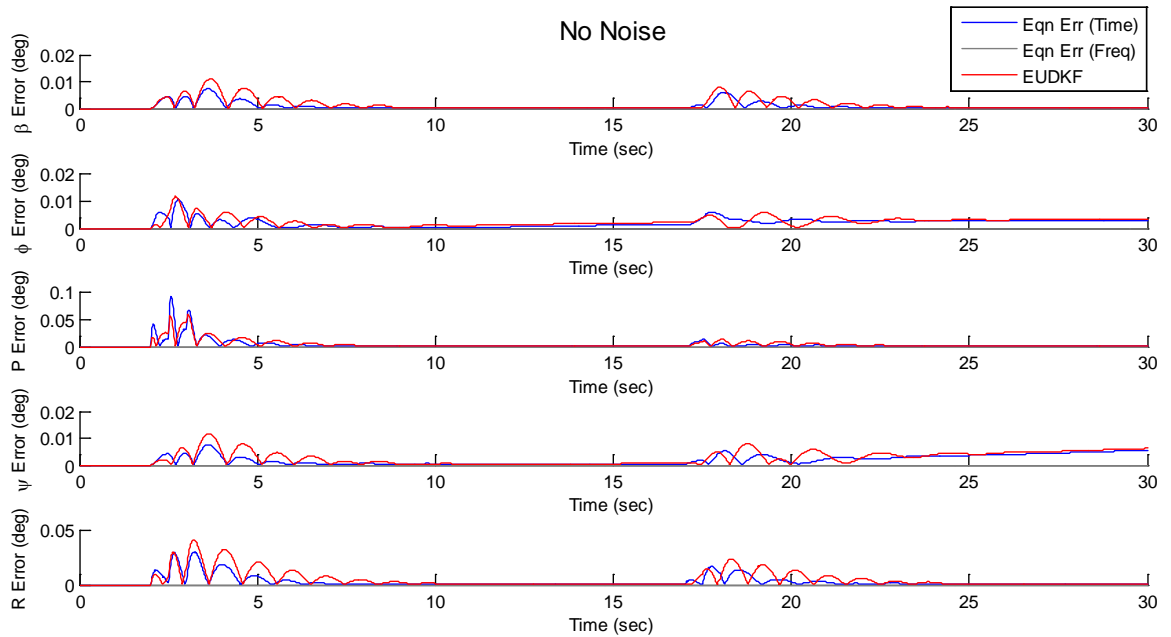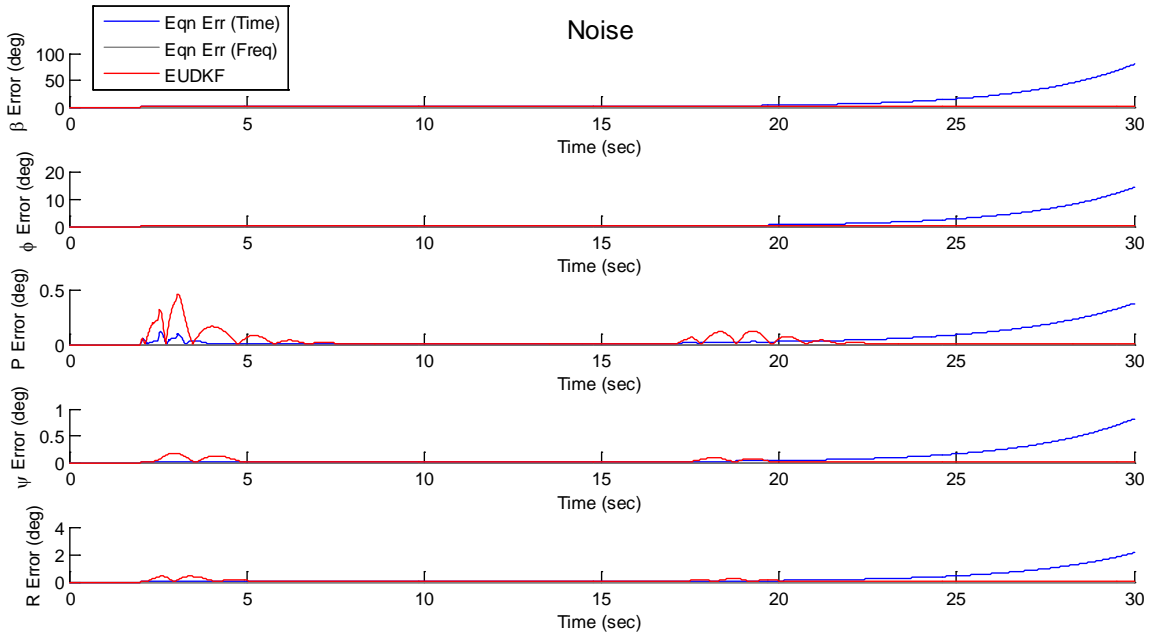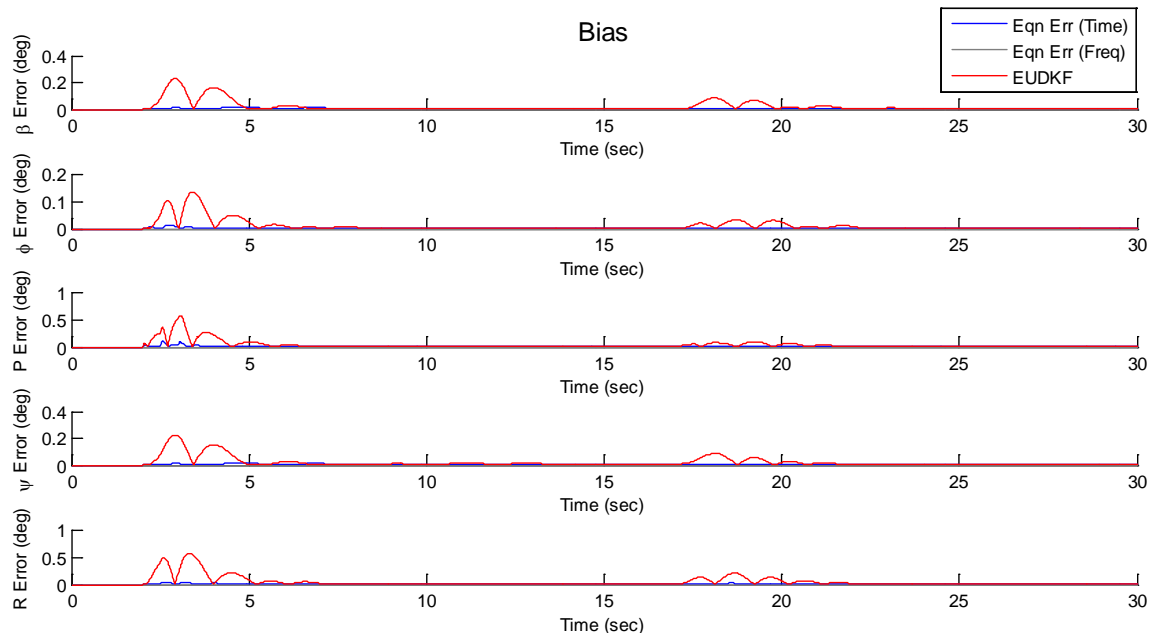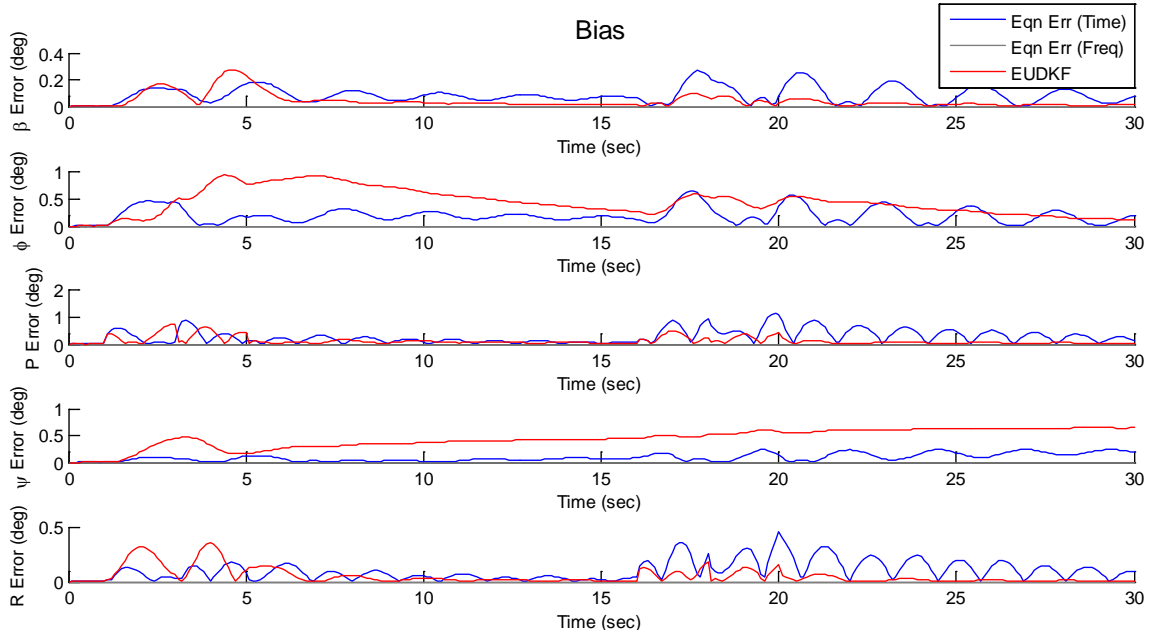**Figure 63.** *SID RMS error, with sensor bias, nonlinear lateral plant*
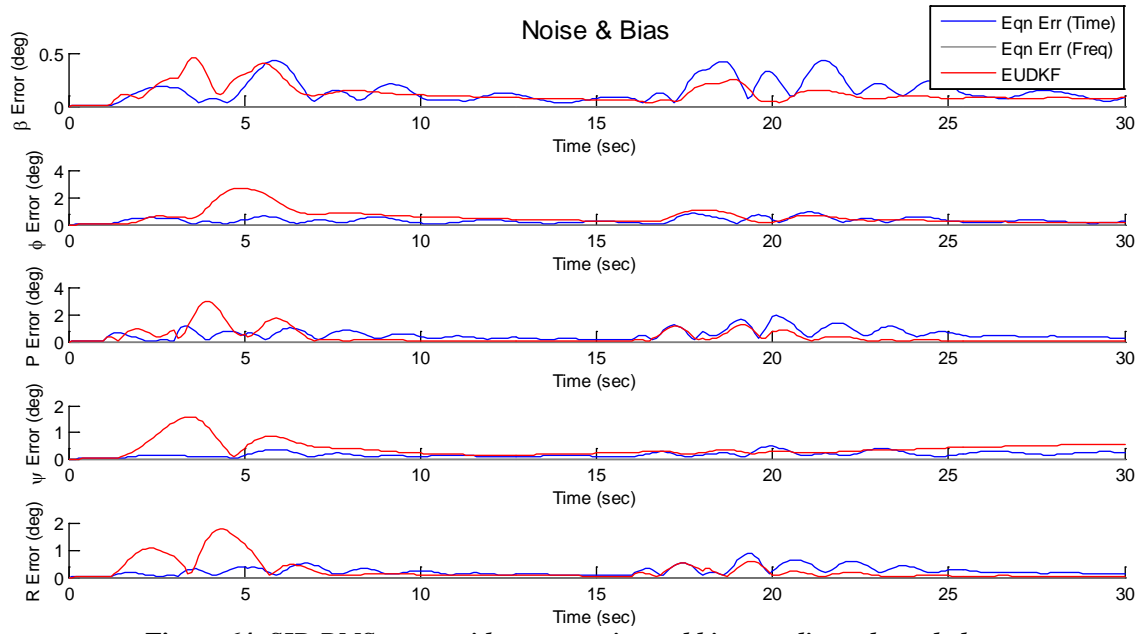
Figure 64. *SID RMS error, with sensor noise and bias, nonlinear lateral plant*

VI. Analysis

A. **Input Waveform**

<u>Turbulence</u>

Both equation error methods diverged quickly and their RMS errors were not plotted. There was not enough energy content in the appropriate frequencies for the equation error methods to generate good estimates. The short, sharp gust portion of turbulence inputs were averaged out by the method due to their Gaussian, zero-mean, statistical properties. The long period gusts were not able to excite the system enough for a good estimate.

Even the mathematically sophisticated Extended UD factorized Kalman Filter was only able to generate a poor estimate. The diverging trends in the RMS error of both states indicated a persistent error in the system estimate. Through the noise in q, it was observable that the error was diverging. This was due to the fact that the input (refer to Figure 40) was not a white noise function. However, one cannot filter a physical phenomenon to improve EUDKF performance. The poor estimate from EUDKF indicated that it was not suitable for use by FQ-BIT.

Regardless of sensor noise, RMS error divergence was observed with all three SID algorithms. Both equation error methods diverged spectacularly. EUDKF was only able to generate a poor estimate that nevertheless showed divergent trends in all RMS errors. The poor performance, lack of regularity, and the general unpredictable nature led to the elimination of turbulence as a valid input waveform.

<u>Doublet vs. Pulse</u>

In examining Figure 47 and Figure 48, it was easily observed that the RMS error behaviors in both α and q showed the same trends. The pulse input showed a diverging trend in error. In

addition, the equation error method in frequency domain diverged and was not plotted while it converged with a doublet input and was plotted in Figure 48. The order of magnitude in RMS error showed the superiority of the doublet input; the pulse RMS error was $O(10^{-2})$ while the doublet RMS error was $O(10^{-3})$. The linearly-growing RMS error (ten to fifteen second segment) observed with both equation error (time domain) method and EUDKF in pulse input was consistent with a steady-state error in the system estimate. Since the pulse maneuver was completed by $t=3$, and the lack of dynamic spikes indicated that the transients had settled down by $t=10$, the steady-state growth of RMS error indicated a sample-invariant error in the estimate.

The RMS error divergent trends and higher error order of magnitude with the pulse input showed it to be a poor input waveform choice. In addition, the high magnitude input may violate the linearity assumption. Hence, the doublet was established as the best of the input waveform candidates. The tests to evaluate SID algorithm candidates were conducted with a doublet input.

B. **SID Algorithm**

Longitudinal Linear Plant

First analyzing the results tested with a linear plant of the longitudinal dynamics, it was noted EUDKF consistently performed the best in view of four differing sensor error types. Equation error (time domain) method performed only slightly worse than EUDKF as sensor error increased in complexity. Equation error (frequency domain) method performed better than EUDKF with a perfect sensor, but noise caused divergence and bias markedly deteriorated the solution.

In no-noise case (Figure 49), the equation error (frequency domain) method performed the best with EUDKF a close second. Equation error (time domain) method performed the worst. This showed the power of the frequency domain technique when no corruption exists. When sensor noise was added (Figure 50), the flaw of the equation error (frequency domain) method shows itself in the divergent trend. Also, the more sophisticated mathematical techniques of EUDKF

showed its worth by performing better than the equation error (time domain) technique. In face of sensor bias (Figure 51), equation error (frequency domain) method converged, but excessive RMS error was observed. Meanwhile, EUDKF and equation error (time domain) method show similar transient performance. With both sensor noise and bias (Figure 52), equation error (frequency domain) showed problems in converging. The convergence with bias still managed to overpower the divergence tendencies with noise so that the technique was globally convergent, albeit with large transient errors. Again, EUDKF and equation error (time domain) techniques were similar in performance with EUDKF showing better transients.

Considering the results from all four sensor error types, EUDKF converged the fastest with the least transient error. Equation error (time domain) method was a close second. This is by no-means an invalidation of the sophisticated mathematical techniques of EUDKF, but merely a reflection of the fact that it was a non-optimal solution for a linear problem due to the numerical errors stemming from the Jacobian calculation. Equation error (frequency domain) method was prone to divergence; this property was observed throughout all the tests performed.

<u>Longitudinal Nonlinear Plant</u>

While staying in the realm of longitudinal dynamics, a nonlinear plant was substituted and the same set of tests repeated. In such environment, equation error (frequency domain) again showed the divergent trends. In addition, EUDKF showed unequivocal superiority over the equation error (time domain) method.

In all cases, equation error (frequency domain) method showed an oscillating trend in the RMS error. This indicated an overly-estimated sensitivity in the $\hat{B}$ matrix. With each control input, the estimated measurements ($\hat{y}$) showed an over-reaction in comparison with the truth measurements (y). It was also noticeable that the RMS errors in equation error (frequency domain) method, while bounded, did not approach zero. With the equation error (time domain)

method, the RMS error approached zero exponentially after the transients died down (post $t=10$). This reflected the averaging nature of the method where the error incrementally approached zero. In contrast, EUDKF abruptly went to zero in a discontinuous fashion due to the statistical probability calculations of the method. While EUDKF showed higher peak RMS error than equation error (time domain) method, the overall RMS error ("area under the curve") was favorable to EUDKF.

In the case of a nonlinear plant, equation error (frequency domain) method showed its instability. EUDKF was superior to equation error (time domain) method because it was designed to deal with a nonlinear plant. In addition, its sophisticated mathematical techniques allowed it to minimize the overall RMS error.

Lateral Linear Plant

In examining the lateral dynamics, the linear plant was first tested. EUDKF proved superior not because it consistently had the least RMS error, but because both equation error methods diverged. Equation error (frequency domain) dramatically diverged and was not plotted.

In the no-noise case (Figure 57), equation error (time domain) method consistently showed slightly better performance across all states. It was noteworthy that both equation error (time domain) and EUDKF showed steady-state RMS error in the "slow" states of $[\phi,\psi]$. This was potentially due to the fact the input waveform did not adequately excite the slow states. The fast states, being more excitable, did not show this deficiency. In the case with sensor noise (Figure 58), equation error (time domain) diverged. With sensor bias (Figure 59), equation error (time domain) showed markedly better performance than EUDKF. However, in the face of both noise and bias (Figure 60), equation error (time domain) diverged in $[\beta]$ even though EUDKF showed the transient spikes of RMS error across all five states. Equation error (time domain) managed to converge with a quite small RMS error in the other four states as the convergent tendencies overpowered the divergent trend from sensor bias.

In the case of a linear lateral plant, EUDKF proved the superior method. Equation error (frequency domain) method diverged in all four sensor error cases. Equation error (time domain) method diverged in face of sensor bias but otherwise performed quite well. While the problem of using EUDKF to estimate a linear system showed itself, the robustness of the method decisively favored the technique.

Lateral Nonlinear Plant

In the test suites performed with a lateral nonlinear plant, the superiority of EUDKF was not quite so clearly established. Equation error (time domain) and EUDKF both showed superiority in different situations. Meanwhile, equation error (frequency domain) method diverged and was not plotted.

As explained in the previous paragraphs, the RMS errors in both slow states showed divergence with EUDKF being the worst offender of the two. In some cases such as in Figure 62, the ability of the EUDKF to abruptly reduce the error while equation error (time domain) incrementally averaged out the error was seen in $[\beta,p,r]$ states past $t=20$. In the case of sensor bias (Figure 63), EUDKF showed less RMS error than equation error (time domain) in the fast states. This was likely due to the assumption made by the equation error method that the error was zero-mean being violated. However, equation error (time domain) method showed less RMS error in estimating with the roll input in Figure 64. Both methods showed superiority under different situations.

While the lateral nonlinear plant provided no conclusive guidance on EUDKF vs. equation error (time domain), it reinforced the observation that equation error (frequency domain) was prone to divergence.

Summary

EUDKF showed superiority in linear and non-linear longitudinal dynamics, as well as linear lateral dynamics. EUDKF and equation error (time domain) showed pre-eminence under different situations with non-linear lateral dynamics. Overall, EUDKF showed superiority in minimizing RMS error, convergence time, and stability.

VII. Conclusions

FQ-BIT increased safety margins on UAS by reliably providing the AVO with accurate real-time flying qualities assessments. The design choices documented in this work showed the robustness, accuracy, and relevance of the metrics generated. In addition, FQ-BIT interfaced with the human operator in a clear and concise manner to facilitate swift and well-informed decision making. While the FQ-BIT is automated, a human is always in-the-loop; the data interpretation and final decision is left to the human.

The FQ-BIT used a doublet input waveform. $[\alpha, q]$ and $[\beta, \phi, p, \psi, r]$ were states gathered for processing in the longitudinal and lateral axis respectively. Extended UD-factorized Kalman Filter generated a linear-time-invariant state equation approximation in state-space form. Flying qualities metrics in the form of τ, ζ, ω, gain margin, phase margin, observability, and controllability were calculated. The results are displayed to the AVO in a color-coded, easy to interpret display. In addition to providing additional safety to operational UASs, FQ-BIT can be used in envelope-expansion flight tests where the real-time results can reduce risks and result in time and cost savings.

Future work on FQ-BIT includes tuning the EUDKF to better track the slow states in the lateral axis and to achieve unequivocal superiority over equation error (time domain) method. In addition, other SID algorithms can be investigated for use as a "sanity check" against the EUDKF results. An air vehicle dependent improvement involves investigating the use of regular artificial disturbances small in magnitude to provide continuous flying qualities monitoring (FQ-CBIT).

For a copy of the MATLAB files accompanying this work, please contact the aerospace department at California Polytechnic State University, San Luis Obispo.

APPENDIX

A. USAF UAS Class A Mishap FY 2000 to 2010

Date	AV	Failure Type
29-Nov-07	MQ-1L	Comms Failure
20-Nov-10	MQ-1B	Electrical Failure
22-Feb-09	MQ-1B	Electrical Failure
20-Apr-09	MQ-1B	Electrical Failure
17-Dec-07	MQ-1B	Electrical Failure
8-May-09	MQ-1B	FCS Failure
14-Sep-09	MQ-1B	FCS Failure
30-Dec-01	RQ-4A	FCS Failure
17-May-02	RQ-1L	FCS Failure
4-Oct-00	RQ-1L	FCS Failure
17-Sep-02	RQ-1L	Hazardous Meteorological Conditions
6-Dec-99	RQ-4A	Mission Planning Error
9-Dec-10	MQ-1B	Pilot Error
31-Aug-10	MQ-9	Pilot Error
16-Aug-10	MQ-1B	Pilot Error
28-Jul-10	MQ-1B	Pilot Error
20-Apr-10	MQ-1B	Pilot Error
3-Oct-09	MQ-1B	Pilot Error
26-Mar-07	MQ-1B	Pilot Error
21-Oct-05	MQ-9	Pilot Error
20-Mar-06	MQ-1	Pilot Error
3-Aug-06	MQ-1B	Pilot Error
13-Oct-04	MQ-1B	Pilot Error
24-Nov-04	MQ-1L	Pilot Error
14-Jun-04	MQ-1L	Pilot Error
22-Sep-04	MQ-1L	Pilot Error
25-Oct-02	RQ-1L	Pilot Error
22-Jan-02	RQ-1L	Pilot Error
25-Jan-02	RQ-1B	Pilot Error
30-Mar-01	RQ-1L	Pilot Error
14-Sep-00	RQ-1L	Pilot Error
11-Dec-03	RQ-1	Pilot Error / FCS Failure
5-May-11	MQ-1B	Propulsion Failure
19-Sep-10	MQ-1B	Propulsion Failure
19-Oct-08	MQ-1B	Propulsion Failure

20-Mar-09	MQ-9	Propulsion Failure
28-Apr-09	MQ-1B	Propulsion Failure
13-Aug-09	MQ-1B	Propulsion Failure
4-Sep-09	MQ-1B	Propulsion Failure
17-Jan-07	MQ-1B	Propulsion Failure
23-Feb-07	MQ-1B	Propulsion Failure
30-Jul-07	MQ-1B	Propulsion Failure
31-Jul-07	MQ-1B	Propulsion Failure
22-Jun-06	MQ-1L	Propulsion Failure
27-Mar-05	RQ-1L	Propulsion Failure
30-Mar-05	RQ-1L	Propulsion Failure
17-Aug-04	MQ-1L	Propulsion Failure
1-Jan-03	RQ-1B	Propulsion Failure
25-May-02	RQ-1B	Propulsion Failure
10-Jul-02	RQ-4A	Propulsion Failure
23-Oct-00	RQ-1K	Propulsion Failure

B. **Linear Plant (State-Space)**

Cessna 172, 5000ft, 120kt VTAS

1. Longitudinal

A =

	VTAS	α	θ	q
VTAS	-0.04422	18.74408	-32.2	0
alpha	-0.00135	-2.20202	0	0.97925
theta	0	0	0	1
q	0.00244	-23.72524	0	-6.13122

B =

	δ_e
VTAS	-6.24803
α	-0.20446
θ	0
q	-39.48824

2. Lateral

A =

	β	φ	p	ψ	r
β	-0.1474	0.14703	-0.00144	0	-0.99184
φ	0	0	1	0	0
p	-28.74922	0	-12.40917	0	2.53464
ψ	0	0	0	0	1
r	10.11937	0	-0.38174	0	-1.25975

B =

	δ_a	δ_r
β	0	0.08892
φ	0	0
p	57.49844	4.74847
ψ	0	0
r	-8.25118	-10.22835

REFERENCES

[1] "Aircraft Accident Investigation: MQ-1L "Predator", S/N 01-003079 at Indian Springs Air Force Auxiliary Air Field on 22 September 2004," USAF, 2004.

[2] "Aircraft Accident Investigation: MQ-1L "Predator", S/N 02-3082 at a deployed location on 20 March 2006," USAF, 2006.

[3] 162nd Fighter Wing, "MQ-1 Predator Unmanned Aerial Vehicle," USAF, March 2008. [http://www.162fw.ang.af.mil/resources/factsheets/factsheet.asp?id=11932   Accessed Feb 6, 2012.]

[4] Annaswamy, A.M., Jiang, J., Lavretsky, E., "Adaptive gain-scheduled controller in the presence of actuator anomalies," In Proceedings of AIAA Guidance, Navigation and Contr. Conf., Honolulu, HI, 2008.

[5] Annaswamy, A.M., Jiang, J., Lavretsky, E., "Stability margins for adaptive gain-scheduled controllers in the presence of time-delay, control saturation, and environmental disturbances," In Proceedings of AIAA Guidance, Navigation and Contr. Conf., Honolulu, HI, 2008.

[6] Burdine, T., Maj, USAF., "The Army's Organic Unmanned Aircraft Systems: An Unhealthy Choice for the Joint Operational Environment," Air & Space Power Journal, June 2009.

[7] Cheney, D., *Conduct of the Persian Gulf Conflict, An Interim Report to Congress,* July 1991.

[8] Drew, C., "Costly Drone Is Poised to Replace U-2 Spy Plane," The New York Times, Aug 2, 2011. [http://www.nytimes.com/2011/08/03/business/global-hawk-is-poised-to-replace-u-2-spy-plane.html Accessed Feb 6, 2012.]

[9] Dydek, Z.T., Annaswamy, A.M., Lavretsky, E., "Adaptive Control and the NASA X–15–3 Flight Revisited," IEEE Control Systems Magazine, vol. 3, pp. 32–48, June 2010.

[10] Dydek, Z.T., Annaswamy, A.M., Lavretsky, E., "Adaptive Control and the NASA X–15 Program: A Concise History, Lessons Learned, and a Provably Correct Design," In Proceedings of Amer. Contr. Conf., Seattle, WA, 2008.

[11] Ehrhard, T.P., "Air Force UAVs: The Secret History," Mitchell Institute, July 2010.

[12] Goodall, J., Miller, J., *Lockheed's SR-71 'Blackbird' family: A-12, F-12, M-21, D-21, SR-71*, Midland Publishing, 2002.

[13] IEEE Std. 952-1997, "Guide and Test Procedure for Single Axis Interferometric Fiber Optic Gyros," IEEE, 1997, p.63.

[14] Jones, C., *Unmanned Aerial Vehicles: An Assessment of Historical Operations and Future Possibilies*, Air Command and Staff College, 1997

[15] Klein, V., Morelli, E. A., *Aircraft System Identification: Theory and Practice*, AIAA, 2006.

[16] Lavretsky, E.,"Adaptive Output Feedback Design Using Asymptotic Properties of LQG / LTR Controllers," In Proceedings of AIAA Guidance, Navigation and Contr. Conf., Toronto, Ontario, Canada, 2010.

[17] Lavretsky, E., Annaswamy, A.M., "On the Computation of Stability Margins for Adaptive Controllers using Linear System Tools," In Proceedings of AIAA Guidance, Navigation and Contr. Conf., Chicago, IL, 2009.

[18] Longino, D. LtCol, USAF, *Role of Unmanned Aerial Vehicles in Future Armed Conflict Scenarios*, Air University Press, Maxwell AFB, AL, 1995.

[19] Mathewson, E., Col, USAF, "Air Force Unmanned Aircraft System Update," USAF. [http://www.wired.com/images_blogs/dangerroom/files/Mathewson.pdf]

[20] Mehiel, E.A., "On direct model reference adaptive controller design for flexible space structures," University of Colorado, CO, 2003.

[21] Morelli, E. A., "Practical Input Optimization for Aircraft Parameter Estimation Experiments," NASA Contractor Report 191462, May 1993.

[22] Morelli, E. A., "Practical Aspects of the Equation-Error Method for Aircraft Parameter Estimation," *AIAA Atmospheric Flight Mechanics Conference*, *Keystone, Co.,* AIAA Paper 2006-6144, 1999.

[23] Morelli, E. A., "Real-Time Parameter Estimation in the Frequency Domain," AIAA Paper 99-4043, 1999.

[24] Navigation Systems, "LN-200 FOG Family Advanced Airborne IMU/AHRS", Nothrop Grumman, Woodland Hills, CA. 2010.

[25] Nelson, R. C., *Flight Stability and Automatic Control Second Edition*, McGraw Hill, 1998.

[26] Pavlath, G.A., "Fiber-Optic Gyros: The Vision Realized", Paper#24235, Northrop Grumman.

[27] Patel, V., Cao, C., Hovakimyan, N., Wise, K., Lavretsky, E., "L1 Adaptive Controller for Tailless Unstable Aircraft in the Presence of Unknown Actuator Failures," International Journal of Control, Vol. 82, Issue 4, pp. 705-720, April 2009.

[28] Peyada, N. K., Sen, A., and Ghosh, A. K., "Aerodynamic Characterization of HANSA-3 aircraft using Equation Error, Maximum Likelihood and Filter Error Methods," *Proceedings of the International MultiConference of Engineers and Computer Scientists 2008 Vol II,* IMECS, Hong Kong, Mar. 2008.

[29] Raol, J. R., Girija, G., Singh, J., *Modeling and Parameter Estimation of Dynamic Systems*, The Institution of Engineering and Technology, London, United Kingdom, 2004.

[30] Sensor Systems, "Brochure for Model 0861CAL Angle of Attack Sensor," Goodrich Corporation, Burnsville, MN, 2002

[31] Sensor Systems, "Brochure for Model 0861DV1 Angle of Attack Sensor," Goodrich Corporation, Burnsville, MN, 2005

[32] Shafer, M. F., "Flight Investigation of Various Control Inputs Intended for Parameter Estimation," NASA Technical Memorandum 85901, Aug. 1984.

[33] Smith, S. W., *The Scientist and Engineer's Guide to Digital Signal Processing,* California Technical Publishing, 1997.

[34] Software Enabled Control, "Non-Linear F-16 Aircraft Model," University of Minnesota, Minnesota. [http://www.aem.umn.edu/people/faculty/balas/darpa_sec/SEC.Software.html]

[35] Stevens, B. L., Lewis, F. L., *Aircraft Control and Simulation Second Edition*, Wiley, 2003.

[36] Stockwell, W., "Angle Random Walk," Crossbow Technology, Inc.

[37] Thornton, C. L., "Triangular Covariance Factorizations for Kalman Filtering". (PhD thesis) NASA Technical Memorandum 33-798, Oct 1976.

[38] Tomita, Y., Damen, A., Van Den Hof, P., "Equation Error versus Output Error Methods," *Ergonomics*, Vol. 35, Nos. 5/6, 1992, pp. 551-564.

[39] Vincent, T. L., Grantham, W. J., *Nonlinear and Optimal control Systems*. John Wiley & Sons, 1997.

[40] Wagner, W., *Lightning Bugs and Other Reconnaissance Drones.* Aero Publishers, 1982.

[41] Wang, J., Patel, V., Cao, C., Hovakimyan, N., Lavretsky, E., "Novel L1 Adaptive Control Methodology for Aerial Refueling with Guaranteed Transient Performance," Journal of Guidance, Control, and Dynamics, Vol. 31, No. 1, January–February, pp. 182-193, 2008.

[42] Widrow, B., Plett, Gregory., Ferreira, Edson., Lamego, Marcelo., "Adaptive inverse control based on nonlinear adaptive filtering," Information Systems Lab, Stanford University, 1998.

[43] Wolski, E., "Unmanned Aircraft Systems", US Department of Defense, Jan 9, 2009. [http://www.wired.com/images_blogs/dangerroom/files/Wolski.pdf]

[44] Young, A., Cao, C., Patel, V., Hovakimyan, N., Lavretsky, E., "Adaptive Control Design Methodology for Nonlinear-in-Control Systems in Aircraft Applications," Journal of Guidance, Control, and Dynamics, Vol. 30, No. 6, November–December, pp. 1770-1782, 2007.