Summer 7-23-2014

State Estimation for Autopilot Control of Small Unmanned Aerial Vehicles in Windy Conditions

David Paul Poorman
University of Colorado Boulder, david.poorman@colorado.edu

STATE ESTIMATION FOR AUTOPILOT CONTROL OF
SMALL UNMANNED AERIAL VEHICLES
IN WINDY CONDITIONS

DAVID PAUL POORMAN

B.S., University of Colorado at Boulder, 2010

A thesis submitted to the
Faculty of the Graduate School of the
University of Colorado in partial fulfillment
of the requirement for the degree of
Master of Science
Department of Aerospace Engineering Sciences
2014

This thesis entitled:
State Estimation for Autopilot Control of Small Unmanned Aerial Vehicles in Windy Conditions
written by David Paul Poorman
has been approved for the Department of Aerospace Engineering Sciences

(Dale Lawrence)

(Eric Frew)

*Date*_______________

The final copy of this thesis has been examined by the signatories, and we
Find that both the content and the form meet acceptable presentation standards
Of scholarly work in the above mentioned discipline.

Abstract

Poorman, David Paul (M.S. Aerospace Engineering Sciences)

State Estimation for Autopilot Control of Small Unmanned Aerial Vehicles in Windy Conditions

Thesis directed by Professor Dale A. Lawrence

The use of small unmanned aerial vehicles (UAVs) both in the military and civil realms is growing. This is largely due to the proliferation of inexpensive sensors and the increase in capability of small computers that has stemmed from the personal electronic device market. Methods for performing accurate state estimation for large scale aircraft have been well known and understood for decades, which usually involve a complex array of expensive high accuracy sensors. Performing accurate state estimation for small unmanned aircraft is a newer area of study and often involves adapting known state estimation methods to small UAVs. State estimation for small UAVs can be more difficult than state estimation for larger UAVs due to small UAVs employing limited sensor suites due to cost, and the fact that small UAVs are more susceptible to wind than large aircraft. The purpose of this research is to evaluate the ability of existing methods of state estimation for small UAVs to accurately capture the states of the aircraft that are necessary for autopilot control of the aircraft in a Dryden wind field. The research begins by showing which aircraft states are necessary for autopilot control in Dryden wind. Then two state estimation methods that employ only accelerometer, gyro, and GPS measurements are introduced. The first method uses assumptions on aircraft motion to directly solve for attitude information and smooth GPS data, while the second method integrates sensor data to propagate estimates between GPS measurements and then corrects those estimates with GPS information. The performance of both methods is analyzed with and without Dryden wind, in straight and level flight, in a coordinated turn, and in a wings level ascent. It is shown that in zero wind, the first method produces significant steady state attitude errors in both a coordinated turn and in a wings level ascent. In Dryden wind, it produces large noise on the estimates for its attitude states, and has a non-zero mean error that increases when gyro bias is increased. The second method is shown to not exhibit any steady state error in the tested scenarios that is inherent to its design. The second method can correct for attitude errors that arise from both integration error and gyro bias states, but it suffers from lack of attitude error observability. The attitude errors are shown to be more observable in wind, but increased integration error in wind outweighs the increase in attitude corrections that such increased observability brings, resulting in larger attitude errors in wind. Overall, this work highlights many technical deficiencies of both of these methods of state estimation that could be improved upon in the future to enhance state estimation for small UAVs in windy conditions.

This work is dedicated to Grandmary and Grandbob.

Contents

1 Introduction 1

2 Nomenclature 4

3 Background Information 7

3.1 Reference Frames 7

3.2 The Wind Triangle 10

3.3 The Extended Kalman Filter 11

3.3.1 Measurement Update 13

3.3.2 Time Update 14

4 Aircraft States Needed for Autopilots 17

4.1 Scoping the Estimation Problem 17

4.2 Overview of the Autopilot 20

4.3 Simulation Setup 24

4.3.1 Benchmark 1: Straight and Level Flight 24

4.3.2 Benchmark 2: Coordinated Turn 25

4.3.3 Benchmark 3: Wings Level Ascent 27

4.3.4 The Dryden Wind Model 28

4.4 Sensors 31

4.5 Demonstration of Autopilot Performance with Limited State Information 32

4.5.1 Autopilot Performance under Benchmark 1 32

4.5.2 Autopilot Performance under Benchmark 2 36

4.5.3 Autopilot Performance under Benchmark 3 39

4.6 Relative Importance of the Chosen States 43

4.7 Autopilot Summary 46

5 Method 1: IMU Attitude, Smoothed GPS, and Pressure Sensors 47

5.1 Description 47

5.1.1 The Transport Theorem 48

5.1.2 Attitude Determination 49

5.1.3 Extended Kalman Filter 1 52

5.1.4 Extended Kalman Filter 2 55

5.1.5 Method 1 Summary 60

5.2 Detailed Simulation Results 62

5.2.1 Zero Wind 63

5.2.2 Dryden Wind 80

5.3 Method 1 Analysis Summary 95

6 Method 2: IMU and GPS Sensor Fusion 97

6.1 Description 97

6.1.1 Direct IMU Numerical Integration 97

6.1.2 Sensor Fusion through Kalman Filtering 100

6.1.3 Method 2 Summary 108

6.2 Detailed Simulation Results 109

6.2.1 IMU Only Solution (No GPS/EKF3 Corrections) 110

6.2.2 Zero Winds 118

6.2.3 Dryden Wind 143

6.2.4 Method 2 Analysis Summary 169

7 Method Comparison and Future Work 171

7.1.1 Direct Attitude Sensing 173

7.1.2 Replace Integration Method in State Estimation Method 2 175

7.1.3 Parallel Estimators 175

8 Conclusion 176

9 References 179

Figures

Figure 3.1: Aircraft Reference Frames 8

Figure 3.2: The Wind Triangle 10

Figure 4.1: Straight and Level Flight Illustration 25

Figure 4.2: Coordinated Turn Illustration 26

Figure 4.3: Wings Level Ascent Illustration 28

Figure 4.4: Example Dryden Wind Model Simulation 30

Figure 4.5: Command Tracking Error for Benchmark 1 in Zero Wind 33

Figure 4.6: Command Tracking Error for Benchmark 1 in Light Wind 35

Figure 4.7: Command Tracking Error for Benchmark 2 in Zero Wind 37

Figure 4.8: Command Tracking Error for Benchmark 2 in Light Wind 38

Figure 4.9: Command Tracking Error for Benchmark 3 in Zero Wind 40

Figure 4.10: Command Tracking Error for Benchmark 3 in Light Wind 42

Figure 4.11: Effects of Bad Pitch Estimation 45

Figure 5.1: Method 1 Flow Chart 62

Figure 5.2: Method 1 State Estimation for Benchmark 1 in Zero Wind and Zero Gyro Bias Drift 64

Figure 5.3: Method 1 State Estimation for Benchmark 1 in Zero Wind 67

Figure 5.4: Estimated Yaw Angle for Benchmark 1 in Zero Wind 68

Figure 5.5: Estimated Yaw Angle for Benchmark 1 in Zero Wind and Zero Gyro Bias Drift 68

Figure 5.6: Method 1 Results for Benchmark 2 in Zero Wind 72

Figure 5.7: Validity of Method 1 Assumptions in Benchmark 2 and Zero Wind 73

Figure 5.8: Method 1 Results for Benchmark 2 in Zero Wind and Zero Gyro Bias Drift 75

Figure 5.9: Method 1 Benchmark 3 Results in Zero Wind 76

Figure 5.10: Error Statistics for Method 1 in Benchmark 3 and Zero Wind 76

Figure 5.11: Method 1Assumption Validity for Benchmark 3 in Zero Wind 77

Figure 5.12: Method 1 State Estimation with True Inertial Velocity Values 78

Figure 5.13: Method 1 Results for Benchmark 1 in Dryden Wind 81

Figure 5.14: Assumption Validity for Benchmark 1 in Dryden Wind 82

Figure 5.15: Assumption (5.7) to Estimated Attitude Zoom Comparison 85

Figure 5.16: Method 1 Results under Benchmark 2 and Dryden Wind 86

Figure 5.17: Method 1 Assumption Validity under Benchmark 2 in Dryden Wind 88

Figure 5.18: Method 1 Results for Benchmark 3 in Dryden Wind 91

Figure 5.19: Method 1 Assumption Validity for Benchmark 3 in Dryden Wind 92

Figure 5.20: Method 1 Assumption Validity for Benchmark 3 in Dryden Wind and Delayed Climb 93

Figure 6.1: Method 2 Flow Chart 109

Figure 6.2: Airspeed Estimation for Method 2 109

Figure 6.3: State Estimation Results of Method 2 without EFK3 in Benchmark 1 in Zero Wind 111

Figure 6.4: Extra State Estimation Results of Method 2 without EFK3 in Benchmark 1 in Zero Wind 113

Figure 6.5: State Estimation Results of Method 2 Benchmark 1 without EFK3 and with True Attitude Knowledge 114

Figure 6.6: Extra State Estimation Results of Method 2 Benchmark 1 without EFK3 and with True Attitude Knowledge 115

Figure 6.7: State Estimation Results for Method 2 without EKF3 under Benchmark 1 in Zero Wind 116

Figure 6.8: Extra State Estimation Results for Method 2 without EKF3 under Benchmark 1 in Zero Wind 117

Figure 6.9: Method 2 State Estimation for Benchmark 1 in Zero Wind and Zero Gyro Bias Drift 119

Figure 6.10: Extra Estimated States from Method 2 for Benchmark 1 with Zero Wind and Zero Gyro Rate Bias 120

Figure 6.11: Dependence of Difference between Accelerometer Measurement and Velocity Errror Growth on Yaw Growth 123

Figure 6.12: Method 2 State Estimates for Benchmark 1 in Zero Wind 126

Figure 6.13: Extra State Estimates for Benchmark 1 in Zero Wind 127

Figure 6.14: Bias Estimates for Method 2 in Benchmark 1 in Zero Wind 129

Figure 6.15: State Estimation Results for Benchmark 2 in Zero Wind 131

Figure 6.16: Bias Estimation for Benchmark 2 in Zero Wind 132

Figure 6.17: Extra State Estimation Results for Benchmark 2 in Zero Wind... 133

Figure 6.18: Bias Estimates for Benchmark 2 with Time-Varying Jerk Vector 137

Figure 6.19: State Estimation Results for Benchmark 3 in Zero Wind 138

Figure 6.20: Extra Estimated States for Method 2 in Benchmark 3 and Zero Wind 140

Figure 6.21: Estimated Bias States for Method 2 in Benchmark 3 and Zero Wind.................................. 141

Figure 6.22: State Estimation Results for Method 2 Benchmark 1 in Dryden Wind and Zero Bias Drift.. 144

Figure 6.23: Extra Estimated States for Method 2 Benchmark 1 in Wind with Zero Rate Bias 150

Figure 6.24: Sensor Bias Estimates for Method 2 Benchmark 1 in Wind with Zero Rate Bias 150

Figure 6.25: State Estimation Results for Method 2 in Benchmark 1 and Dryden Wind.......................... 153

Figure 6.26: Extra State Estimates for Method 2 in Benchmark 1 and Dryden Wind 155

Figure 6.27: Bias State Estimates for Method 2 in Benchmark 1 and Dryden Wind 156

Figure 6.28: State Estimation Results for Method 2 in Benchmark 2 in Dryden Wind............................ 158

Figure 6.29: Gyro Bias Estimates for Benchmark 2 in Dryden Wind... 160

Figure 6.30: Extra State Estimation Results for Method 2 in Benchmark 2 and Dryden Wind 162

Figure 6.31: State Estimation Results for Method 2 in Benchmark 3 and Dryden Wind.......................... 163

Figure 6.32: Extra State Estimation Results for Method 2 in Benchmark 3 and Dryden Wind 165

Figure 6.33: Bias Estimation Results for Method 2 in Benchmark 3 and Dryden Wind 166

1 Introduction

In all fields of aviation, from large commercial aircraft and military fighters to small scale Unmanned Aerial Vehicles (UAVs), accurate state estimation is a critical part of an aircraft's control system that is often taken for granted or simply overlooked. In large expensive aircraft like airliners or military fighters, the methods of obtaining accurate state information are well known and usually involve a complex array of numerous sensors to get the information [1]. However, today state estimation of small UAVs is at least as much of a topic of concern as the design of a controller. State estimation of small UAVs is a newer area of study and can in some ways be more difficult to implement than state estimation for large complex aircraft. This is due to two main aspects of small UAVs.

First, large aircraft employ a sophisticated array of air data sensors to obtain detailed information about the airflow around the aircraft at high frequencies. They use this information combined with specialized, high-precision, and often expensive Inertial Measurement Units (IMUs), Global Positioning System (GPS) receivers, and other inertial sensors to accurately estimate vehicle states. Unfortunately, small UAVs can typically only support a limited sensor suite due to the size, weight, and cost of a large number of sophisticated sensors. Thus, state estimation has to be done with both fewer and lower cost sensors than is typically done on large aircraft. This means that the algorithms used for state estimation must be re-assessed before they are applied to small UAVs.

Second, the effects of wind on the motion of a small UAV is much greater than on large aircraft. Reference [2] shows that as the scale of an aircraft shrinks, the aircraft is increasingly subject to disturbances caused by wind and turbulence. For example, it is not atypical for the wind speeds experienced by small aircraft to be 50% of their total airspeed. That is almost never the case on large aircraft. Many techniques for estimating small UAV states during zero or steady wind have been developed and widely published, such as in [3], [4], [5], and [6]. These techniques generally work well for their designed purpose of zero or steady wind. However, in practice, if highly variable, high frequency gusty wind is encountered, these methods can produce significant amounts of error between the estimated value of the states and the true value of the states. If this error is large enough, depending on the design or the aircraft's controller, this can start unstable behavior in both the estimator and controller, and cause the aircraft to crash.

The purpose of this research is to assess the previously unassessed effectiveness of existing methods for state estimation for small UAVs in highly variable windy conditions. There are two common methods of state estimation that are used today by small UAVs. This work simulates both of these methods in a MATLAB®/Simulink® aerodynamic model under different benchmark maneuvers in varying wind strengths to assess their performance with and without wind to find and analyze the reasons for the observed behavior of the state estimate errors in wind. The contributions of this work are detailed understandings of why these methods of state estimation undergo their observed behavior in wind so that these estimators can be improved upon in the future to improve the accuracy of state estimation for small UAVs in wind.

The first method that is analyzed makes assumptions about the acceleration and velocity of the aircraft in order to measure attitude and then uses this attitude to smooth GPS data to get translational information [4]. This work addresses these assumptions with and without wind present in order to understand how wind affects this state estimation method. The second method that is analyzed does not make assumptions on the acceleration or velocity of the aircraft, and instead integrates upon a-priori knowledge of attitude and translation states using gyro and accelerometer sensor data to update the attitude and translation estimates [7]. It then corrects these integrated attitude and translation estimates with GPS provided position and velocity. The effectiveness of this method's ability to correct both the translational states and the attitude states will be heavily scrutinized with and without wind present so that the change in the observability of the IMU error between zero wind and wind can be assessed. This detailed analysis of the change in performance of both of these methods when wind is added will provide the insights that are required to improve upon these existing methods in the future.

2 Nomenclature

B Designates the aircraft body frame. Always used as a subscript or superscript.

${}^{B}[\boldsymbol{A}]$ Designates the vector $\boldsymbol{A}$ written in aircraft body B frame components

N Designates the inertial North-East-Down (N-E-D) earth-fixed frame. Earth-fixed is considered the inertial reference unless explicitly stated otherwise.

${}^{N}[\boldsymbol{A}]$ Designates the vector $\boldsymbol{A}$ written in inertial N frame components.

p_n Inertial Earth position towards North, origin at simulation initial position

p_e Inertial Earth position towards East, origin at simulation initial position

p_d Negative Earth relative altitude. Zero altitude is ground level.

$\boldsymbol{p}_{B/N}$ Vector of the position of the aircraft body frame B relative to the inertial Earth N frame.

h Height of the aircraft above ground level. Negative of p_d.

u Component of inertial velocity vector $\boldsymbol{V}_{\boldsymbol{B/N}}$ along the X axis of the aircraft.

v Component of inertial velocity vector $\boldsymbol{V}_{\boldsymbol{B/N}}$ along the Y axis of the aircraft.

w Component of inertial velocity vector $\boldsymbol{V}_{\boldsymbol{B/N}}$ along the Z axis of the aircraft.

$\boldsymbol{V}_{B/N}$ The inertial velocity vector. The velocity of the aircraft body frame B relative to the inertial Earth N frame.

ϕ Roll angle in degrees. Third Euler angle.

θ Pitch angle in degrees. Second Euler angle.

ψ Yaw angle in degrees. First Euler angle.

$\boldsymbol{\psi}$ Vector of Euler angle components, relating the N frame to the B frame: $[\phi\ \theta\ \psi]^T$

χ Course angle. Only the same as yaw angle in zero wind and sideslip.

p Roll rate

q Pitch rate

r Yaw rate

$\boldsymbol{\omega}_{B/N}$ Angular velocity of the aircraft body frame relative to Earth inertial frame.

$\boldsymbol{\alpha}_{B/N}$ Angular acceleration of the aircraft body frame relative to the Earth inertial frame.

V_a Airspeed. Magnitude of the airspeed vector $\boldsymbol{V}_a$

$\boldsymbol{V}_a$ Vector representing the speed of the air mass relative to the aircraft. Points into the direction the air mass is coming from.

α Angle of Attack

β Angle of Sideslip

a_x The component of the inertial acceleration of the aircraft in the direction of the body X axis

a_y The component of the inertial acceleration of the aircraft in the direction of the body Y axis

a_z The component of the inertial acceleration of the aircraft in the direction of the body Z axis

$\boldsymbol{a}_{B/N}$ Acceleration of body frame relative to Earth inertial frame.

w_x Component of the wind velocity $\boldsymbol{V}_w$ in the direction of the aircraft body positive X axis.

w_y Component of the wind velocity $\boldsymbol{V}_w$ in the direction of the aircraft body positive Y axis.

w_z Component of the wind velocity $\boldsymbol{V}_w$ in the direction of the aircraft body positive Z axis.

w_n Component of the wind velocity $\boldsymbol{V}_w$ in the direction *from* the north.

w_e Component of the wind velocity $\boldsymbol{V}_w$ in the direction *from* the east.

w_d Component of the wind velocity $\boldsymbol{V}_w$ in the downward direction.

$\boldsymbol{V}_w$ Wind speed velocity vector. Indicates which direction the wind is coming *from*.

$\boldsymbol{\Phi}_{\mathrm{EKFY}}$ State transition matrix. Maps the values of one state to the next state inside the time update section of a Kalman filter.

$\boldsymbol{P}_{EKFY}$ Covariance matrix of the states in extended Kalman filter Y.

$\boldsymbol{R}_{EKFY}$ Matrix of expected sensor noise that is used in extended Kalman filter Y.

$\boldsymbol{x}_{EKFY}$ The vector of states that is being estimated by extended Kalman filter Y.

$\boldsymbol{y}_{EKFY}$ The vector of measurements that are used in extended Kalman filter Y.

$\boldsymbol{u}_{EKFY}$ The vector of known truth inputs used by extended Kalman filter Y.

$\boldsymbol{F}_{EKFY}$ The matrix that maps the measurements and inputs of an extended Kalman filter Y to the time derivatives of the states of the extended Kalman filter Y

$\boldsymbol{H}_{EKFY}$ The matrix that maps the states and inputs of an extended Kalman filter Y to the measurements of the extended Kalman filter Y.

ϵ State noise vector.

η Measurement noise vector.

δ_e Elevator deflection, as commanded by the autopilot.

δ_a Aileron deflection, as commanded by the autopilot.

δ_r Rudder deflection, as commanded by the autopilot.

δ_t Percentage of available thrust that is commanded by the autopilot, on a unit scale where $\delta_t = 0$ means use no thrust and $\delta_t = 1$ means use all available thrust. Thrust force is simulated as being directly proportional to δ_t.

μ Value that represents the mean of a set of data. Same as the first moment of a set of data. The expected value operator returns the mean.

σ Value that represents the standard deviation of a set of data. The standard deviation of a dataset is the square root of the variance of the dataset.

E Operator that calculates the expected value of a set of data. The value returned is the mean of the set of data that was operated on.

min Represents the minimum value of a set of data.

max Represents the maximum value of a set of data.

$\frac{d^{B}}{dt}$ Time derivative performed in the B Frame.

$\hat{x}$ The hat symbol designates the output of the state estimator for the variable x

b_{g_x} The gyro bias about the body frame axis designated by x

b_{a_x} The accelerometer bias in the direction of the body frame axis designated by x

3 Background Information

This section contains technical background information that is used throughout this document.

3.1 Reference Frames

When expressing the components of vectors, there are two frames of reference that are used in this document. Those are the aircraft body fixed B frame and the earth-fixed inertial N frame. An earth fixed frame isn't truly inertial, as the frame rotates as the earth rotates, and the earth rotates about the Sun, and the Sun rotates around the galaxy, and so on. However, in this document, the earth-fixed N frame is considered the base inertial reference frame, unless it is otherwise explicitly stated. The one exception is in the covariance calculation in the sensor fusion of method 2.

The body fixed B frame is a Cartesian frame whose origin is at the center of gravity of the aircraft. The X (1st) axis points outward from the nose of the aircraft, the Y (2nd) axis points out the right side of the aircraft, and the Z (3rd) axis points directly downward out of the aircraft. Knowing the exact geometric orientation of this frame with respect to aircraft parts is not necessary for this study. For the purposes of this study, all sensors will be assumed to be at the center of gravity of the aircraft.

The inertial earth-fixed N frame is also referred to as the North-East-Down or N-E-D frame. The N frame is also a Cartesian frame, and its origin is at the surface of the Earth, at the position of the start of the simulations in this document. All distances measured in the N frame are with respect to the starting point of the simulation, not actual latitude and longitude coordinates.

A visualization of these two frames is shown in Figure 3.1.

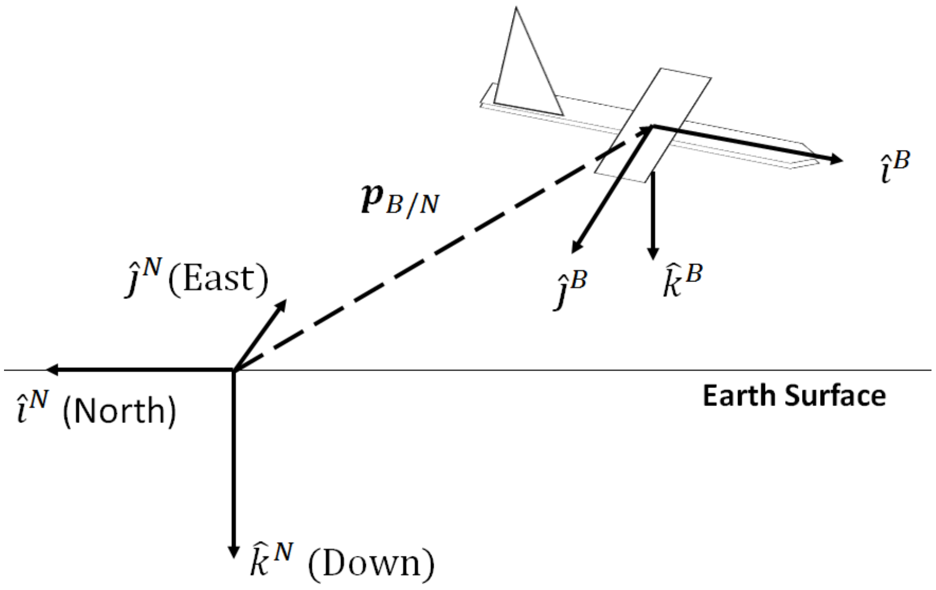


Figure 3.1: Aircraft Reference Frames

The relative orientation of these two frames is described in this paper using 3-2-1 Euler angles. These Euler angles describe the orientation of the aircraft B frame relative to the inertial N frame through a series of three rotations. The B frame is obtained by rotating the first two axes of the N frame about the 3rd axis by an amount known as the yaw angle ψ, then a rotation about the new transformed axes' 2nd axis by an amount known as the pitch angle θ, and then a last rotation about this new transformed axes' 3rd axis by an amount known as the roll angle ϕ. These three measures of rotation comprise the 3-2-1 Euler angle collection in equation (3.1).

$$\boldsymbol{\psi} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \tag{3.1}$$

This collection of Euler angles is referred to as the aircraft's attitude. Using this definition, the attitude information can be used to map a vector written in body frame components into inertial frame components with equation (3.2).

$$^{N}[\boldsymbol{A}] = \boldsymbol{R}_{NB}\, ^{B}[\boldsymbol{A}] \tag{3.2}$$

$\boldsymbol{R}_{NB}$ in Equation (3.2) represents the direction cosine matrix (DCM) or rotation matrix that maps vectors from the B frame into the N frame. The definition of this DCM for 3-2-1 Euler angles (3.1) is in equation (3.3), which is provided in [8].

$$\boldsymbol{R}_{NB} = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \cos\phi\sin\psi & \cos\phi\cos\psi\sin\theta + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\phi\cos\psi & \cos\phi\sin\psi\sin\theta - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{3.3}$$

A DCM is an orthogonal matrix such that its transpose performs an inverse mapping, such as in equations (3.4) and (3.5).

$$\boldsymbol{R}_{BN} = \boldsymbol{R}_{NB}^{T} \tag{3.4}$$

$${}^{B}[\boldsymbol{A}] = \boldsymbol{R}_{BN}\,{}^{N}[\boldsymbol{A}] \tag{3.5}$$

Finally, it is important to realize that the time derivatives of Euler angles are not simply the rate of change of their values, but are instead the rate of change of their angles in their respective intermediate frames. Reference [9] provides this relationship, given in equation (3.6).

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} {}^{B}[\boldsymbol{\omega}_{B/N}] \tag{3.6}$$

${}^{B}[\boldsymbol{\omega}_{B/N}]$ is the angular velocity of the aircraft with respect to the inertial frame, written as aircraft body frame components of roll rate, yaw rate, and pitch rate, as given in equation (3.7).

$${}^{B}[\boldsymbol{\omega}_{B/N}] = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3.7}$$

Thus, the relationship between the yaw, pitch, and roll rates and the Euler angle rates is equation (3.8).

$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} = \begin{bmatrix}1 & \sin\phi\tan\theta & \cos\phi\tan\theta\\ 0 & \cos\phi & -\sin\phi\\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{3.8}$$

3.2 The Wind Triangle

The next piece of information that is used throughout this document is the concept of the wind triangle. The wind triangle relates the three measures of velocity that are used throughout this document. The wind triangle is shown in Figure 3.2.

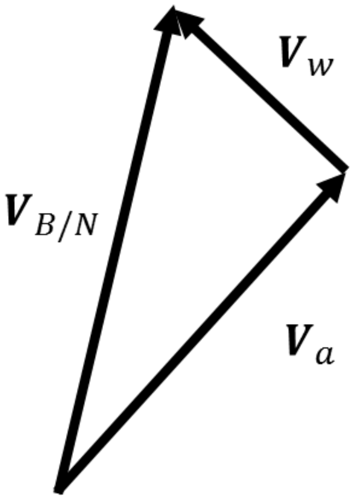


Figure 3.2: The Wind Triangle

The vector $\boldsymbol{V}_{B/N}$ represents the velocity of the aircraft with respect to the inertial Earth-fixed frame. In this paper, vector subscripts "X/Y" designate the vector quantity of the X frame relative to the Y frame. The way to read $\boldsymbol{V}_{B/N}$ is "the velocity of the body frame B with respect to the inertial frame N." The vector $\boldsymbol{V}_a$ represents the velocity of the air over the aircraft, or the velocity of the air with respect to the body B frame. It points in the direction that the air mass is flowing *towards*. The vector $\boldsymbol{V}_w$ is the velocity of the wind, or the velocity of the air with respect to the inertial N frame. It points in the direction the wind is coming *from*. As illustrated by the wind triangle in Figure 3.2, these vectors are related through equation (3.9).

$$\boldsymbol{V}_{B/N} = \boldsymbol{V}_a + \boldsymbol{V}_w \tag{3.9}$$

Now, the notion of air data is also used throughout this document. Air data is the coined term for the scalar values for airspeed V_a, angle of attack α, and angle of sideslip β. When the airspeed vector is written in aircraft body coordinates, air data is related to the vector through the following relationship.

$${}^{\boldsymbol{B}}[V_a] = V_a \begin{bmatrix} \cos\alpha\cos\beta \\ \sin\beta \\ \sin\alpha\cos\beta \end{bmatrix} \tag{3.10}$$

3.3 The Extended Kalman Filter

A Kalman filter is a software algorithm that determines a "best guess" estimate of a set of states given relationships of noisy sensor inputs to the states that are being estimated. This "best guess" is the values for the states that give the minimum variance of the errors between those estimated state values and the state values that a sensor model gives. This sensor model is a known or expected relationship from available sensor data to the states that are being estimated. To explain that in mathematical terms, an extended Kalman filter (EKF) filters the states that are provided by a nonlinear state-space model that describes the relationship of the states to sensors and the relationship of the states to state time derivatives. This set of relationships is described by the set of equations (3.11) and (3.12).

$$\dot{\boldsymbol{x}}_{EKF,expected} = \boldsymbol{F}_{EKF}\left(\boldsymbol{x}_{EKF,expected}, \boldsymbol{u}_{EKF}\right) + \epsilon \tag{3.11}$$

$$\boldsymbol{y}_{EKF} = \boldsymbol{H}_{EKF}\left(\boldsymbol{x}_{EKF,expected}, \boldsymbol{u}_{EKF}\right) + \eta \tag{3.12}$$

In this non-linear state space relationship, $\boldsymbol{x}_{EKF,expected}$ is a vector of the states being estimated by the EKF, $\boldsymbol{y}_{EKF}$ is a vector of the measurements, and $\boldsymbol{u}_{EKF}$ is the vector of pre-determined inputs to the EKF that are to be taken as truth. As discussed later, not all values needed by a Kalman filter can be treated as measurements. Most often, some values need to be treated as true error-free measurements for the state-space system in (3.11) and (3.12) to be formed. A normal (not extended) Kalman filter is an extended Kalman filter whose state space relationships in (3.11) and (3.12) are linear.

Now, the variance of the error in the estimate of the states is represented by the covariance matrix $\boldsymbol{P}_{EKF}$ whose definition is in equation (3.13).

$$\boldsymbol{P}_{EKF} = E\left[\left(\boldsymbol{x}_{EKF} - \boldsymbol{x}_{EKF,expected}\right)\left(\boldsymbol{x}_{EKF} - \boldsymbol{x}_{EKF,expected}\right)^{T}\right] \tag{3.13}$$

$\boldsymbol{x}_{EKF}$ Represents the estimated state vector that is the output of the EKF. $\boldsymbol{x}_{EKF,expected}$ Represents the states that are computed via the expected relationship to the measurements that is given by equations (3.11) and (3.12). The operator E is the expected value, which returns the first moment of a dataset, the mean. The expected value operator determines the mean of all of the occurrences of the expression it operates on. When Kalman filters are used in real-time control systems like for small UAVs, equation (3.13) calculates the mean of all the error covariances of all the updates to the states that have occurred thus far.

Next, the minimum error variance estimate is obtained by minimizing the covariance matrix $\boldsymbol{P}_{EKF}$. In an EKF, this is done in a two-step procedure; once each time an update for the measurements $\boldsymbol{y}$ is obtained, using relationship (3.12), and once each time a prediction can be made using relationship (3.11).

3.3.1 Measurement Update

The procedure that is followed at each update of the measurement vector $\boldsymbol{y}_{EKF}$ is called the measurement update. The measurement update creates a minimum variance estimate of the states $\boldsymbol{x}_{EFK,k}$ and updates the covariance matrix $\boldsymbol{P}_{EKF,k}$ for the current update k. These actions are performed by first calculating the Kalman gain matrix $\boldsymbol{K}_{EKF,k}$ for the current update k as is outlined in equation (3.14).

$$\boldsymbol{H}_{EKF,k} = \boldsymbol{H}_{EKF}\left(\boldsymbol{x}_{EKF,k-1}, \boldsymbol{u}_{EKF,k}\ \right)$$

$$\boldsymbol{K}_{EKF,k} = \boldsymbol{P}_{EKF,k-1}\boldsymbol{H}_{EKF,k}^{T}\left(\boldsymbol{H}_{EKF,k}\boldsymbol{P}_{EKF,k-1}\boldsymbol{H}_{EKF,k}^{T} + \boldsymbol{R}_{EKF}\right)^{-1} \qquad (3.14)$$

$\boldsymbol{R}_{EKF}$ is a constant (unchanging) matrix representing the expected covariance of the measurements and the $\boldsymbol{H}_{EKF}$ function is the matrix in the relationship in equation (3.12). The subscript $k-1$ on a tensor indicates using the values from the previous Kalman filter update of that tensor. The subscript k indicates the values for the current update of that tensor. Once it's computed in (3.14), the Kalman gain matrix $\boldsymbol{K}_{EKF,k}$ for this measurement update is used to compute the updated state estimate $\boldsymbol{x}_{EKF,k}$ and the updated covariance matrix $\boldsymbol{P}_{EKF,k}$, as in equations (3.15) and (3.16), respectively.

$$\boldsymbol{x}_{EKF,k} = \boldsymbol{x}_{EKF,k-1} + \boldsymbol{K}_{EKF,k}\left[\boldsymbol{y}_{EKF} - \frac{\partial \boldsymbol{H}_{EKF}}{\partial \boldsymbol{x}_{EKF}}\boldsymbol{x}_{EKF,k-1}\right] \qquad (3.15)$$

$$\boldsymbol{P}_{EKF,k} = \left(I - \boldsymbol{K}_{EKF}\frac{\partial \boldsymbol{H}_{EKF,k}}{\partial \boldsymbol{x}_{EKF,k}}\right)\boldsymbol{P}_{EKF,k-1} \qquad (3.16)$$

The matrix $\frac{\partial \boldsymbol{H}_{EKF}}{\partial \boldsymbol{x}_{EKF}}$ is the Jacobian of the non-linear measurement relationship $\boldsymbol{H}_{EKF}$in equation (3.12). At this point, both the updated minimum error variance estimate for the states $\boldsymbol{x}_{EKF,k}$ and the updated error covariance matrix $\boldsymbol{P}_{EKF,k}$are saved off for the next (measurement or time) update.

3.3.2 Time Update

The procedure that is followed each time a state prediction update can be made using relationship (3.11) is called the time update. While the measurements $\boldsymbol{y}_{EKF}$ do not have to be available, the inputs $\boldsymbol{u}_{EKF}$ do need to be available. Typically, inputs $\boldsymbol{u}_{EKF}$ are not available continuously, but if equation (3.11) can be evaluated continuously, a time interval is chosen so that a balance between processor usage and estimator performance is achieved. In either case of discrete or continuous inputs, the time update is performed many times in between each measurement update. At each of these time updates, the state estimate is predicted from the measurement relationship in equation (3.11), using a simple first-order integration scheme as depicted by equation (3.17).

$$\boldsymbol{x}_{EKF,k} = \boldsymbol{x}_{EKF,k-1} + \frac{T_s}{N}\, \boldsymbol{F}_{EKF}(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k) \qquad (3.17)$$

The index $k - 1$ indicates to use the information from the most recent previous time or measurement update. If time and measurement updates occur at the same time, a time update is always performed first, and then the k indexed output of the time update becomes the $k - 1$ information used in the measurement update. Note that with a nonlinear EKF, the above integration to estimate the state vector $\boldsymbol{x}_{EKF,k}$ can be run multiple times to reduce errors due integration errors from nonlinearities in the state propagation. The variable N in equation (3.17) is the total number of iterations and T_S is the time that has elapsed since the last measurement update. Inherently, there is trade space between processor consumption and performance of the time update.

Now, to update the covariance matrix at each time update, a matrix is created that relates the states of the previous time step $\boldsymbol{x}_{EKF,k-1}$to the states of the current time step $\boldsymbol{x}_{EKF.k}$. This matrix is called the state transition matrix Φ, and is derived using the relationship $\boldsymbol{F}_{EKF}$ from equation (3.11) as depicted in equation (3.18).

$$\boldsymbol{x}_{EKF,k} = \Phi_{\mathrm{EKF,(k,k-1)}}\boldsymbol{x}_{EKF,k-1} \qquad (3.18)$$

$$\dot{\boldsymbol{\Phi}}_{\mathrm{EKF,(k,k-1)}} = \frac{\partial \boldsymbol{F}_{EKF}(\boldsymbol{x}_{EKF,k},\boldsymbol{u}_{EKF,k})}{\partial \boldsymbol{x}_{\boldsymbol{EKF}}}\boldsymbol{\Phi}_{\mathrm{EKF,(k,k-1)}} \qquad (3.19)$$

$$where \quad \boldsymbol{\Phi}_{EKF,(k-1,k-1)} = \boldsymbol{I} \qquad (3.20)$$

The subscripts (a, b) on the state transition matrix $\boldsymbol{\Phi}_{EKF}$ designates that the matrix transitions the state vector $\boldsymbol{x}_{EKF}$ from update b to update a, like the transition matrix in (3.18) maps the previous $k-1$ update to the current update k. Equation (3.19) could be solved analytically or numerically. A numerical solution would require linearizing the state dynamics of $\boldsymbol{F}_{EKF}$ in equation (3.11) about the updated state vector $\boldsymbol{x}_{EKF,k}$ that was determined in equation (3.17). With a solution for the state transition matrix, like is obtained in state estimation method 2, the state error covariance matrix can be updated via equation (3.21).

$$\boldsymbol{P}_{EKF,k} = \boldsymbol{\Phi}_{\mathrm{EKF,(k,k-1)}}\boldsymbol{P}_{EKF,k-1}\boldsymbol{\Phi}^{T}_{\mathrm{EKF,(k,k-1)}} + \boldsymbol{Q}_{EKF} \qquad (3.21)$$

The matrix $\boldsymbol{Q}_{EKF}$ is a tunable parameter that represents the expected covariance of the state derivatives as determined by relationship (3.11).

However, solutions for the state transition matrix are often not easily obtainable. If the state transition matrix is not obtained, a linearized solution of (3.19) could be used to update the covariance matrix. . The first term of the Taylor series expansion of the solution to the differential equation (3.19) is derived in [4], and it results in the ability to update the covariance matrix as is described in equation (3.22).

$$\boldsymbol{P}_{EKF,k} = \boldsymbol{P}_{EKF,k-1} + \frac{T_s}{N}\left(\left[\frac{\partial \boldsymbol{F}_{EKF}(\boldsymbol{x}_{EKF,k},\boldsymbol{u}_{EKF,k})}{\partial \boldsymbol{x}_{\boldsymbol{EKF}}}\right]\boldsymbol{P}_{EKF,k-1}\left[\frac{\partial \boldsymbol{F}_{EKF}(\boldsymbol{x}_{EKF,k},\boldsymbol{u}_{EKF,k})}{\partial \boldsymbol{x}_{\boldsymbol{EKF}}}\right]^{T}\right) + \boldsymbol{Q}_{EKF} \qquad (3.22)$$

It is important to note that this is a linearized solution of the nonlinear relationship between two states, as depicted by (3.18), and so like the time update for the state vector, updating the covariance matrix can also be iterated a number of times to improve the estimate. The variable N in equation (3.22) is the iteration number, and the matrix $\boldsymbol{Q}_{\boldsymbol{EKF}}$ is again the expected covariance of the error in relationship (3.11). Performing this calculation is essentially the same a performing a numerical integration to determine the state transition matrix that is needed for (3.21), but both forms of updating the covariance matrix (3.21) and (3.22) are used in this document.

4 Aircraft States Needed for Autopilots

Before attacking any state estimation problem, it is critical to first know what states needs to be gathered. Ideally, a state estimator would be able to accurately reproduce every state of an aircraft. . However, full state estimation is often impossible given a finite set of sensors like the ones that are typically available on a small UAV. Further, estimating more states than are needed can be excessively costly and computationally intensive. This paper will focus on capturing only the aircraft states that are necessary for autopilot control of small UAVs.

4.1 Scoping the Estimation Problem

True total information on the kinematic state of an object would involve a description of the continuous motion of an object, such that all orders of derivatives of the motion can be definitively described. However, a description of the continuous motion of an aircraft is not possible in a real world system, as the true motion of the aircraft can never be known ahead of time, and digital sensors that provide information are discrete. Thus, state estimation for aircraft is performed as determining the current states of an aircraft a given point in time. While it could be desired to obtain all of the time derivatives of the motion of an aircraft, second order information is all that is required for most UAV applications. In this research, the collection of states in (4.1) will be considered the complete set of states that could be desired to be gathered.

$$\boldsymbol{x}_{complete} = \{\boldsymbol{p}_{B/N},\ \boldsymbol{V}_{B/N},\ \boldsymbol{a}_{B/N},\ \boldsymbol{\psi},\ \boldsymbol{\omega}_{B/N},\ \boldsymbol{\alpha}_{\boldsymbol{B/N},} \boldsymbol{V}_{w}\} \tag{4.1}$$

The vectors in $\boldsymbol{x}_{complete}$ describe the inertial position, velocity, and acceleration of the aircraft, the angular position, angular velocity, and angular acceleration of the aircraft relative to the inertial frame, and the velocity of the airflow around the aircraft. It can be noted that any combination of two of the velocity vectors $\boldsymbol{V}_{B/N}$, $\boldsymbol{V}_a$, and $\boldsymbol{V}_w$ can be used in (4.1) to fully describe the velocities of the aircraft and the air, as the three are related through equation (3.10). Deeper information could be included in this list, such as the angular acceleration of the aircraft, the rate of change of wind velocity, or even the rotational motion of the air. However, deeper information is not necessary for autopilot control, as will be shown throughout this paper. Deeper information is also often very difficult, expensive, and impractical for small UAVS. While there are a number of different existing autopilot algorithms that each require a different states of states, (4.1) is considered a superset of the states that all common small UAV autopilots would use to control a small UAV.

This is a superset because some of the states listed in (4.1) are not necessary for autopilots to perform acceptably well. Much work has been done in estimating the states that makeup the wind vector $\boldsymbol{V}_w$, such as in [10], [11], and [12]. However, full 3-dimensinoal measurements of the wind vector, even in windy conditions, are not necessary for an autopilot to function. They would be needed for an autopilot that employs an energy harvesting algorithm, or for a mission to gather wind information, but is not strictly necessary for autopilots to maintain basic stable flight, as will be shown shortly. While it is not important to recover full information about the wind vector, it is important to obtain a rough measure of the airspeed of the aircraft for the purpose of ensuring the aircraft doesn't travel too slow in large tailwinds and stall. With just inertial velocity information, airspeed has to be assumed close to the inertial velocity. Such an assumption can easily lead to stall because it is not uncommon for small aircraft to experience wind speeds close to the speed of their inertial velocity. Airspeed sensors that can remedy this problem are readily available today for under $50 USD.

Nonetheless, the lack of a need for full wind vector $\boldsymbol{V}_w$ stems from the aerodynamic behavior of small UAVs. Small UAVs are lightweight and thus naturally quickly align themselves into the airflow. This results in relatively small disturbances in angles of attack and angles of sideslip when compared to the disturbances seen by large scale aircraft. Small UAVs are also typically designed to be aerodynamically stable, so rapid control authority is not necessary to maintain attitude. In contrast, some large modern maneuver aircraft, such as the Lockheed Martin F-22, are aerodynamically unstable and require significant and rapid control authority to maintain attitude.

This behavior common to most small UAVs is often used to actually represent the wind vector in terms of other estimated states. By assuming that sideslip is zero and that the angle of attack is equivalent to the pitch angle, equation (3.10) for the airspeed vector $\boldsymbol{V}_a$ can be simplified to equation (4.2).

$$ {}^{\boldsymbol{B}}[\boldsymbol{V}_a] = V_a \begin{bmatrix} \cos\alpha\cos\beta \\ \sin\beta \\ \sin\alpha\cos\beta \end{bmatrix} \xRightarrow{\beta=0,\alpha=\theta} {}^{\boldsymbol{B}}[\boldsymbol{V}_a] = V_a \begin{bmatrix} \cos\theta \\ 0 \\ \sin\theta \end{bmatrix} \tag{4.2} $$

This assumes that the airspeed magnitude V_a is either directly measured or estimated. As discussed above, obtaining the airspeed magnitude is necessary to avoid stall and thus V_a should always be available in some fashion. If the inertial velocities are also estimated, the airspeed vector in equation (3.9) can be substituted with equation (4.2) and rearranged to obtain equation (4.3), which is a rough estimate of the wind vector $\boldsymbol{V}_a$.

$$ {}^{\boldsymbol{B}}[\boldsymbol{V}_w] = {}^{\boldsymbol{B}}[\boldsymbol{V}_{B/N}] - V_a \begin{bmatrix} \cos\theta \\ 0 \\ \sin\theta \end{bmatrix} \tag{4.3} $$

This simplification is the basis for showing that estimating the angle of attack and angle of sideslip are not necessary.

Both the autopilot and the two state estimation methods described and simulated in this document do not estimate the full wind vector $\boldsymbol{V}_w$. Instead, they use an airspeed sensor and the assumption in (4.2) where air data is required. The validity of this assumption that is discussed in the prior section is demonstrated here using a simulation of the RECUV Tempest small UAV, employing a total energy controller that processes altitude, course angle rate, and airspeed commands.

4.2 Overview of the Autopilot

While the focus of this document is on state estimation, an autopilot is needed to control the flight of the aircraft in the simulations so that analysis of the state estimation methods can be done under different benchmark maneuvers. The autopilot that is used to perform this task is a total energy based controller, which is a common controller to use for small UAVs. The specific controller used here takes commanded airspeed V_{a_c}, altitude h_c, and course angle rate $\dot{\chi}_c$ and commands thrust, aileron, rudder, and elevators. It does this by following the algorithm in [4] and [13].

Table 4.1: Total Energy Autopilot Algorithm Description

Step	Description
1	Calculate a commanded bank angle $\phi_c$using a Proportional, Integral, and Derivative (PID) controller that process the error between the commanded course angle rate and the course angle rate that is necessary for the desired turn, which is in equation (4.4). $\phi_c = PID(\dot{\chi}_c - \dot{\chi}) = PID\left(\dot{\chi}_c - \frac{g}{V_g}\tan\phi\right)$ (4.4) This equation assumes that the course angle rate is equivalent to $\frac{g}{V_g}\tan\phi$. This is only the case in zero wind. The effects of this will be seen in the simulation results that follow.

2	Calculate aileron deflection δ_a to control the bank angle ϕ using a PID control that process the bank error, as in equation (4.5). $\delta_a = PID(\phi_c - \phi)$ (4.5)
3	Determine commanded pitch angle θ_c from a PID controller on the airspeed error, as in equation (4.6). $\theta_c = PID\left(V_{a_c} - V_a\right)$ (4.6) This essentially just increases pitch to decrease airspeed and decrease pitch to increase airspeed. This means that in order for the aircraft to climb, there must be excess airspeed. This observable is seen in all of the simulations in this document.
4	Calculate elevator deflection from commanded pitch using a PID controller, as in equation (4.7) $\delta_e = PID(\theta_c - \theta)$ (4.7)
5a	If the aircraft is within 10 meters of the ground, assume plane is in takeoff, and set the commanded thrust to a constant as in equation (4.8). $\delta_t = 0.8$ (4.8) Note that there is no landing mode for this autopilot.
5b	If the aircraft is more than 10 meters off the ground, use a total energy approach to command thrust. This total energy calculation is uses a PID controller on the error between the

	commanded and current sum of the kinetic and gravitational potential energies, as depicted in equation (4.9). $\delta_t = PID\left(g(h_c - h) + \frac{1}{2}\left(V_{a_c}^2 - V_a^2\right)\right)$ (4.9)
6	Calculate the rudder deflection that enforces zero sideslip. This could be done with a PID controller against estimated sideslip angle β, but this autopilot recognizes that β is often not available and assumed to be zero as in equation (4.2). So, the goal in this case is to instead zero the specific side force, which is the acceleration along the body Y axis a_y. The sum of this acceleration component and the component of the gravity vector along the body Y axis is provided by the state estimator as the accelerometer measurement y_{a_y}. This yields equation (4.10). $\delta_r = k_{p,side\ force} a_y = k_{p,side\ force}\left(y_{a_y} - g\cos\theta\sin\phi\right)$ (4.10) Equation (4.10) depicts how a proportional controller is used to zero the specific side force, or in other words zero the acceleration along the body Y axis. On a longitudinally symmetric aircraft, this also zeroes the sideslip angle β, even in wind. It can be noted that as discussed earlier, small UAVs typically quickly align themselves into the airflow. This autopilot employs this controller as an additional measure to ensure sideslip is zero and thus coordinated turns are forcibly maintained. Judging the necessity of a small UAV's autopilot to actively control sideslip is outside of the scope of this research.

The gains for the PID controller in the algorithm were designed such that they achieve acceptable closed loop tracking of their states.

So in summary, this specific autopilot that is used throughout this document requires the inputs listed in equation (4.11).

$$\boldsymbol{u}_{autopilot} = \left[h\ p\ q\ \phi\ \theta\ \ V_a\ V_g\ y_{a_y}\right] \tag{4.11}$$

The states in Equation (4.11) are the states that this paper will focus on estimating, as they are what is needed for autopilot control. Note that this is still a subset of (4.1), as the airspeed is a function of the state vectors in (4.1), as depicted by (4.12), and the ground speed V_g is also a function of the state vectors in (4.1), as depicted by (4.13).

$$V_a = \|\boldsymbol{V}_a\| = \left\|\boldsymbol{V}_{B/N} - \boldsymbol{V}_w\right\| \tag{4.12}$$

$$V_g = f\left(\boldsymbol{V}_{B/N}, \boldsymbol{\psi}\right) = \left\| {}^{\boldsymbol{N}}\boldsymbol{V}_{B/N}\ [1 \quad 1 \quad 0]\right\| = \left\|\boldsymbol{R}_{NB}\ {}^{\boldsymbol{B}}\boldsymbol{V}_{B/N}\ [1 \quad 1 \quad 0]\right\| \tag{4.13}$$

Now, a distinction should be made here that these are the states needed for autopilot control and not overall navigation. A navigation algorithm is what takes user commands, such as destination, course angle, and/or altitude, and turns them into autopilot commands, which in this case are altitude, course angle $rate$, and airspeed. Analyzing accurate navigational layer control in windy conditions is outside of the scope of this research because state estimation algorithms inherently provide the information necessary for navigation. All that is really needed for navigation is inertial position and velocity estimates, and those can be obtained at a low rate directly from a GPS receiver. The only complexity of navigation that should be mentioned here is that the navigation layer needs a calibrated closed loop controller on course angle χ, as the autopilot takes course rate $\dot{\chi}$ commands. As will be explained, method 1 estimates the course angle for this purpose, and if the first two inertial frame components of the inertial velocity vector are available, like in method 2, the course angle can be determined via equation (4.14).

$$\chi = \tan^{-1}\left(\frac{{}^{N}[\boldsymbol{v}_{B/N}][1 \quad 0 \quad 0]}{{}^{N}[\boldsymbol{v}_{B/N}][0 \quad 1 \quad 0]}\right) \tag{4.14}$$

Note that the equation (4.14) must be calculated using a quadrant-safe inverse tangent function. Course angle could be determined directly from GPS, but using a state estimator's output provides a higher sample rate that the navigation layer can take advantage of.

4.3 Simulation Setup

Before focus can be turned to the state estimation methods, it must first be shown through simulation that this autopilot is able to effectively control a small UAV with the limited suite of state information listed in (4.11).

To properly assess the autopilot's tracking of its commanded altitude, course angle rate, and airspeed, assessments must be made in flight regimes that small UAV autopilots and their state estimators are expected to be used in. To ensure each of these flight regimes is tested, the simulations throughout this document employ a number of benchmark maneuvers, which are described here.

4.3.1 Benchmark 1: Straight and Level Flight

The first benchmark maneuver that is used in the simulations is straight and level flight, with initial conditions equal to the commanded course angle rate, altitude, and airspeed, as depicted in (4.15).

$$h_0 = h_c = 100\ m, \dot{\chi}_c = \dot{\chi} = 0, V_{a_0} = V_{a_c} = 15\ m/s \tag{4.15}$$

In all of the benchmark maneuvers in this document, the initial conditions in (4.16) are used.

$$p_0 = q_0 = r_0 = \phi_0 = \theta_0 = \psi_0 = p_e = p_d = 0 \tag{4.16}$$

An illustration of the movement of the simulated aircraft is in Figure 4.1.

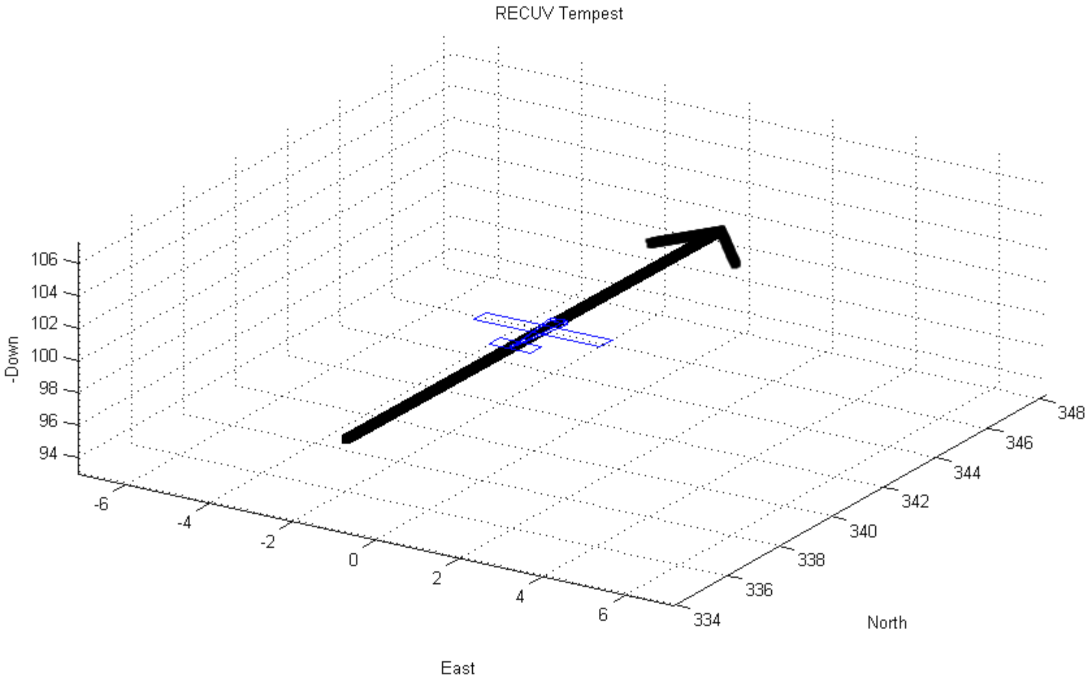


Figure 4.1: Straight and Level Flight Illustration

4.3.2 Benchmark 2: Coordinated Turn

The second benchmark maneuver that is used in the simulations is a coordinated turn. A coordinated turn is a turning maneuver that involves elevator, aileron, and rudder deflections such that no acceleration is experienced in the lateral Y axis direction of the aircraft. Such a maneuver is illustrated in Figure 4.2.

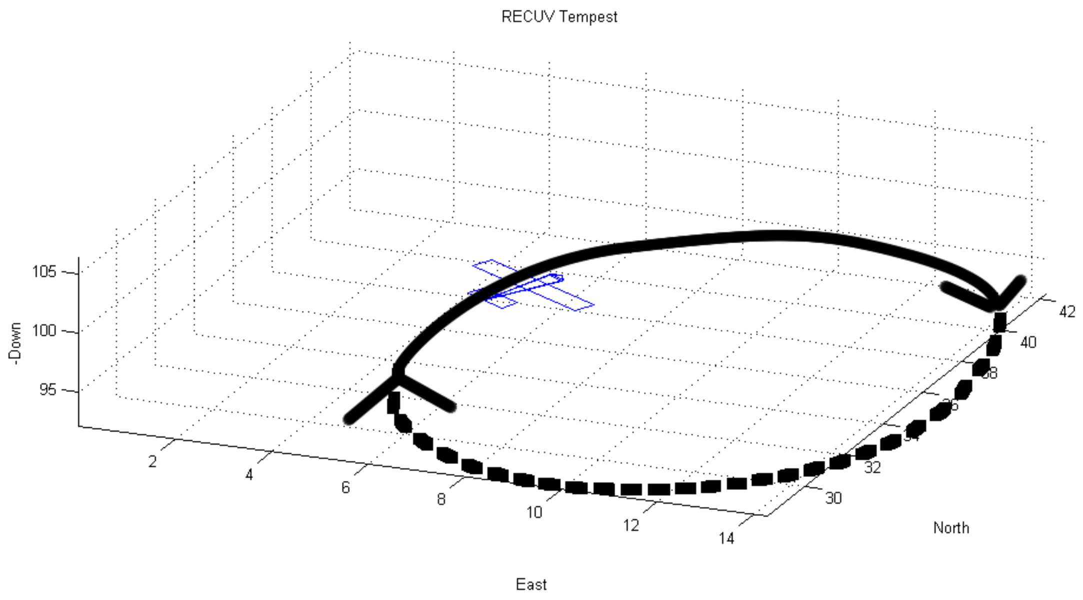


Figure 4.2: Coordinated Turn Illustration

The simulations that are performed in this document for benchmark 2 are 90 seconds in duration, which is the time required for 2.5 complete circles at a constant course angle rate of 10 degrees per second. The autopilot enforces a coordinated turn through controlling the lateral specific force to zero, as in equation (4.10), commanding roll angle required for a coordinated turn, as in equation (4.4), and by commanding pitch through commanded airspeed, as in equation (4.6). It can be noted that the actual equation for required roll angle for a coordinated turn is equation (4.17).

$$\dot{\chi} = \frac{g}{V_g} \tan \phi \cos(\chi - \psi) \tag{4.17}$$

In the absence of wind, the course angle χ is equal to the yaw angle ψ, which transforms equation (4.17) into equation (4.18).

$$\dot{\chi} = \frac{g}{V_g} \tan \phi \tag{4.18}$$

Since the autopilot still uses error in commanded airspeed instead of error in commanded course rate via (4.17) or even (4.18) in a coordinated turn, there is an observable lag in achieving a coordinated turn in the simulations when a change in commanded course angle is applied. This lag is characterized by an oscillation in pitch that is short in duration, which will be shown in simulation shortly. To illustrate this behavior, simulations are performed for coordinated turns by initially having the aircraft at the same straight and level initial conditions as benchmark 1 in (4.15) and (4.16), but with a commanded course angle rate $\dot{\chi}_c$ of 10 degrees per second.

4.3.3 Benchmark 3: Wings Level Ascent

The third benchmark that is used in this document is a wings level ascent. For this benchmark, the same commanded altitude, airspeed, and course angle rate are the same as in the benchmark 1 (4.15). The initial airspeed and course angle rate are also the same as (4.15), but the initial altitude is taken to be zero. This setup is summarized in (4.19).

$$h_0 = 0,\ h_c = 100\,m,\ \dot{\chi}_0 = 0,\ \dot{\chi}_c = 0,\ V_{a_0} = V_{a_c} = 15\,m/s \tag{4.15}$$

This essentially simulates takeoff, but beginning at the full commanded airspeed so that the aircraft doesn’t stall, as rolling on the ground is not part of this simulation. Simulations of this benchmark maneuver are stopped when the aircraft becomes within 1 meter of its commanded altitude. Figure 4.3 illustrates this maneuver.

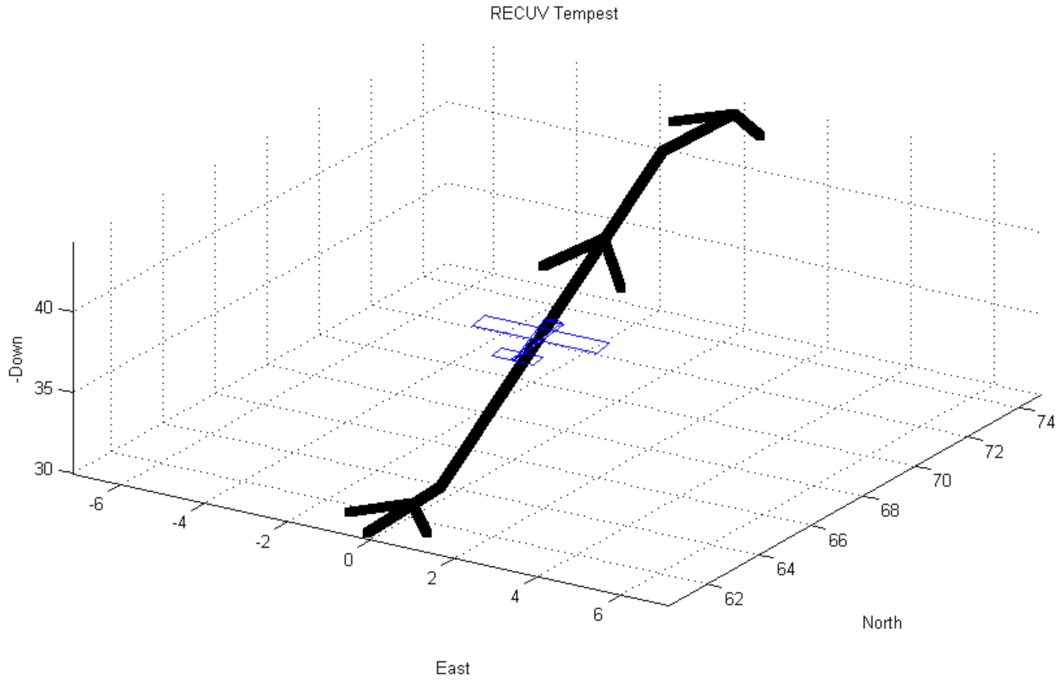


Figure 4.3: Wings Level Ascent Illustration

Simulation of all three pieces of the ascent – takeoff, climb, and cruising altitude approach – enables observations and insights that are discussed later.

4.3.4 The Dryden Wind Model

This document simulates the wind necessary for analysis using the Dryden wind model. The Dryden wind model is used throughout department of defense projects as a standard by which aircraft systems should be designed to when considering wind. The Dryden wind model that is used in this document generates wind noise by summing sinusoidal signals with frequencies of relative power spectral densities that are given by military specification MIL-F-8785C. The equations for the spectral density take two parameters – scale length and the wingspan of the aircraft that's being simulated. These two parameters provide how winds at high altitude should be scaled to those at low altitude, and the power of disturbances on the aircraft's angular rates, respectively. Note that for these simulations, high altitudes, which are on the order of 2000 feet above ground level, are never reached, so the scale length has negligible impact on the wind speeds in our simulations.

The simulations in this document use both outputs of the Dryden wind model – wind speed disturbances and angular rate disturbances. The Dryden wind model uses a frozen turbulence field, so the wind speed output of the Dryden model must be added to the mean wind speed to obtain the actual wind speed value. This is depicted by (4.16a) below.

$$\begin{bmatrix} w_n \\ w_e \\ w_d \end{bmatrix} = \begin{bmatrix} w_n \\ w_e \\ w_d \end{bmatrix}_{mean} + \begin{bmatrix} \delta w_n \\ \delta w_e \\ \delta w_d \end{bmatrix}_{Dryden}$$

This wind velocity output is subtracted from the inertial velocity to obtain the airspeed that is used in the linearized aerodynamic model of the Tempest UAV that the simulation uses. This subtraction is in accordance with the wind vector convention in equation (3.9). Second, the rate disturbance output of the Dryden wind model is added to the angular rates of the aircraft after the aerodynamic forces and moments are used to calculate the rigid body dynamics of the aircraft. This is depicted in equation (4.16b).

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{model} + \begin{bmatrix} \delta p \\ \delta q \\ \delta r \end{bmatrix}_{Dryden} \qquad (4.16b)$$

These wind-augmented angular rates are then used as the input to the rigid body dynamics equations in the simulations.

The simulations throughout this document use a mean wind speed of 7 meters per second and a turbulence probability of exceedance of high-altitude intensity measurement set to 10^-5. That level of turbulence is classified by the Dryden model as severe. Figure 4.4 below shows wind and rate disturbances that result from this wind.

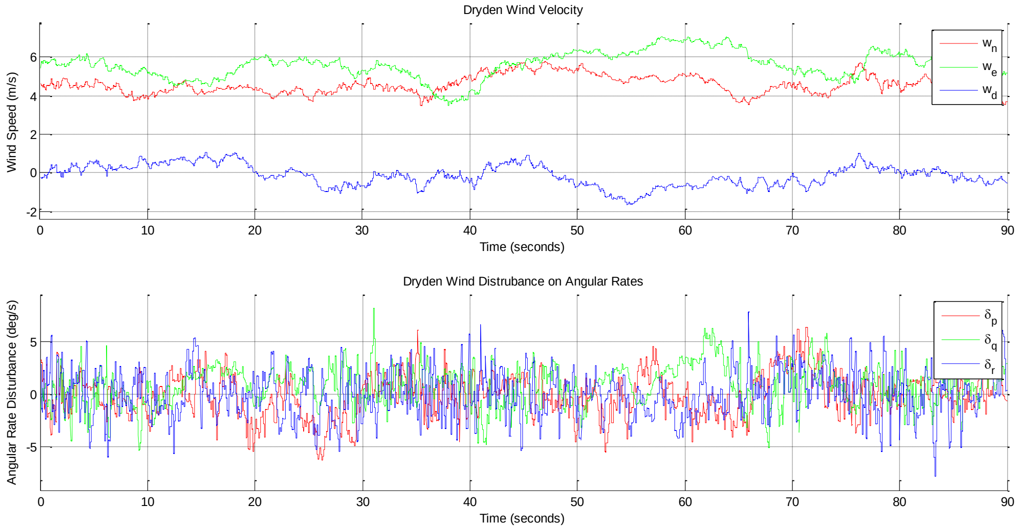


Figure 4.4: Example Dryden Wind Model Simulation

Statistics on the absolute value of the wind and angular rate disturbance values presented in Figure 4.4 are shown in Table 4.2.

Table 4.2: Statistics of the Absolute Value of the Dryden Wind Components

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
w_n	4.575743	0.473044	4.575743	0.473044	3.469525	5.790123
w_e	5.544791	0.709703	5.544791	0.709703	3.477738	7.082888
w_d	-0.12786	0.544442	0.457334	0.321883	0.000002	1.692289
δp	-0.13099	2.025574	1.618597	1.224839	0.000012	6.35785
δq	0.79809	1.929265	1.694187	1.220133	0.000042	8.172666
δr	0.030636	2.295561	1.822656	1.395865	0.000026	7.868352

The benchmark maneuvers that were previously discussed all use a commanded airspeed $V_{a_c}$of 15 m/s. The wind speeds shown in Figure 4.4 become as large as about half of this value. Thus, as also previously discussed, it is imperative that the state estimators used here employ an airspeed sensor to ensure that the aircraft does not stall. Also, the Beaufort wind scale [14] classifies a wind speed of 7 m/s as a “Moderate Breeze” at which tree branches begin to move. This is a wind speed that small UAVs will often encounter. It will be seen shortly that even this moderate amount of wind is sufficient to highlight the strengths and weaknesses of the two state estimation methods analyzed here.

Moreover, extreme winds not a part of this simulation. The scope of this research is limited to the wind that UAVs might typically see outside of severe weather anomalies, as successfully controlling aircraft in such conditions involves more complex autopilots and navigational algorithms that are not used here.

4.4 Sensors

All of the simulations in this document employ an identical suite of sensors. All simulations involve the use of a MEMS gyro and a MEMs accelerometer, which comprise the IMU sensor system. The gyros provide the angular rates of the body fame relative to the inertial frame, and the accelerometers provide measured N frame accelerations along the body frame components, plus gravity. Also included is a GPS sensor, which provides measures for the position in the inertial N frame and the velocity in the inertial N frame. The accuracy of the sensors modeled used in the simulations are determined from the spec sheets of actual off the shelf sensors that may be used in UAVs. The IMU sensor models used in the simulation are set to closely match the following sensors:

- Gyroscope: Analog Devices ADXRS450
- Accelerometer: Analog Devices ADXL325

The GPS sensor model that is used in the simulations in this document have Gaussian noise of 0.21 meters in the north and east directions and .4 meters in the vertical direction, modelled as Gauss-Markov process with a time constant of 1100. The GPS sensor produces a non-time varying Gaussian error of 0.2 meters per second in the north, east, and down directions.

4.5 Demonstration of Autopilot Performance with Limited State Information

Now that the setup of the simulation has been discussed, a simulation can be performed to validate the notion that the autopilot input states in (4.11) are at least a superset of the states that need to be estimated for an autopilot to successfully pilot an aircraft in both windy and non-windy conditions. That is, it needs to be shown that this autopilot can successfully control the aircraft with the sates in (4.11). This by no means rules out the possibility that an autopilot could control an aircraft successfully in wind with fewer states. But by illustrating that the states in (4.11) are sufficient for autopilot control of an aircraft in windy conditions, state estimation methods that obtain those state can be said to be sufficient to enable autopilot control of an aircraft in windy conditions.

So, the performance of the autopilot discussed here will be measured by its error against its commanded inputs (airspeed, altitude, and course angle rate), under the three different benchmarks that were discussed. To demonstrate the effectiveness of the autopilot in these simulations, the autopilot uses the true values of the states in (4.11) rather than through a state estimation method. An analysis of the sensitivity of the autopilot's performance to errors in the different states is beyond the scope of this research, as the purpose of this research is solely to measure the performance of the state estimation methods' sensitivity to wind, and not an autopilot's sensitivity to state errors. Further, an analysis of the sensitivity the autopilot to the states using the autopilot here would not necessarily be representative of autopilots as a whole. Nonetheless, a limited sensitivity analysis using constant state errors under benchmark 2 is performed in section 4.6.

4.5.1 Autopilot Performance under Benchmark 1

Benchmark 1 is straight and level flight. Figure 4.5 below shows the error of commanded course angle rate, $\dot{\chi} - \dot{\chi}_c$, the error in the commanded airspeed $V_a - V_{a_c}$, and error in the commanded altitude $h -$

h_c. This is the sign convention for error that is used throughout this paper – the desired value subtracted from the actual value. This means that a positive error represents the actual value being higher than the desired value. For example, here, a positive airspeed error represents the actual airspeed being higher than the commanded airspeed.

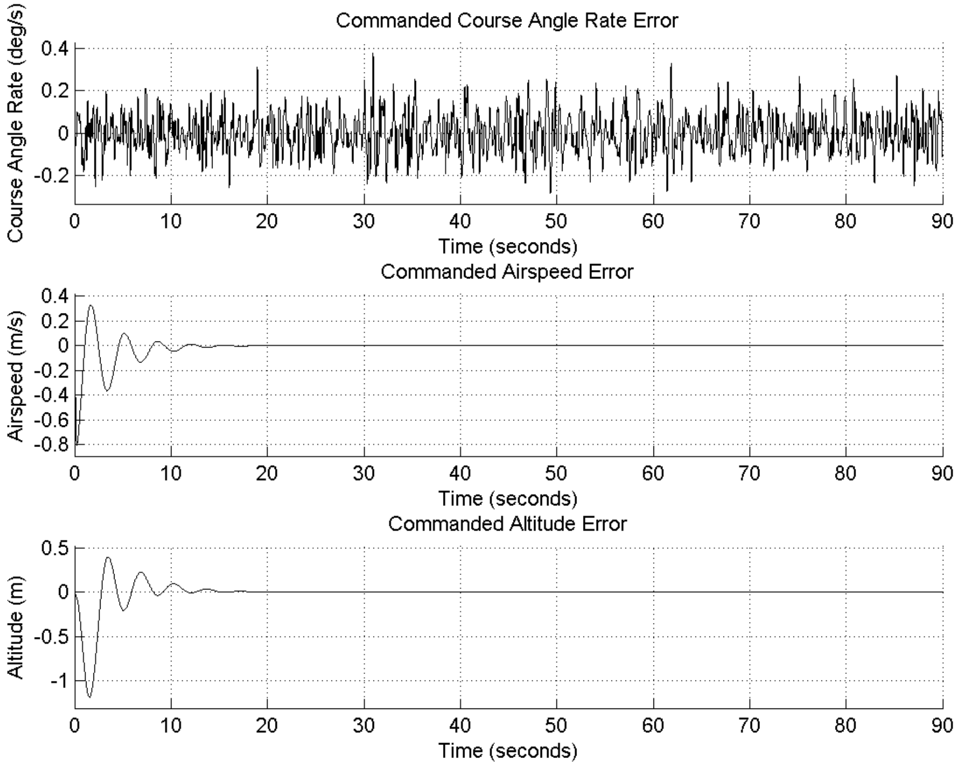


Figure 4.5: Command Tracking Error for Benchmark 1 in Zero Wind

Statistics of the error shown in Figure 4.5 are provided in Table 4.3. The subscript abs in Table 4.3 indicates that the statistic is calculated on the absolute value of the corresponding state error. The absence of the subscript abs denotes that statistic is calculated on the raw state error, without first performing an absolute value operation. This convention will be used throughout this document.

Table 4.3: Command Tracking Error Statistics for Benchmark 1 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}

Course Angle Rate (deg/sec)	-0.00038	0.095553	0.076424	0.057351	0	0.376379
Airspeed (meters/sec)	-0.00872	0.076007	0.018394	0.074261	0	0.803293
Altitude (meters/sec)	-0.01102	0.140657	0.032702	0.137245	0	1.179383

Figure 4.5 and Table 4.3 show that the autopilot closely tracks all three command inputs. Relating to the standard deviation of the raw error commands, the autopilot typically keeps the course angle rate within 0.1 deg/s, the airspeed with 0.08 m/s, and the altitude within 0.q5 m. It can be noticed that although the initial airspeed and altitude of the vehicle are set to the commanded airspeed and altitude, the autopilot induces a small deviation from the commanded airspeed and altitude for a short time. These deviations are small, roughly 0.5 m/s in airspeed and roughly 1 m in altitude, and are due to the fact that the initial pitch angle is set to zero rather than a slightly positive number, which is needed to maintain the commanded altitude. Also, the continual oscillations in course angle rate are small, and are due to the non-zero noisy input of the accelerometer that measures side force, y_{a_y}. The accelerometer that measures acceleration along the body Y axis is passed through a low pass filter, as described later, but its output is still noisy due to the high cutoff frequency that is used in the low pass filter. This measurement is used by the autopilot to command rudder deflection, artificially creating sideslip and thus a non-zero course angle rate. If y_{a_y} is taken to be zero, simulations not illustrated here have produced exactly zero course angle rate error.

Overall, the autopilot performs remarkably well for straight and level flight with zero wind. Next, Figure 4.6 below is a plot of the autopilot's performance for straight and level flight under the light wind Dryden model illustrated in section 4.3.4.

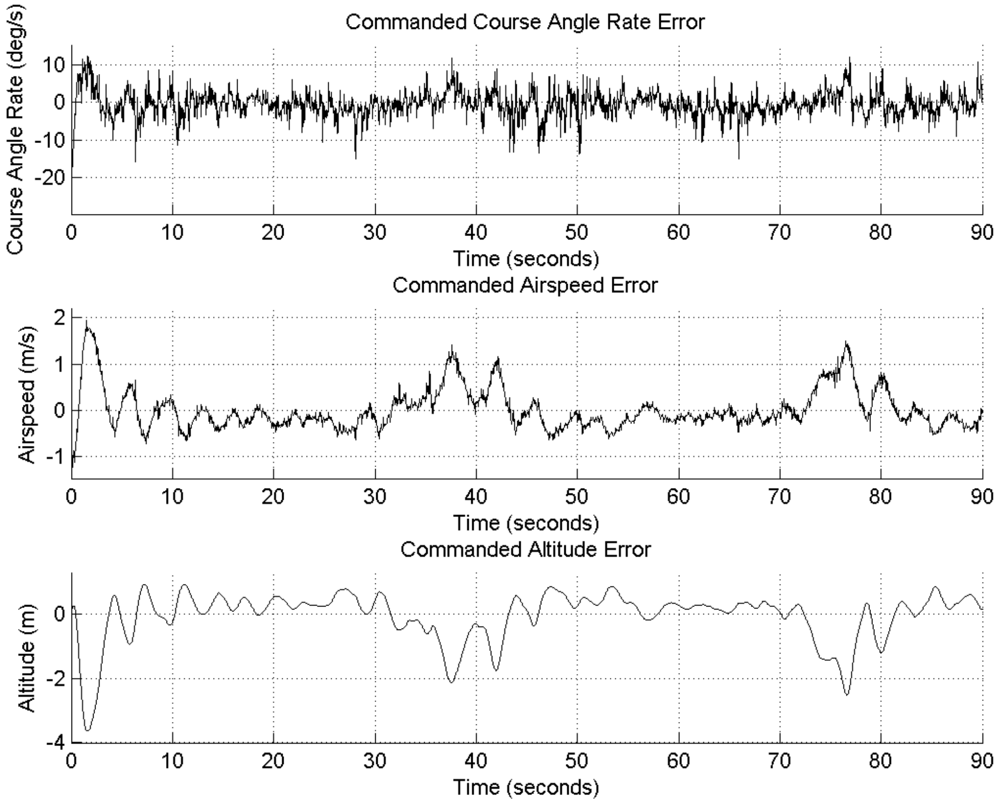


Figure 4.6: Command Tracking Error for Benchmark 1 in Light Wind

When Figure 4.6 is compared to Figure 4.5, it can be seen that the wind induces a definite measurable error in the state tracking. Statistics for the error in the autopilot's tracking of the commanded states is quantified by Table 4.4 below.

Table 4.4: Command Tracking Statistics for Benchmark 1 in Light Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
Course Angle Rate (deg/sec)	0.012568	0.991571	0.593818	0.794174	0.000035	24.80099
Airspeed (meters/sec)	-0.00944	0.225685	0.179037	0.137713	0.000001	1.169365
Altitude (meters/sec)	-0.0122	0.35684	0.280212	0.221261	0	1.36271

When the statistics for light wind in Table 4.4 are compared to the statistics for zero wind in Table 4.3, it can be seen that the command tracking error in zero wind is relatively negligible. The average values of the absolute values of the error in light wind for course angle rate, airspeed, and altitude, are much greater than the errors in light wind. This, along with the oscillations seen in Figure 4.6, are caused by both the effects of the wind velocity on the aerodynamic forces and moments on the aircraft, and the angular rate disturbances that the Dryden wind model provides. This combination of perturbations causes changes in the course angle rate, airspeed, and altitude that the autopilot must correct for this. These statistics illustrate that the autopilot does a very good job of maintaining the commanded course rate, airspeed, and altitude, as all three values stay very close to their commanded values.

Now, before moving on, it is important to note that the wind observed in the simulations is not exactly identical to those in Figure 4.4. The Dryden wind model used here uses seed values for its random noise generator, but the Dryden model varies its frequency spectrum based on the ground speed and orientation of the aircraft to simulate a consistent wind field that the aircraft can fly through in different manners. Thus, the wind speeds and angular rate disturbances as seen by the aircraft are not always exactly the same.   They will only be the same if the aircraft maintains straight and level flight at a constant airspeed through the wind field. With an aircraft flying in wind, that is never the case, so the wind seen by the aircraft is slightly different in each simulation. To keep the discussion concise, plots and statistics of the exact wind field are only included when they become relevant enough to the discussion.

4.5.2 Autopilot Performance under Benchmark 2

Next, the aircraft's ability to track its states is measured under a coordinated turn maneuver. Figure 4.7 shows the error between the commanded course angle rate, airspeed, and altitude states during a coordinated turn at a commanded 10 deg/s course angle rate in zero wind.

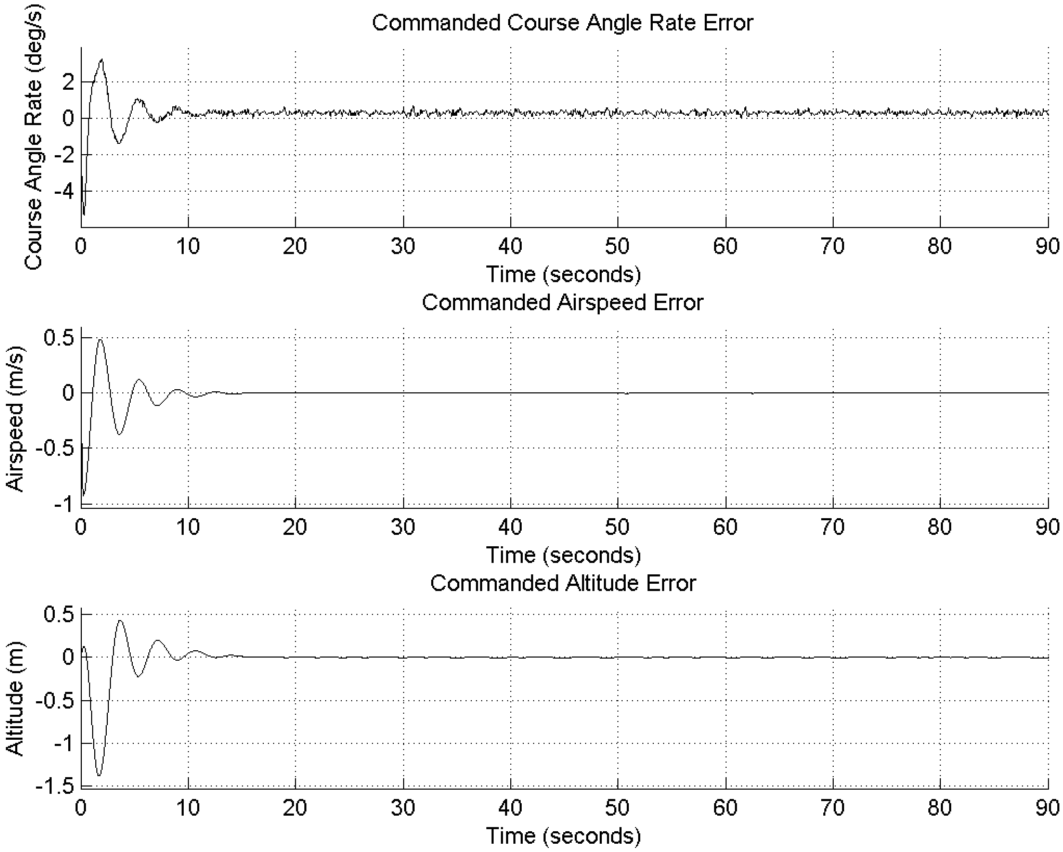


Figure 4.7: Command Tracking Error for Benchmark 2 in Zero Wind

Like in benchmark 1, it can be seen how an initial pitch angle of zero creates a small initial error in commanded airspeed and attitude. However, it also creates a small disturbance in the initial course angle rate. This simulation could be started once the aircraft is in full a coordinated turn, or the initial condition required for pitch to avoid these oscillations could be done, but it is not necessary to analyze the ability of the autopilot to maintain a coordinated turn. In fact, it also illustrates how the autopilot is also able to enter a coordinated turn. Table 4.5 below contains the statistics of the error in tracking the commanded states for this coordinated turn in zero wind.

Table 4.5: Command Tracking Statistics for Benchmark 2 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}

Course Angle Rate (deg/sec)	0.140568	0.49356	0.243822	0.451561	0.000064	5.440359
Airspeed (meters/sec)	-0.00737	0.093255	0.022611	0.090772	0	0.926243
Altitude (meters/sec)	-0.01329	0.15741	0.036695	0.153648	0	1.349807

Both Figure 4.7 and Table 4.5 show that the error in tracking these states is very small and that the autopilot tracks the commanded states for a coordinated turn very well.

Next, Figure 4.8 shows the error in the command tracking during a coordinated turn in Dryden wind.

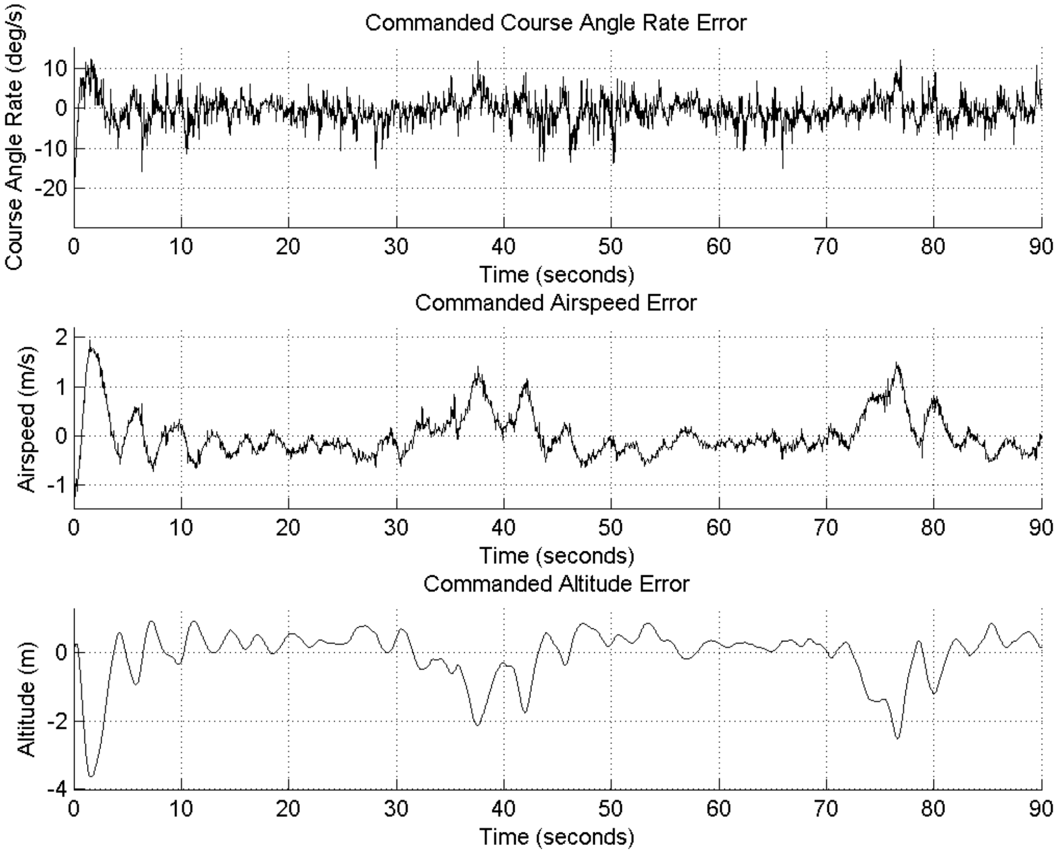


Figure 4.8: Command Tracking Error for Benchmark 2 in Light Wind

Figure 4.8 shows that like benchmark 1 in wind, there is significant oscillation in the error of the course angle rate, airspeed, and altitude. Statistics on the magnitude of this error oscillation is in Table 4.6.

Table 4.6: Command Tracking Statistics for Benchmark 2 in Light Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
Course Angle Rate (deg/sec)	0.355298	4.144099	2.773074	3.099569	0.000463	24.04694
Airspeed (meters/sec)	-0.05336	0.215181	0.111793	0.191441	0.000003	3.347057
Altitude (meters/sec)	-0.05581	0.208895	0.164445	0.140365	0	0.731389

The tracking results for airspeed and altitude for benchmark 2 in wind do not appear to differ much from the results for benchmark 1 in wind. However, the error in the course angle rate for benchmark 2 in light wind is about double the error in the course angle rate for benchmark 1 in light wind. While a coordinated turn is not always held during severe gusts, this increase in course angle rate error is best described by the course angle rate equation (4.17). Equation (4.17) shows that as course angle increases and the difference between course angle and yaw angle increases, the course angle rate becomes larger in a sinusoidal fashion and not a linear fashion. For example, a rate disturbance causing a 10 degree difference between the yaw angle and the course angle has a much larger effect on the course angle rate when the bank angle is large than when it is small. Regardless, even in wind, the autopilot manages to track its command course rate with a near zero mean error.

4.5.3 Autopilot Performance under Benchmark 3

The last benchmark test is a wings level ascent. This benchmark highlights some of the assumptions made in the state estimation methods that are described later. Figure 4.9 below shows the errors in the commanded states for a wings level ascent in zero wind.

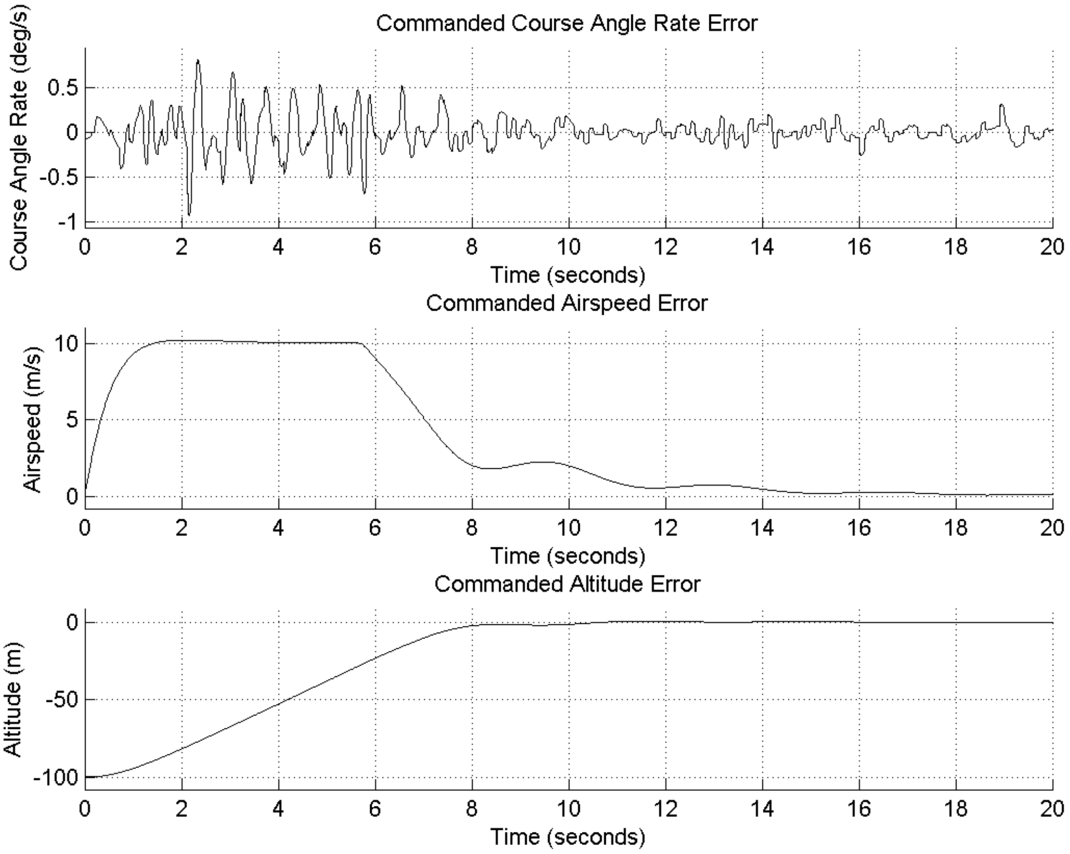


Figure 4.9: Command Tracking Error for Benchmark 3 in Zero Wind

The command error plots in Figure 4.9 show that the aircraft starts at zero altitude at the commanded airspeed and successfully achieves the commanded altitude of 100 meters and airspeed. It can be noticed that there is an excess in airspeed as the aircraft climbs. This is due to the total energy controller driving airspeed up through commanded thrust to account for the insufficient gravitational potential energy prior to the aircraft reaching its commanded height. It can also be seen that there is still a small oscillation in course angle rate. As previously discussed in the straight and level flight benchmark, this is due to the noisy inputs of the lateral accelerometer input y_{a_y}, which in turn creates small sideslip angles and subsequent non-zero course rate angles.

Table 4.7 below contains statistics for these errors in commanded state tracking.

Table 4.7: Command Tracking Statistics for Benchmark 3 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
Course Angle Rate (deg/sec)	-0.00162	0.193157	0.131727	0.14125	0	0.928668
Airspeed (meters/sec)	3.763075	4.141277	3.763075	4.141277	0	10.24756
Altitude (meters/sec)	-20.9374	32.67926	21.35784	32.4059	0.009374	100

The statistics on airspeed and altitude are not very revealing. The error statistics for altitude and airspeed have large zero means, but in this benchmark, these large errors are expected to occur. The simulations for this benchmark are terminated when the aircraft reaches 1 meter of its commanded altitude, and due to the total energy controller as previously mentioned, airspeed is in excess until the commanded altitude is achieved. What really matters here is the fact that the controller successfully brings the aircraft to the commanded airspeed and altitude by the end of the simulation, which for zero wind is depicted by Figure 4.9. The only item of true interest in Table 4.7 is the statistics on the course angle rate. However, the average error of the course angle rate is very close to zero, and the standard deviation of the error about its mean is only 0.16 degrees. In conclusion, these results show that the autopilot is able to track the commanded air speed, altitude, and course angle rate well in zero wind.

Figure 4.10 below shows the performance of the autopilot under Dryden wind.

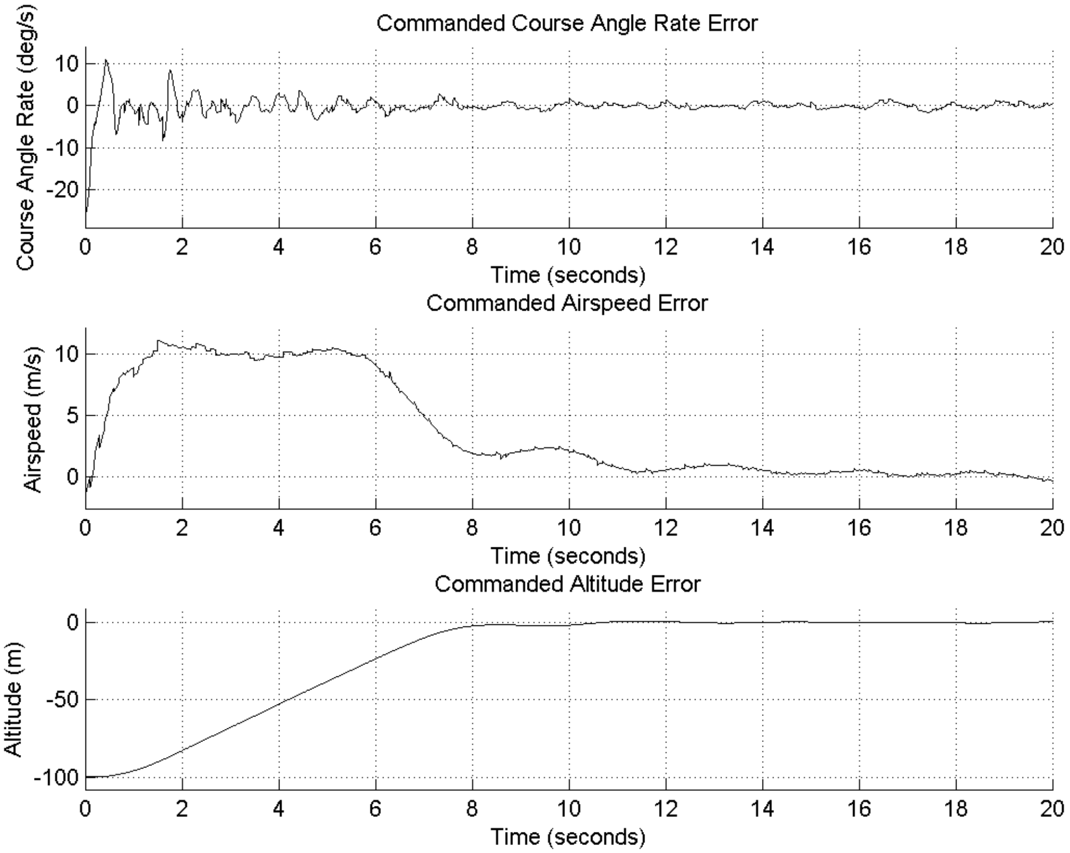


Figure 4.10: Command Tracking Error for Benchmark 3 in Light Wind

Figure 4.10 shows that the autopilot again successfully drives the initial airspeed and altitude command error to zero in Dryden wind. Table 4.8 below contains the statistics for the errors illustrated in Figure 4.10.

Table 4.8: Command Tracking Statistics for Benchmark 3 in Light Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
Course Angle Rate (deg/sec)	-0.14144	2.333533	1.12711	2.048018	0.000973	25.95318
Airspeed (meters/sec)	3.740615	4.14763	3.761287	4.128884	0.003035	11.16299
Altitude (meters/sec)	-21.1698	32.95503	21.50988	32.73398	0.000211	100

Table 4.8 shows that the course angle rate error is again small, with its mean being again essentially zero. The standard deviation of the course angle rate error is larger with wind, but that is to be

expected, and is the same order of magnitude of the previous wind simulations. This is still not concerning, because it is expected that the aircraft will be blow around by the wind, and these results show that the induced course rate error is still quickly driven to zero.

In conclusion, the autopilot has been shown to perform well in Dryden wind, in that it consistently drives the wind-induced errors in commanded airspeed, altitude, and course angle rate to zero in all three benchmark maneuvers. This validates that the set of states identified in (4.11) are sufficient to achieve autopilot control of a small UAV in wind, so long as the states are estimated perfectly.

4.6 Relative Importance of the Chosen States

Now, a short analysis of the sensitivity of the total energy controller autopilot's ability to maintain its commanded states is included here. This is not too detailed, as thoroughly analyzing a specific autopilot's sensitivity to state errors is not within the scope of this research. However, because state estimation and controller design are tied so close together, it is necessary to show which of the state estimates need to have the most stringent requirements so that the performance of different methods of state estimation can be properly assessed.

This simple autopilot sensitivity analysis was performed by making constant errors in each of the states in (4.11), one at a time, keeping all other states equal to their true value. A more sophisticated analysis would involve randomly varying errors, but that is not performed here, as only high level conclusions on the effects of poor state estimation is desired. Table 4.9 below contains the results of such simulations, all performed using a benchmark 2 coordinated turn maneuver in light Dryden wind.

Table 4.9: Sensitivity of the Autopilot's Command Tracking to Constant State Errors

State	Introduced Error	Course Rate Error (deg/s)		Airspeed Error (m/s)		Altitude Error (m)	
		μ	σ	μ	σ	μ	σ

BASELINE	**0**	**0.355298**	**4.144099**	**-0.05336**	**0.215181**	**-0.05581**	**0.208895**
h	+5 m	-0.47946	3.352474	0.021929	0.631637	-0.0776	1.111576
V_a	+3 m/s	0.472136	4.118756	-3.04509	0.161344	0.032143	0.233071
ϕ	-10 deg	7.443571	5.341716	0.528943	1.264033	-0.93142	2.272585
θ	-10 deg	-0.5131	3.115593	-0.03362	0.402857	0.018373	0.708731
p	+5 deg/s	-0.51173	3.165355	0.00868	0.436952	-0.0477	0.784684
q	+5 deg/s	0.012568	0.991571	-0.00944	0.225685	-0.0122	0.35684
V_g	+5 m/s	3.428997	3.855824	0.067841	0.605879	-0.14851	1.086253
y_{a_y}	+1 m/s2	-0.82063	3.501202	-0.07812	0.45316	0.056595	0.86667

Looking at Table 4.9 by commanded state individually, the autopilot's ability to track the desired course rate is most sensitive to the roll angle ϕ, as a 10 degree error in the roll angle estimate creates a steady state tracking error of about 7 deg/s. If the commanded course rate by the navigation layer controller cannot compensate for the autopilot not successfully tracking a desired course rate, this could cause a small UAV to perform indefinite circling maneuvers. Further, if this error is large enough, the aircraft could attempt to perform turns that are too sharp for it to maintain flight, and cause it to eventually crash.

It should also be noted that the course angle rate is also sensitive to the ground speed estimate V_g . This makes sense, as the coordinated turn equation (4.17) shows that ground speed is inversely proportional to the course angle rate. When an excess ground speed is estimated, the autopilot's roll hold equation (4.4) will think the actual course angle rate is smaller than it really is, resulting in a larger roll angle and faster turn.

Next, the largest cause for error in airspeed tracking is the measurement of airspeed itself. The autopilot controls airspeed by directly measuring airspeed and commands correction to any error in what it measures with thrust. In most small UAV applications, maintaining airspeed is only critical to avoid stall. Thus, airspeed will be considered here to be one of the least important measures of state estimator performance, so long as the magnitude of the airspeed errors stay well below the difference between the flight speed and the stall speed.

Last, the largest error in altitude isn't due to error only in one state estimate. Errors in commanded altitude tracking are affected by errors in the estimates for altitude, roll, pitch, pitch rate, ground speed, and side specific force. This makes sense because the autopilot does not track altitude directly. The altitude command only appears in the equation for total energy, which outputs commanded thrust. Altitude is changed most directly by changing pitch, but commanded pitch is only a function of commanded airspeed. Thus, if pitch is off, the only effect is the aircraft performs a slow oscillation in altitude and subsequent airspeed. This notion is backed up by a simulation with an exaggerated pitch angle error of 30 degrees, the results of which are in Figure 4.11 below.

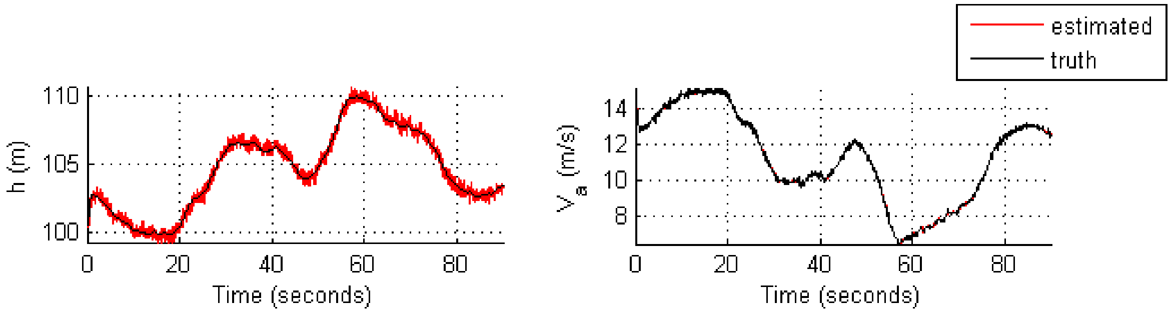


Figure 4.11: Effects of Bad Pitch Estimation

The effect of a pitch error is that the autopilot is commanding the appropriate pitch angle based on altitude, but instead tracks an offset in pitch angle. This causes the aircraft to pitch up or down, depending on the sign of the pitch angle error, and climb away or descend from the desired altitude, which in turn affects airspeed. Fortunately though, the increased airspeed that is accompanied with a decrease in pitch and flight path angle results in an increased pitch command, and thus the aircraft never crashes; it instead oscillates in altitude. Moreover, the autopilot also employs saturation on elevator deflection so that an extreme pitch angle is never reached. Thus, an error in pitch angle results in an undesired and interesting oscillation in altitude and subsequently airspeed. Fortunately though, an aircraft is still able to get to its destination this manner.

In conclusion, roll angle ϕ is the most important state to estimate correctly, as errors in it can cause the aircraft to indefinitely circle or in some extreme cases crash due to trying to undergo turns that are too sharp. Unfortunately, the state estimation methods used later in this document show that this is also the most difficult state to estimate.

4.7 Autopilot Summary

This section has shown that a total energy controller autopilot is sufficient to control the RECUV Tempest small UAV in the zero wind, light wind, and heavy winds that are outlined in section 4.3.4, under the 3 benchmark maneuvers described in section 4.3. This proves that that the states listed in (4.11), $h, p, q, \phi, \theta,\ V_a,\ V_g,$ and y_{a_y}, are sufficient information for an autopilot to control a small UAV in wind.

The set of states in (4.11) are the set of states that this research will focus on estimating. Again, estimating air data states are not in the scope of this research. Air data states are more critical to autopilots for full scale aircraft, as illustrated by [15], and air data states may be necessary for particular small UAV applications, such as for wind energy harvesting.

5 Method 1: IMU Attitude, Smoothed GPS, and Pressure Sensors

5.1 Description

Because of their required small size and low cost, small UAVs typically possess three types of sensors as input to their state estimators: micro-electro-mechanical systems (MEMS) gyros, MEMS accelerometers, and a GPS receiver. Thanks to the recent explosion of the demand for these devices in personal electronic devices, the cost of these sensors has dropped and supply has increased dramatically. Sometimes small UAVs also have other sensors, like three axis magnetometers (TAMs), horizon sensors, cameras [16], or range finders [17]. All of these additional sensors can present their own unique implementation issues, but more importantly, they are not inherently critical to state estimation for autopilot control, which will be shown shortly.

Gyro and accelerometer sensors are often bundled together into a single unit which is often called an inertial measurement unit (IMU) or inertial navigation system (INS). Two main methods exist to estimate the state of an aircraft from an IMU and a GPS receiver. These two methods, which are described shortly, involve different approaches. Method 1 does not use a priori information at each time step, and instead simplifies the equations of motions to get estimated states. It uses two different EKFs to take care of sensor errors and errors induced by simplifying the equations of motion. Method 2 uses a priori information to numerically integrate its sensor inputs to propagate its position, velocity, and attitude estimates from the last time step, without needing to simplify the equations of motion. It then uses an EKF to smooth and filter the estimate.

5.1.1 The Transport Theorem

Both of the state estimation methods in this paper rely on measurements of inertial acceleration, aircraft angular rates, and inertial (GPS) position and velocity measurements. To begin the description of the equations of motion used in these methods, an equation is needed to relate the states to these measurements. This is provided by [18] as the transport theorem in equation (5.1).

$$\boldsymbol{a}_{B/N} = \frac{{}^{B}d}{dt}\boldsymbol{V}_{B/N} + \omega_{B/N} \times \boldsymbol{V}_{B/N} \tag{5.1}$$

Now, for the purposes of this document, it is simpler to write everything in body frame coordinates like in equation (5.2).

$$^{B}\left[\boldsymbol{a}_{B/N}\right] = \frac{\boldsymbol{d}^{\boldsymbol{B}}}{\boldsymbol{dt}}\left[\boldsymbol{V}_{B/N}\right] + {}^{B}\left[\omega_{B/N}\right] \times {}^{B}\left[\boldsymbol{V}_{B/N}\right] \tag{5.2}$$

In body frame components, the inertial acceleration of the aircraft $^{B}\left[\boldsymbol{a}_{B/N}\right]$ can be related to the accelerometer measurements $y_{a_x}, y_{a_y}, y_{a_z}$ through the use of the gravity vector $\boldsymbol{g}$. This is because the accelerometers measure acceleration with respect to the N frame plus the acceleration due to gravity. Because the N frame is fixed to a point on the Earth's surface, any point that is not changing position relative to the N frame experiences the same gravity. Thus, the acceleration due to gravity must be taken out of the accelerometer readings to obtain the acceleration of the aircraft relative to the N frame. The gravity vector is defined in inertial N frame components as the acceleration due to gravity g in the downward direction, as written in equation (5.3).

$$^{N}[\boldsymbol{g}] = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{5.3}$$

The gravity vector can be rewritten in terms of aircraft B frame components through the use of the rotation matrix defined in equation (3.4). So, its relationship to the inertial acceleration as reported by the accelerometers is given in equation (5.4).

$$ {}^{B}[\boldsymbol{a}_{B/N}] = \begin{bmatrix} y_{a_x} \\ y_{a_y} \\ y_{a_z} \end{bmatrix} + \boldsymbol{R}_{BN} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{5.4} $$

Now, the transport theorem of equation (5.2) can be written entirely in body B frame component form, as in equation (5.5).

$$ \begin{bmatrix} y_{a_x} \\ y_{a_y} \\ y_{a_z} \end{bmatrix} + \boldsymbol{R}_{BN} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \frac{d}{dt} \begin{bmatrix} u \\ v \\ w \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{5.5} $$

Substituting equations (3.3) and (3.4) into (5.5) and solving yields equation (5.6).

$$ \begin{bmatrix} y_{a_x} \\ y_{a_y} \\ y_{a_z} \end{bmatrix} = \begin{bmatrix} \dot{u} + qw - rv + g\sin\theta \\ \dot{v} + ru - pw - g\cos\theta\sin\phi \\ \dot{w} + pv - qu - g\cos\theta\cos\phi \end{bmatrix} \tag{5.6} $$

This transport theorem based relationship (5.6) is the basis of the kinematics behind both methods of state estimation.

5.1.2 Attitude Determination

The first method of state estimation that is analyzed in this paper is derived in [4], but is also used in [5]. The method first realizes that by making a few assumptions, the transport theorem relationship (5.6) can be used to solve for the attitude roll and pitch angles given accelerometer measurements and angular rates. These assumptions are:

1) The inertial velocity of the aircraft along its axes is constant, as described by equation (5.7).

$$(\dot{u} = \dot{v} = \dot{w} = 0) \tag{5.7}$$

This assumption is made simply because there is no means to directly measure these quantities. It should be noted though that even though the body frame components of the inertial velocity vector are assumed constant, that doesn't mean the inertial frame components of the velocity vector are not changing. For instance, in a perfect coordinated turn, the body frame components of the velocity vector do not change, while the inertial frame components do. This assumption will be re-addressed later with simulation results.

2) The velocity of the air relative to the vehicle is solely due to the inertial velocity of the aircraft, which in other words means that the wind velocity vector $\boldsymbol{V}_w$ is zero. This greatly simplifies the relationship in equation (3.9). Equation (5.8) is this simplified relationship written in body frame coordinates, substituting in equation (3.10) for $\boldsymbol{V}_a$.

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = V_a \begin{bmatrix} \cos\alpha\cos\beta \\ \sin\beta \\ \sin\alpha\cos\beta \end{bmatrix} \tag{5.8}$$

This has huge implications when the purpose of this estimator is perform well in wind. The effects of this assumption will be addressed in the simulation results.

3) Although it has already been assumed that there is zero wind, that doesn't mean that the sideslip angle is always zero in non-stalled conditions, even if the airframe is designed to maintain zero sideslip in normal flight. Although it is typically not done, the autopilot could induce a slipping turn where sideslip is used to turn the aircraft. In any case, in significantly varying wind, sideslip will arise. However, this method assumes sideslip to be zero because there is no direct measure for sideslip. This simplifies the above equation (5.8) further to equation (5.9).

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = V_a \begin{bmatrix} \cos\alpha \\ 0 \\ \sin\alpha \end{bmatrix} \tag{5.9}$$

4) The final assumption assumes that angle of attack α is close to the pitch angle. In both steady turns and significant wind, this is not the case. However, just like sideslip, without sophisticated air data

sensors, there is no direct measure of the angle of attack. Thus, equation (5.9) becomes equation (5.10). The validity of this will again be addressed in the simulation results.

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = V_a \begin{bmatrix} \cos\theta \\ 0 \\ \sin\theta \end{bmatrix} \tag{5.10}$$

With these assumptions, the transport theorem relationship in equation (5.6) is simplified to equation (5.11)

$$\begin{bmatrix} y_{a_x} \\ y_{a_y} \\ y_{a_z} \end{bmatrix} = \begin{bmatrix} qV_a \sin\theta + g\sin\theta \\ rV_a\cos\theta - pV_a\sin\theta - g\cos\theta\sin\phi \\ -qV_a\sin\theta - g\cos\theta\cos\phi \end{bmatrix} \tag{5.11}$$

This gross simplification enables calculating the pitch and roll attitude angles from the accelerometers and angular rates. With the assumptions that were required to obtain this relationship, large errors can be expected in turning flight, accelerating flight, and in significant amounts of wind.

This method attempts to combat these expected errors by first applying low pass filters to the gyro inputs. Performing low pass filters on the sensor inputs removes high frequency noise in the sensors and high frequency vehicle dynamics. Without sensing high frequency dynamics, equation (5.11) will not be as affected by high frequency angular rate dynamics. This avoids high frequency measurement error η in EKF 1 that is about to be discussed. However, the effects of such removal of high frequency dynamics on the actual attitude estimates is assessed in the simulation results. The angular rates needed in equation (5.11) are obtained through equation (5.12).

$$\begin{bmatrix} \hat{p}_{method1} \\ \hat{q}_{method1} \\ \hat{r}_{method1} \end{bmatrix} = \begin{bmatrix} LPF\left(y_{gyro_p}\right) \\ LPF\left(y_{gyro_q}\right) \\ LPF\left(y_{gyro_r}\right) \end{bmatrix} \tag{5.12}$$

In this state estimation method, these angular rates are considered to be the final estimate, and are also used as inputs to later steps of this method's state estimation process. Second, the accelerometers inputs are used without filtering as in equation (5.13).

$$\begin{bmatrix} y_{a_{x,method1}} \\ y_{a_{y,method1}} \\ y_{a_{z,method1}} \end{bmatrix} = \begin{bmatrix} y_{a_x} \\ y_{a_y} \\ y_{a_z} \end{bmatrix} \tag{5.13}$$

It is important to realize that the process of removing bias from both the gyro and accelerometer sensors is expected to be done in a manual calibration procedure outside of this state estimation process. For short duration flights, a manual calibration by leveling the aircraft on the ground can be sufficient, such as is mentioned in [4]. However, for longer flights this requires periodic manual in-flight calibration. This requires another inertial measurement system, such as on-board video to ensure straight and level flight, to perform the calibration procedure. Such a requirement, depending on the application, may not be allowable. Thus, the effects of drifting gyro biases, which is common for cheaper sensors, is also simulated in this document.

5.1.3 Extended Kalman Filter 1

Reference [4] demonstrates that a using a simple low pass filter on both accelerometer and gyro inputs for use in equation (5.11) is not enough to avoid the errors encountered from the assumptions used to derive equation (5.11). It shows that fast changes in attitude angles are sometimes completely missed, resulting in large errors in the attitude angles that last a few seconds. To overcome this, this method augments these low pass filters with an extended Kalman Filter (EKF). Reference [4] is not very clear on whether or not one should remove the low pass filters for both the gyros and accelerometer sensors prior for use by the Kalman filter, but it is chosen for this demonstration to keep the low pass filters on the gyro sensors and use the accelerometer sensors as is, since as is about to be discussed, the gyro sensors are used as inputs to EKF1, and the accelerometers are the measurements. Note however though the cutoff frequency of the low pass filter was set such that optimal performance was obtained, which was about 10 Hz. Tweaking or removing this method of filter could affect the high frequency dynamics of the results, and further research could investigate this area.

This EKF uses the angular rates from equation (5.12) and the airspeed from a pre-calibrated pitot-static sensor as the inputs $\boldsymbol{u}_{EKF1}$, as described in equation (5.14).

$$\boldsymbol{u}_{EKF1} = \left[p_{LPF}\ q_{LPF}\ r_{LPF}\ V_{a,LPF}\ \right]^T \tag{5.14}$$

While an airspeed sensor is not part of an IMU, cheap pitot-static tube based airspeed sensors are readily available for use on small UAVs, as discussed by [19]. However, before the airspeed sensor's input is treated as truth for this EKF, it is also passed through a low pass filter, as shown in the block diagram of this method in Figure 5.1 and written in equation (5.15).

$$V_{a,LPF}\ = LPF\left(y_{V_a}\right) \tag{5.15}$$

Now, the purpose of this EKF is to estimate the attitude. Thus, the EKF uses the roll and pitch angles as the states it's estimating, as depicted by equation (5.16).

$$\boldsymbol{x}_{EKF1} = [\ \phi\ \theta\ ]^T \tag{5.16}$$

Note that the yaw angle ψ is not included here, and is found from EKF2, which is discussed shortly. Finally, the EKF uses the low pass filtered accelerometer sensors as the measurements:

$$\boldsymbol{y}_{EKF1} = \left[\ y_{a_x} y_{a_y}\ y_{a_z}\right]_{LPF}^T \tag{5.17}$$

The low pass filtered angular rates p, q, and r in equation (5.14) cannot be treated as measurements for this EKF because they are needed to define the time derivative of the roll and pitch angle states that are being estimated. This relationship to the state derivatives is written in equation (5.18). This relationship is the $\boldsymbol{F}_{EKF1}$ matrix that is needed to define part of the EFK state-space relationship in (3.11). Equation (5.18) comes from the definition of 3-2-1 Euler angle rates that is provided in equation (3.8).

$$\boldsymbol{F}_{EKF1}(\boldsymbol{x}_{EKF1}, \boldsymbol{u}_{EKF1}) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \end{bmatrix} \begin{bmatrix} p_{LPF} \\ q_{LPF} \\ r_{LPF} \end{bmatrix} \tag{5.18}$$

Multiply the two matrices in equation (5.18) yields equation (5.19).

$$\boldsymbol{F}_{EKF1}(\boldsymbol{x}_{EKF1}, \boldsymbol{u}_{EKF1}) = \begin{bmatrix} p_{LPF} + q_{LPF}\sin\phi\tan\theta + r_{LPF}\cos\phi\tan\theta \\ q_{LPF}\cos\phi - r_{LPF}\sin\phi \end{bmatrix} \tag{5.19}$$

Last, the Jacobian of $\boldsymbol{F}_{EKF1}$ with respect to the states is needed for the EKF time update via the linearized equation (3.22). Computing the Jacobian yields equation (5.20).

$$\frac{\partial \boldsymbol{F}_{EKF1}(\boldsymbol{x}_{EKF1}, \boldsymbol{u}_{EKF1})}{\partial \boldsymbol{x}_{EKF1}} = \begin{bmatrix} q_{LPF}\cos\phi\tan\theta - \sin\phi\tan\theta & \frac{q_{LPF}\sin\phi + r_{LPF}\cos\phi}{\cos^2\theta} \\ -q_{LPF}\sin\phi - r_{LPF}\cos\phi & 0 \end{bmatrix} \tag{5.20}$$

Now, the measurement relationship $\boldsymbol{H}_{EKF1}$needed for the EKF in equation (3.12) comes directly from the simplified transport theorem relationship in equation (5.11), but taking care to use EKF input low pass filtered angular rates and airspeed. This relationship is depicted in equation (5.21).

$$\boldsymbol{H}_{EKF1}(\boldsymbol{x}_{EKF1}, \boldsymbol{u}_{EKF1}\ ) = \begin{bmatrix} q_{LPF}V_{a,LPF}\sin\theta + g\sin\theta \\ r_{LPF}V_{a,LPF}\cos\theta - p_{LPF}V_{a,LPF}\sin\theta - g\cos\theta\sin\phi \\ -q_{LPF}V_{a,LPF}\sin\theta - g\cos\theta\cos\phi \end{bmatrix} \tag{5.21}$$

Now, the Jacobian of the $\boldsymbol{H}_{EKF1}$ matrix is needed for the measurement update. This is directly computed from equation (5.21) to be equation (5.22).

$$\frac{\partial \boldsymbol{H}_{EKF1}}{\partial \boldsymbol{x}_{EKF1}} = \begin{bmatrix} 0 & q_{LPF}V_{a,LPF}\cos\theta + g\cos\theta \\ -g\cos\phi\cos\theta & -r_{LPF}V_{a,LPF}\sin\theta - p_{LPF}V_{a,LPF}\cos\theta + g\sin\phi\sin\theta \\ g\sin\phi\cos\theta & \left(q_{LPF}V_{a,LPF} + g\cos\phi\right)\sin\theta \end{bmatrix} \tag{5.22}$$

This is how EKF 1 obtains high frequency estimates of the roll and pitch angles. It is a piece of the Method 1 estimation algorithm, as depicted in the flow diagram in Figure 5.1.

5.1.4 Extended Kalman Filter 2

Now, another process is needed to determine the remaining states that are listed in (4.11), which are the altitude h and the ground speed V_g. Note that it is also necessary to calculate the course angle χ so that a navigational layer algorithm can track a desired course. In [3], altitude comes from a low pass filter on a barometric pressure sensor to obtain frequently updated altitude information, as depicted by equation (5.23).

$$h = LPF(y_{h_{static\ pressure}}) \tag{5.23}$$

This removes the necessity for the altitude h, or position p_d, to be part of the Kalman filter that is about to be discussed.

Now, the determination of inertial velocities and position is fairly involved. This method uses the known attitude to “smooth” the GPS measurements to determine high frequency estimates for the states in (5.24) using a second EKF that for the purposes of this paper is called EKF 2.

$$\boldsymbol{x}_{EKF2} = \left[\, p_n \; p_e V_g \; \chi \; w_n \; w_e \; \psi \,\right]^T \tag{5.24}$$

GPS updates for low cost receivers are typically on the order of 1-2 updates per second, so this method uses EKF 1 information to propagate the inertial velocities and positions between GPS updates. Such an act is necessary because updates on inertial position data are simply not frequent enough for autopilots to counteract the high frequency dynamics small UAVs experience. See reference [2] for more information on the high frequency nature of small UAVs. It should be noted, however, that the method here is to smooth the existing GPS data given what data is available between GPS updates. The accelerometers have already been used to determine the attitude, and they are not reused here because they do not give direct mappings to all of the state derivatives, such as $\dot{p}_n$ and $\dot{p}_e$. Such a relationship would involve numerically integrating accelerometer inputs (see Method 2).

Instead, to accomplish this smoothing, EKF 2 uses the attitude output of EKF1 as well as the inputs to EKF1 as its truth inputs, which is depicted in equation (5.25).

$$\boldsymbol{u}_{EKF2} = \left[\, \hat{\phi}_{EKF1}\; \hat{\theta}_{EKF1}\; p_{LPF}\; q_{LPF}\; r_{LPF}\; V_{a,LPF} \right]^T \tag{5.25}$$

EKF 2 uses only the GPS measurements, as depicted by (5.26).

$$\boldsymbol{y}_{EKF2} = \begin{bmatrix} y_{p_{n,GPS}} \\ y_{p_e,GPS} \\ y_{V_{g,GPS}} \\ y_{\chi_{GPS}} \\ 0 \\ 0 \end{bmatrix} \tag{5.26}$$

Equation (5.26) assumes that the GPS sensor provided a north position $y_{p_{n,GPS}}$, east position $y_{p_e,GPS}$, ground speed $y_{V_{g,GPS}}$, and course angle $y_{\chi_{GPS}}$. This is slightly different form the GPS measurements that are used in method 2, which are discussed later. The last two measurements listed in (5.26) are zero. These corresponds to two "pseudo-measurements" that this method uses to represent the measured wind in the north and east directions. Assuming that the flight path angle of the aircraft is zero and that the wind has no vertical component, these "pseudo-measurements" are related to components of the EKF states x and truth inputs u as follows:

$$y_{5_{EKF2}} = 0 = V_a \cos\psi + w_n - V_g \cos\chi \tag{5.27}$$

$$y_{6_{EKF2}} = 0 = V_a \sin\psi + w_e - V_g \sin\chi \tag{5.28}$$

This assumption of flat wind and level flight will be addressed in the simulation results. Realizing that the remainder of the GPS estimates match directly up with their state vector counterparts in equation (5.24), the measurement-to-state relationship $\boldsymbol{H}_{EKF2}$ in equation (3.12) can now be written as equation (5.29).

$$\boldsymbol{H}_{EKF2}(\boldsymbol{x}_{EKF2}, \boldsymbol{u}_{EKF2}) = \begin{bmatrix} p_n \\ p_e \\ V_g \\ \chi \\ V_a \cos\psi + w_n - V_g \cos\chi \\ V_a \sin\psi + w_n - V_g \sin\chi \end{bmatrix} \tag{5.29}$$

Though it is large, the Jacobian that is required for the measurement update in equations (3.15) and (3.16) is fairly straightforward to derive from equation (5.29). This measurement update Jacobian is in equation (5.30).

$$\frac{\partial \boldsymbol{H}_{EKF2}(\boldsymbol{x}_{EKF2}, \boldsymbol{u}_{EKF2})}{\partial \boldsymbol{x}_{EKF2}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\cos\chi & V_g \sin\chi & 1 & 0 & -V_{a,LPF} \sin\psi \\ 0 & 0 & -\sin\chi & -V_g \cos\chi & 0 & 1 & V_{a,LPF} \cos\psi \end{bmatrix} \tag{5.30}$$

Now, in order to relate the state derivatives to the states and inputs as is needed for EKF relationship (3.11), some additional assumptions had to be made.

1) An assumption is made that the flight path angle is zero. This yields the relationships (5.31) and (5.32).

$$\dot{p}_n = V_g \cos\chi \tag{5.31}$$

$$\dot{p}_e = V_g \cos\chi \tag{5.32}$$

2) Next, assumptions that wind and airspeed are constant had to be made to yield the following relationships (5.33) and (5.34). Detailed derivation of these are found in [4].

$$\dot{V}_g = \frac{(V_a \cos\psi + w_n)(-v_a \dot{\psi} \sin\psi) + (V_a \sin\psi + w_e)(V_a \dot{\psi} \cos\psi)}{V_g} \tag{5.33}$$

$$\dot{\psi} = q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta} \tag{5.34}$$

Assumption 2 that was performed as part of the attitude determination process assumed that wind speed was zero, but it did not assume that airspeed was constant. Here, wind must be taken into account in order to for relationship (5.33) to somewhat properly measure ground speed at each time update of EKF2. Since the assumption of zero wind was used to determine the attitude, the estimated yaw rate could be expected to suffer from this, and thus affect the ground speed estimate. Using an assumption of constant airspeed just lets equation (5.33) neglect higher order terms of airspeed, for which there is no information. These assumptions will make themselves apparent and be discussed in the simulation results. Note that equation (5.34) itself does not involve any assumptions about the motion of the aircraft, and is just necessary for the calculation of (5.33).

3) Last, an assumption that the aircraft always experiences coordinated turns (even in wind) was used to derive the relationship (5.35).

$$\dot{\chi} = \frac{g}{V_g}\tan\phi\cos(\chi - \psi) \tag{5.35}$$

It should be noted that in the transport theorem simplification (5.7) that was used forEKF1, it was assumed that the aircraft was not accelerating. While that assumption is not made here, errors in the roll angle and yaw angle could be expected to make their way through EKF1. The existence of such errors would have an effect on this coordinated turn relationship (5.35) and thus the estimated course angle rate. One of the benchmarks is a coordinated turn, and this will be discussed with the results of that simulation.

Placing these simplified relationships together yields the $\boldsymbol{F}_{EKF2}$ matrix that is needed by the EKF relationship (3.11). This relationship is written in equation (5.36).

$$\boldsymbol{F}_{EKF2}\left(\boldsymbol{x}_{EKF2}, \boldsymbol{u}_{EKF2}\right) = \begin{bmatrix} V_g \cos\chi \\ V_g \sin\chi \\ \frac{(\hat{V}_{a,LPF}\cos\psi + w_n)(-\hat{V}_{a,LPF}\dot{\psi}\sin\psi) + (\hat{V}_{a,LPF}\sin\psi + w_e)(\hat{V}_{a,LPF}\dot{\psi}\cos\psi)}{V_g} \\ \frac{g}{V_g}\tan\phi\cos(\chi-\psi) \\ 0 \\ 0 \\ \hat{q}\frac{\sin\hat{\phi}}{\cos\hat{\theta}} + \hat{r}\frac{\cos\hat{\phi}}{\cos\hat{\theta}} \end{bmatrix} \quad (5.36)$$

Now, the Jacobian of $\boldsymbol{F}_{EKF2}$ that is needed for the time update equation (3.22) can be derived from (5.36) to (5.37).

$$\frac{\partial \boldsymbol{F}_{EKF2}(\boldsymbol{x}_{EKF2}, \boldsymbol{u}_{EKF2})}{\partial \boldsymbol{x}_{\boldsymbol{EKF2}}} = \begin{bmatrix} 0 & 0 & \cos\chi & -V_g\sin\chi & 0 & 0 & 0 \\ 0 & 0 & \sin\chi & V_g\cos\chi & 0 & 0 & 0 \\ 0 & 0 & -\frac{\dot{V}_g}{V_g} & 0 & -\frac{\dot{\psi}V_{a,LPF}\sin\psi}{V_g} & \frac{\dot{\psi}V_{a,PLF}\cos\psi}{V_g} & \frac{\partial V_g}{\partial\psi} \\ 0 & 0 & \frac{\partial\dot{\chi}}{\partial V_g} & \frac{\partial\dot{\chi}}{\partial\chi} & 0 & 0 & \frac{\partial\dot{\chi}}{\partial\psi} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (5.37)$$

Note the zeroes in the last three rows of the Jacobian for $\dot{w_n}$, $\dot{w_e}$, and $\dot{\psi}$. This comes from the assumptions that wind speed is constant and the fact that yaw rate does not depend on any of the states of EKF2. This just means that the time update will not update the covariance matrix for wind speed or course angle. The equations for the partials used above are:

$$\frac{\partial V_g}{\partial \psi} = -\frac{\dot{\psi} V_{a,LPF}(w_n \cos\psi + w_e \sin\psi)}{V_g} \tag{5.38}$$

$$\frac{\partial \dot{\chi}}{\partial V_g} = -\frac{g}{V_g^2} \tan\hat{\phi}\cos(\chi - \psi) \tag{5.39}$$

$$\frac{\partial \dot{\chi}}{\partial \chi} = -\frac{g}{V_g} \tan\hat{\phi}\sin(\chi - \psi) \tag{5.40}$$

$$\frac{\partial \dot{\chi}}{\partial \psi} = \frac{g}{V_g} \tan\hat{\phi}\sin(\chi - \psi) \tag{5.41}$$

This concludes the description for EKF2, and its place in Method 1's algorithm can be seen in Figure 5.1.

5.1.5 Method 1 Summary

Now that EKF 2 is in place, it can be noticed that nothing has actually estimated the inertial velocities u, v, w. While they are not needed for the minimum set of states (4.11), they can still be backed out from the output estimated states $\boldsymbol{x}_{EKF2}$ if the wind is zero. A zero wind assumption is necessary for this because the inertial velocities are related to the wind vector through (3.9) and (3.10) in body frame B component form as depicted by equations (5.42) and (5.43).

$$^{B}[\boldsymbol{V}_a] = {}^{B}[\boldsymbol{V}_{B/N}] - {}^{\boldsymbol{B}}[\boldsymbol{V}_w] \tag{5.42}$$

$$V_a \begin{bmatrix} \cos\alpha\cos\beta \\ \sin\beta \\ \sin\alpha\cos\beta \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} - \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} - \boldsymbol{R}_{BN} \begin{bmatrix} w_n \\ w_e \\ w_d \end{bmatrix} \tag{5.42}$$

Moreover, in this estimator, angle of attack α and angle of sideslip β are not available. In EKF 1, sideslip was assumed be zero and angle of attack was assumed to be equivalent to the pitch angle. Further, it should be noted that many assumptions were made to come up with the EKF that estimates wind. Simulation of this filter even in zero wind show that the estimates for the wind are unacceptable. This is likely due to many sensor and modelling errors manifesting themselves in the wind estimate. Thus, it has been observed through simulation that simply leaving these wind terms out of the equation produces far better estimates of the inertial velocities. Thus, equation (5.42) simplifies to equation (5.43).

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = V_a \begin{bmatrix} \cos\theta \\ 0 \\ \sin\theta \end{bmatrix} \tag{5.43}$$

Again, inertial velocity information is not necessary for the autopilot input states listed in (4.11), but this simplification in (5.3) provides a comparison point between methods 1 and 2.

So, in conclusion, method 1 produces high frequency information for the following states, listed in Table 5.1.

State	Used by Autopilot	Noteworthy Comment
p_n	N	
p_e	N	
h	Y	Directly from low pass filter of pressure sensor.
p	Y	Directly from low pass filter of gyro.
q	Y	Directly from low pass filter of gyro.
r	Y	Directly from low pass filter of gyro.
u	Y	Equal to $V_a \cos\theta$, assumes zero wind.
v	Y	Always zero
w	Y	Equal to $V_a \sin\theta$, assumes zero wind.
V_g	N	
ϕ	Y	See section 5.1.2 for assumptions.
θ	Y	See section 5.1.2 for assumptions.
ψ	N	
w_e	N	Pseudo-estimate that contains other errors
w_d	N	Pseudo-estimate that contains other errors
χ	N	

Table 5.1: Summary of States Captured by Method 1

Note that some of the states not used in the autopilot system here could be used in a navigation layer controller. However, straight GPS measurements are usually sufficient for navigation layer control.

Below in Figure 5.1 is a flow diagram that summarizes this method.

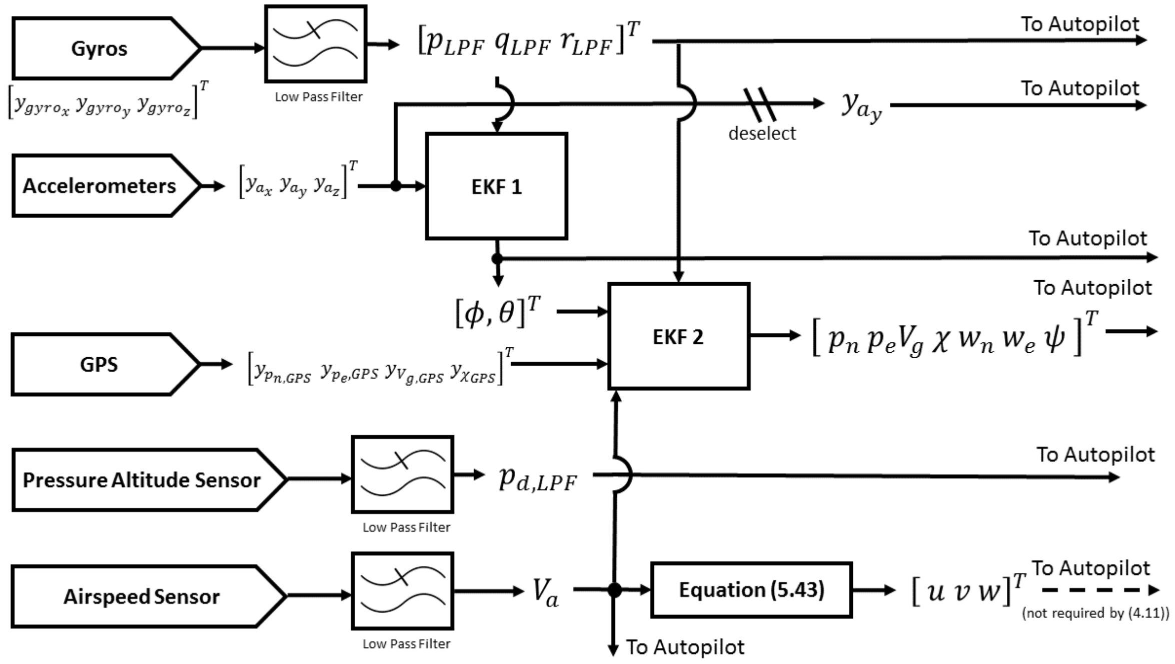


Figure 5.1: Method 1 Flow Chart

5.2 Detailed Simulation Results

Now that method 1 has been presented in detail, the effectiveness of this method can be assessed simulating the state estimation method. The simulation setup that is used here is the same that was described in section 4.3. To assess the effectiveness of the estimation method, the states that are needed by autopilot control, which are listed in (4.11), are deeply scrutinized. The results shown in this section contain plots of the state estimates that state estimation method 1 produces on the same plots as the true states that the simulation produces. Again, the simulations here use a linearized model of the RECUV Tempest small UAV. Nonlinear effects of wind on the dynamics of the aircraft are not taken into account. The fact that these simulations do not involve a nonlinear aerodynamic model is discussed in section 0 on future studies.

Furthermore, the autopilot previously discussed is used to maintain the benchmark maneuvers here. The autopilot is fed the true states that the simulation provides and not the estimated states that method 1 provides. The purpose of this is to assess the dynamics of the state estimation methods individually so that their strengths and weaknesses can be judged without the effects of closed loop control on the state estimation.

5.2.1 Zero Wind

5.2.1.1 Benchmark 1: Straight and Level, without Gyro Bias Drift

All simulations in this document, except the one in this immediate section, were performed with a linear rate gyro bias drift of $\frac{1}{30}\frac{degrees}{second^2}$ on each gyro axis. This creates a total bias drift of $3\frac{degrees}{second}$ over the course of the 90 second simulations. The rate gyro bias will always begin at zero, as it is assumed that the gyro rate biases are calibrated out at the beginning of each simulation. However, the gyro bias is not included here to better illustrate the basic results of a benchmark 1 maneuver in zero wind. The plots below contain results of this simulation. The plots contain both the true value of the states listed in (4.11) and the estimated value of the states listed in (4.11) that method 1 generates.

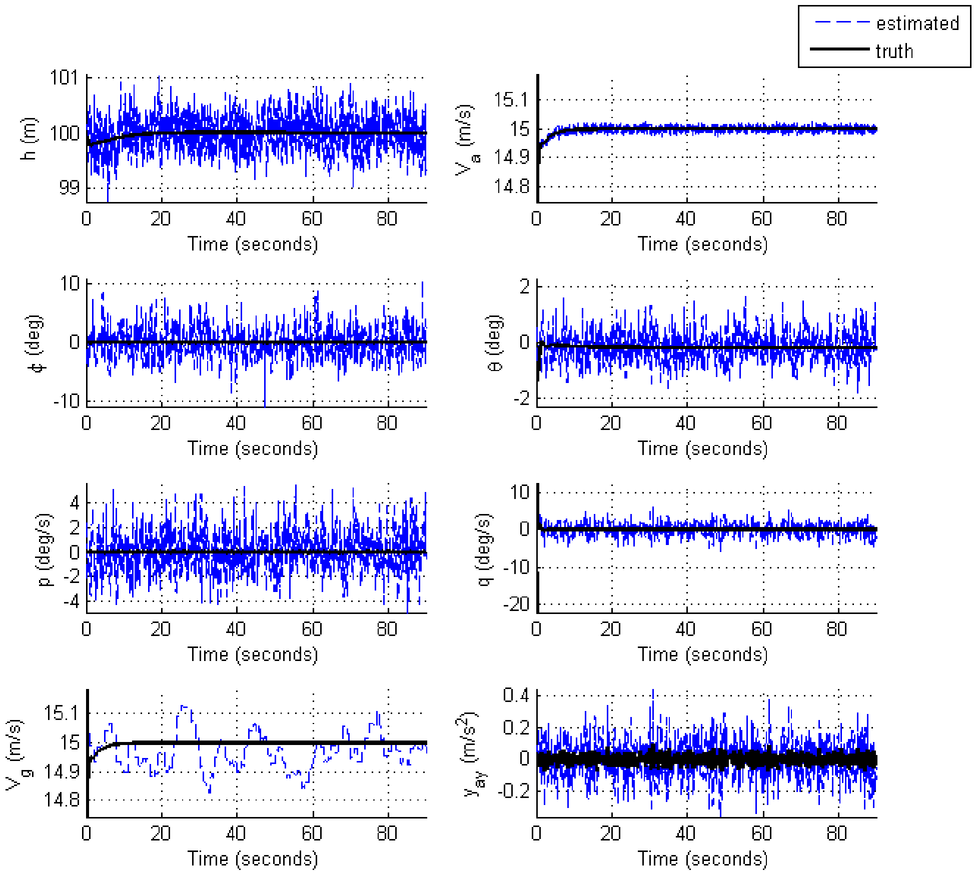


Figure 5.2: Method 1 State Estimation for Benchmark 1 in Zero Wind and Zero Gyro Bias Drift

This shows that method 1 was able to successfully converge on an estimate of the states in straight and level flight and zero wind. It is important to elaborate in detail on the results here, as this is used as the base results that are compared against in the rest of this section. To do this, Table 5.2 below contains statistics on the error between the estimated values of the each state and the true values of each state. The sign convention that is used for the non-absolute value terms in the state error statistics tables in this document is the estimated value subtracted from the true value. So, if an estimated value of a state is larger than the true value of the state, the error will be negative.

Table 5.2 : Statistics of Method 1 State Error for Benchmark 1 in Zero Wind and Zero Gyro Bias

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}

$h\ (m)$	-0.01583	0.26844	0.213742	0.163156	0.000075	1.110218
$V_a\ (m/s)$	-6.4E-05	0.007263	0.005578	0.004651	0	0.116425
$\phi\ (deg)$	0.032404	2.433725	1.907496	1.511661	0.000052	10.92272
$\theta\ (deg)$	-0.02229	0.986594	0.426362	0.889977	0.000041	43.50571
$p\ (deg/s)$	-0.04444	1.579859	1.240624	0.979087	0.000235	5.599666
$q\ (deg/s)$	-0.06805	1.631634	1.262134	1.036195	0.000295	17.64951
$V_g\ \ (m/s)$	0.024623	0.067741	0.0585	0.042102	0.00001	0.300109
$y_{a_y}\ (m/s^2)$	-0.00376	0.096081	0.075681	0.059308	0.000007	0.34756

The results here show that the mean of the errors are all essentially zero. State estimation method 1 made many assumptions of straight and level flight and zero wind, and this shows that there is in fact no bias in the estimates when flying in straight and level flight with zero wind. Now, although the mean of the errors is zero, the values are not always zero. The standard deviation of the errors and the mean of the absolute values of the errors both reflect how much error there typically is in the estimate for straight and level flight with zero wind. The errors in the ground speed V_g, side specific force y_{a_y}, pitch angle θ, altitude h, and airspeed V_a are very small. However, the errors in roll angle ϕ, roll rate p, and pitch rate q are not negligible. The roll angle estimate is shown to typically vary by about 2 degrees, and the pitch rate typically vary about 1 deg/s. While such a variance is not ideal, it is to be expected given the noisy nature small UAV sensors. Roll error of 2 degrees is within the tolerance of what a human pilot would fly a small UAV at, and the autopilot sensitivity analysis of section 4.6 show how only steady state errors of 10 degrees begin to seriously affect the autopilot's ability to track the states.

Now, the source of this error can be strictly attributed to sensor noise. A simulation was performed with zero sensor noise, and the following estimation performance was achieved.

Table 5.3: Statistics of Method 1 State Error for Benchmark 1 in Zero Wind, Zero Gyro Bias, and Zero Sensor Noise

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	0	0.001846	0.00038	0.001807	0	0.021227

$V_a\ (m/s)$	0	0.002356	0.000109	0.002354	0	0.116063
$\phi\ (deg)$	-4.5E-05	0.00281	0.000045	0.00281	0	0.231522
$\theta\ (deg)$	-0.00778	0.851058	0.039712	0.850167	0	43.59248
$p\ (deg/s)$	0	0	0	0	0	0
$q\ (deg/s)$	-8E-06	0.346004	0.012717	0.34577	0	19.39894
$V_g\ \ (m/s)$	14.99637	0.016945	14.99637	0.016945	14.74299	15.18658
$y_{a_y}\ (m/s^2)$	0	0	0	0	0	0

All of the errors are essentially exactly zero. Thus, for method 1 in straight and level flight in zero wind, and without gyro bias drift, the errors are solely attributable to sensor noise and not the system model.

5.2.1.2 Benchmark 1: Straight and Level

Next, the same simulation as the previous section was performed, but the gyro rate bias drift of $\frac{1}{30}\frac{degrees}{second^2}$ on each of the 3 axes was added back into the simulation. Below are the results of this simulation.

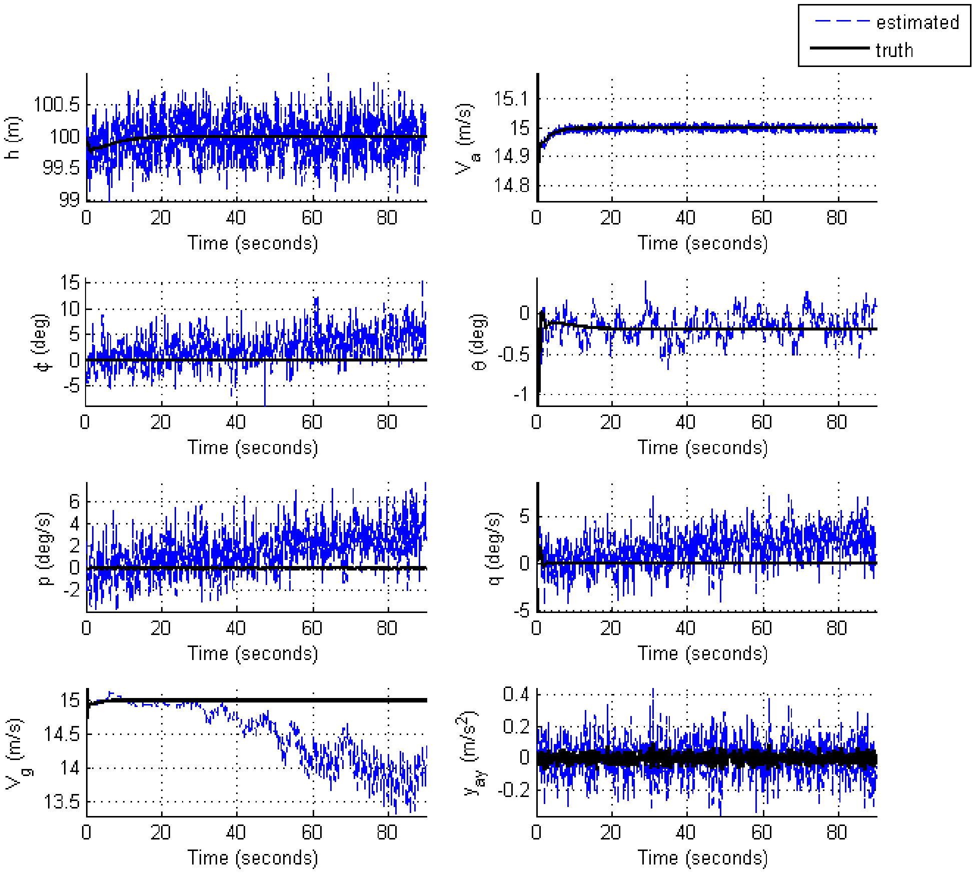


Figure 5.3: Method 1 State Estimation for Benchmark 1 in Zero Wind

Table 5.4: Error Statistics of Method 1 State Estimation for Benchmark 1 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	0.009319	0.263634	0.21069	0.158728	0.000008	0.999435
$V_a\ (m/s)$	0.00013	0.007123	0.00544	0.004599	0.000001	0.114609
$\phi\ (deg)$	-2.43329	2.84864	3.000781	2.2429	0.000158	16.11988
$\theta\ (deg)$	-0.02924	0.174277	0.135	0.114019	0.000023	1.950974
$p\ (deg/s)$	-1.5435	1.865565	1.95948	1.422327	0.000516	7.854573
$q\ (deg/s)$	-1.56711	1.818091	1.967571	1.374701	0.000305	17.64951
$V_g\ \ (m/s)$	0.474439	0.456241	0.488593	0.441049	0.003294	1.709627
$y_{a_y}\ (m/s^2)$	-0.00376	0.096081	0.075681	0.059308	0.000007	0.34756

These results show there are a number of differences in the performance of state estimation method 1 when compared with the zero bias drift simulation of the previous section. They are similar to the zero gyro bias drift simulation of the previous section, but with a few exceptions.

First, the results show an accumulation of error in the estimated ground speed V_g. This makes sense, as the system model's time update equation for ground speed is affected by the input angular rates. The time update equation (5.33) for V_g calculates the change in ground speed from airspeed and yaw angle, as depicted in equation (5.33). The time update equation (5.34) for the yaw angle ψ is affected by the angular rates, and as depicted in Figure 5.4, EKF2 does sufficiently correct for yaw angle error over time.

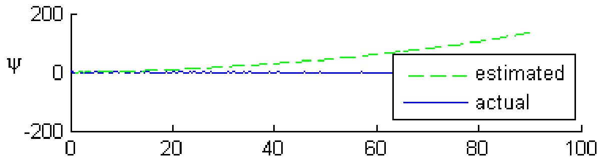


Figure 5.4: Estimated Yaw Angle for Benchmark 1 in Zero Wind

IT makes sense that this unbounded yaw drift occurs, as EKF2 has no state or other mechanism that accounts for gyro bias. The EFK2 time update equation of (5.36) relates the yaw angle to the direct roll and yaw rate inputs, which are just low pass filtered and do not account for sensor bias. Then, the covariance matrix update equation of (5.37) defines the time rate of change of the yaw angle as zero. It should also be noted that this drift does not occur when gyro bias drift is not present. The yaw angle estimate of the previous section is shown in Figure 5.5.

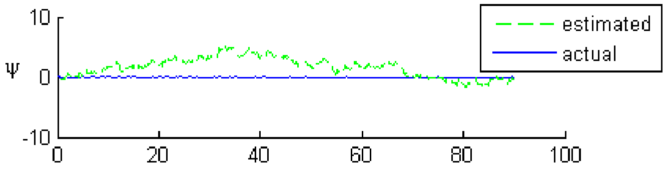


Figure 5.5: Estimated Yaw Angle for Benchmark 1 in Zero Wind and Zero Gyro Bias Drift

When gyro rate bias was zero, EKF2 was able to estimate the yaw angle within bounds of about 5 degrees.

The degradation of the yaw angle when gyro bias drift is present directly feeds into the ground speed time update equation of (5.33). Corrections to this propagated error in the ground speed are performed by the measurement model in equation (5.29), which equates the GPS provided ground speed to the ground speed estimate. However, over time, inaccurate models in the time update cause the covariance matrix to place far less than 100% weight on the GPS measurements. Thus, the jagged behavior of substantial V_g errors being partially corrected once a second in Figure 5.3 is observed.

The reason the degradation in yaw angle estimate occurs is because the yaw angle is not really observable, and thus the Kalman filter gives very little weight to updating the estimate for ψ based on GPS inputs. Unfortunately, a traditional linear system observability analysis using the state-to-measurement matrix in equation (5.30) and state derivative relationships in (5.37) does not yield much insight, as such an analysis yields a full rank observability matrix. The problem lies in the fact that the in the state-to-measurement relationship (5.30), the yaw angle is only affected by "pseudo-measurements". As discussed in section 5.1.4 and is shown in equations (5.27) and (5.28), these "pseudo-functions" are functions of inputs and states – namely the estimated airspeed V_a, ground speed V_g, course angle χ, the estimated wind velocity components w_n, and w_e, and even the yaw angle ψ! If EFK2 was setup such that there were not fake measurements, the matrix in (5.30) would not have equations for the last three states - w_n, w_e, and ψ. Thus, the observability matrix would not be full rank. In conclusion, if an estimate for V_g is desired that is not affected by angular rate errors, interpolating GPS measurements would be one better approach. This topic could be an area of future work.

Second, the results in Figure 5.3 show a drift in the estimate of the angular rate p and q and the estimate for the bank angle ϕ. The fact that the angular rate bias drift shows in the attitude estimate is because of the time and measurement update equations for EKF1. The attitude propagation equation (5.19), even in straight and level flight where $\phi = \theta = 0$, propagates the error in the angular rates p, q, r into both the bank and pitch angle estimates. Equation (5.44) below is equation (5.19) in straight and level flight conditions.

$$\boldsymbol{F}_{EKF1}(\boldsymbol{x}_{EKF1}, \boldsymbol{u}_{EKF1}) = \begin{bmatrix} p_{LPF} \\ q_{LPF} \end{bmatrix} \tag{5.44}$$

This would imply that both the pitch and bank angle estimates are affected by angular rate biases. However, it can be recalled that unlike the time update equations, the measurement update equations of EKF1 take gravity into account. At straight and level flight, the calibrated accelerometer measurements are equal to the opposite of the gravity vector:

$$\begin{bmatrix} y_{accel_x} \\ y_{accel_y} \\ y_{accel_z} \end{bmatrix} \cong \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \tag{5.45}$$

Now, if the bank angle ϕ is assumed to be about 0, the measurement to state relationship becomes:

$$\boldsymbol{H}_{EKF1}(\boldsymbol{x}_{EKF1}, \boldsymbol{u}_{EKF1}\ ) = \begin{bmatrix} (qV_{a,LPF} + g)\sin\theta \\ rV_{a,LPF}\cos\theta - pV_{a,LPF}\sin\theta \\ (-qV_{a,LPF} - g)\cos\theta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} = \begin{bmatrix} y_{accel_x} \\ y_{accel_y} \\ y_{accel_z} \end{bmatrix} \tag{5.55}$$

The best solution to this system of equations is a pitch angle close to zero. The top equation of the above relationship is only satisfied when $\theta = 0$. If θ=0, the second two relationships are also closely satisfied as long as the centripetal acceleration terms ($rV_{a,LPF}$ and $qV_{a,LPF}$) are small, which is indeed the case in straight and level flight. Thus, when EKF1 finds an optimal solution for the pitch angle, it is always very close to zero. Additional simulations that were performed that are not presented here show that if the centripetal acceleration terms are very large, such as with angular rate biases of 50 degrees per second, the pitch angle estimate can become off by 1-5 degrees.

Conversely, if the pitch angle θ is assumed to be zero, the measurement to state relationship becomes:

$$\boldsymbol{H}_{EKF1}(\boldsymbol{x}_{EKF1}, \boldsymbol{u}_{EKF1}\ ) = \begin{bmatrix} 0 \\ rV_{a,LPF} - g\sin\phi \\ -qV_{a,LPF} - g\cos\phi \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} = \begin{bmatrix} y_{accel_x} \\ y_{accel_y} \\ y_{accel_z} \end{bmatrix} \tag{5.56}$$

The best solution to this system of equations is a non-zero bank angle ϕ, as solving these equations for ϕ relates ϕ to only the angular rates and gravity. By finding a minimum error estimate to these equations, EFK1 continuously "corrects" the roll angle to this non-zero value that incorporates the incorrect angular rates. That is why the gyro bias drifts appear in the bank angle estimate but not the pitch angle estimate in this simulation. The time update equations add the angular rate errors to the estimate, but the measurement update equations corrects for it, resulting in different behavior for the two different Euler angles. Finally, it is important to note that this behavior is only the case when the estimated pitch and bank angle are close to zero. In other commanded maneuvers, ϕ and θ can't always assumed to be zero.

5.2.1.3 Benchmark 2: Coordinated Turn

Next, below in Figure 5.6 are the results for simulating state estimation method 1 in zero wind. Just like the previous section, gyro rate bias drifts of $\frac{1}{30}\frac{degrees}{second^2}$ on each axis were included.

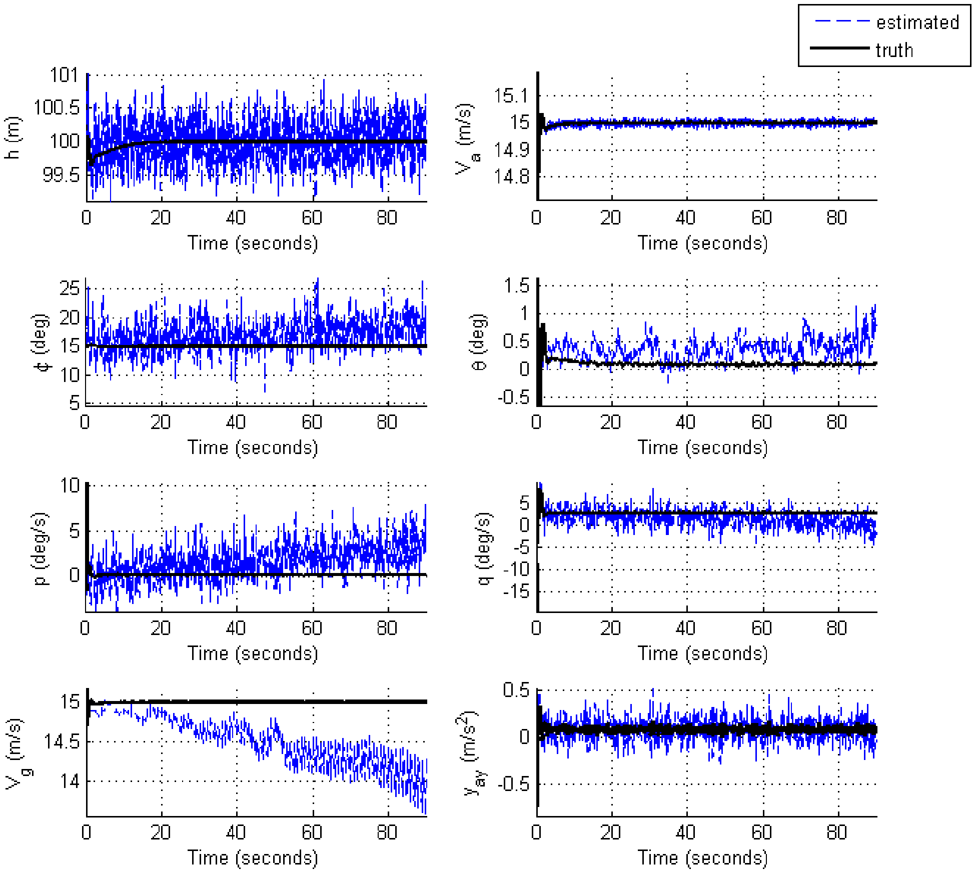


Figure 5.6: Method 1 Results for Benchmark 2 in Zero Wind

Table 5.5: Error Statistics of Method 1 State Estimation for Benchmark 2 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	-0.02615	0.267583	0.215338	0.160962	0.000014	0.983516
$V_a\ (m/s)$	0.000005	0.007345	0.00556	0.004799	0.000002	0.112739
$\phi\ (deg)$	-1.99478	2.543703	2.59284	1.930397	0.00045	11.9702
$\theta\ (deg)$	-0.25169	0.272464	0.285705	0.236546	0.000138	3.891958
$p\ (deg/s)$	-1.5435	1.869367	1.962648	1.422945	0.000645	8.955241
$q\ (deg/s)$	1.181991	1.7954	1.705016	1.308937	0.00027	15.87927
$V_g\ \ (m/s)$	0.476404	0.310111	0.479903	0.304668	0.000072	1.433798
$y_{a_y}\ (m/s^2)$	-0.00378	0.096415	0.075702	0.059823	0.000002	0.710144

These results are very similar to the benchmark 1 results in Figure 5.3. Even the standard deviations of the error in attitude are very similar. However, it can be noticed that there is now a steady-state constant error in the pitch angle. This is due to the assumption that the aircraft is in straight and level *unacceleарated* flight. In a coordinated turn, there is an increase in the z-axis component of the measured acceleration. The measurement model assumes that an increase in vertical acceleration is due to a change in the orientation of the gravity vector and thus due to an increase in the pitch angle. In a coordinated turn, the vertical acceleration increases due to the turn, and not a change in the pitch angle. In mathematical terms, equation (5.10) assumes that the body vertical component of inertial velocity is a function of the airspeed and the sine of the pitch angle. In benchmark 2, the body vertical component of the velocity is non-zero while the pitch angle is zero, which breaks this assumption. This behavior is illustrated in Figure 5.7.

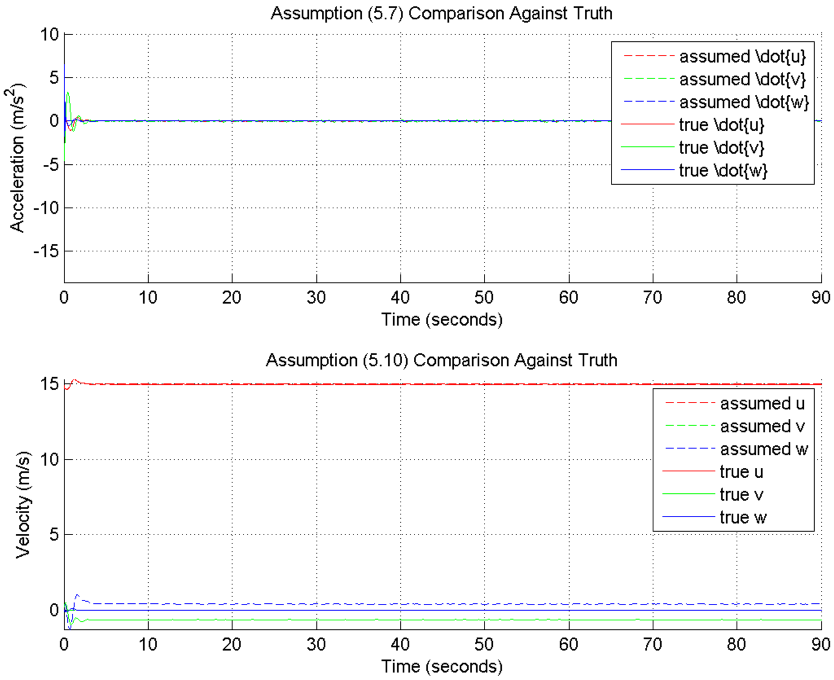


Figure 5.7: Validity of Method 1 Assumptions in Benchmark 2 and Zero Wind

Figure 5.7 shows that the assumptions made in method 1 in order to back out attitude directly from the transport theorem equations without a-priori data are correct in this case, with the exception of the body frame inertial velocity component w. The fluctuations seen in the first second of the plot are initial fluctuations in angular rates that the autopilot induces at the beginning of the simulation, and can be ignored for now until these dynamics are later analyzed. The benchmark 1 simulations previously discussed had zero error on all of these assumptions, and more errors in these assumptions will be discussed later.

Last, for comparison, Figure 5.8 below contains the visual results of a benchmark 2 simulation in zero wind and zero gyro bias drift.

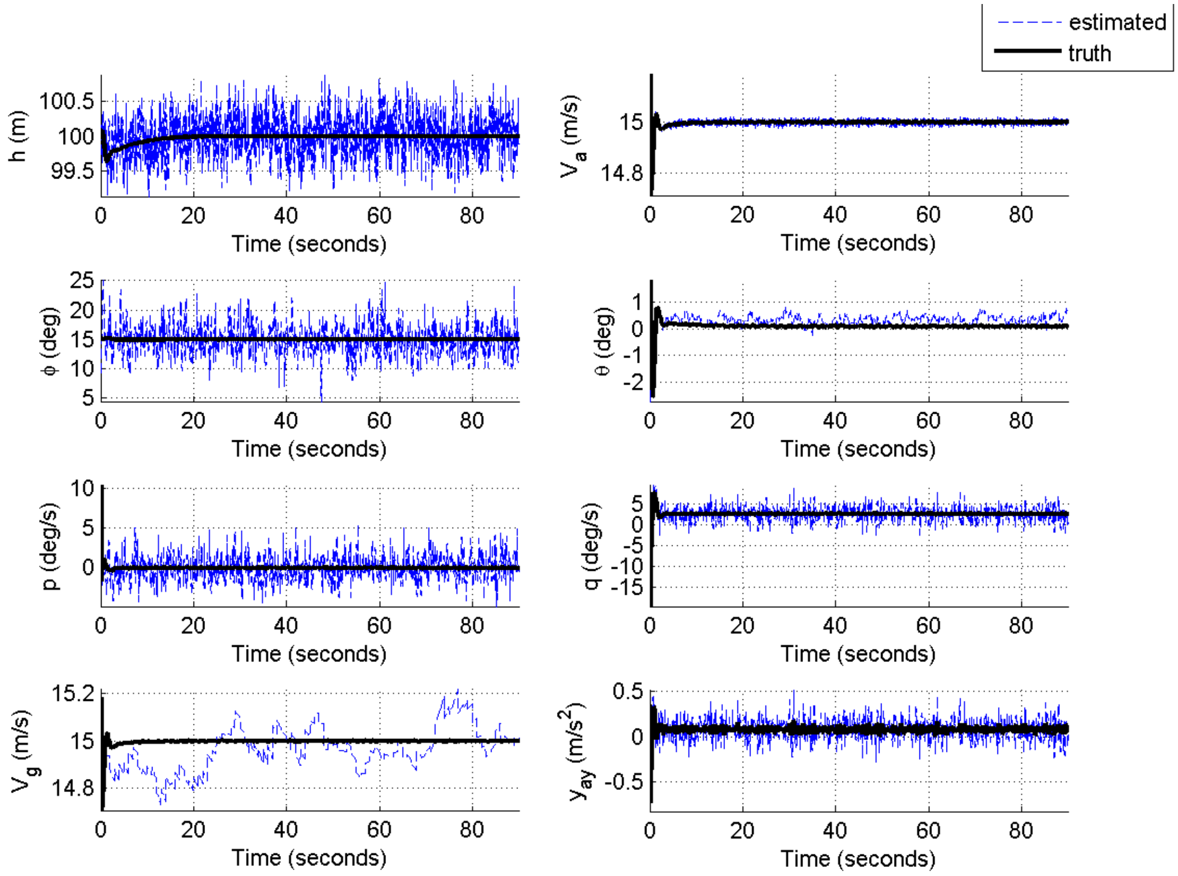


Figure 5.8: Method 1 Results for Benchmark 2 in Zero Wind and Zero Gyro Bias Drift

As expected, the drift in V_g and ϕ disappears when gyro bias drift is removed, but the steady state error in the pitch angle remains.

5.2.1.4 Benchmark 3: Wings Level Ascent

Now, the last benchmark maneuver that is simulated is a wings level ascent from zero altitude to the commanded altitude. Figure 5.9 below contains the state tracking results of estimation method 1 during this maneuver in zero wind. To show the effects of rate gyro bias drift, for just this benchmark, the drift rate was accelerated to $\frac{3}{20}\frac{degrees}{second^2}$ so that the gyro measurements drifted by 3 degrees over the time it took for the aircraft to climb to its commanded altitude.

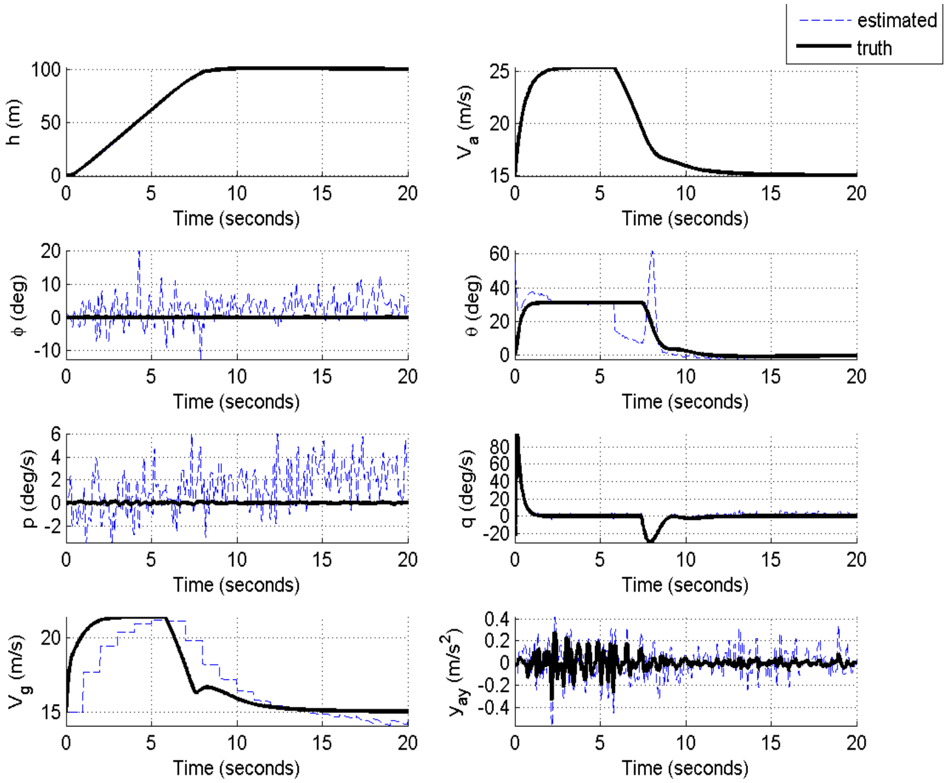


Figure 5.9: Method 1 Benchmark 3 Results in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	0.365771	0.574592	0.519614	0.440316	0.000137	2.066185
$V_a\ (m/s)$	0.000456	0.038864	0.018327	0.034272	0.000004	0.265422
$\phi\ (deg)$	-2.7123	3.614769	3.661495	2.648307	0.00206	20.19237
$\theta\ (deg)$	0.732737	8.955657	4.204742	7.940546	0.001127	59.16003
$p\ (deg/s)$	-1.36551	1.825752	1.86311	1.313774	0.001403	6.191559
$q\ (deg/s)$	-1.61983	2.688693	2.260978	2.177071	0.001985	28.90685
$V_g\ (m/s)$	0.174749	1.56939	1.104937	1.127849	0.001232	5.173123
$y_{a_y}\ (m/s^2)$	-0.00692	0.092843	0.073451	0.057185	0.000001	0.266522

Figure 5.10: Error Statistics for Method 1 in Benchmark 3 and Zero Wind

These results highlight a major inherent flaw in this state estimation method. In the attitude determination step via the simplified transport theorem, two main assumptions were made. Equation (5.7) presents the assumption that the aircraft was not accelerating along its body axes. This is not the case during a climbing maneuver using the total energy controller, as the total energy controller increases airspeed during the first half of the climb, and then decreases it as it nears its commanded total energy. This is in direct violation of the assumption that $\dot{u} = 0$. The second assumption of equation (5.10) assumes that the aircraft is close to level flight. By taking pitch to be the same as the angle of attack, it is assuming that the vector of the incoming air $\boldsymbol{V}_a$ is parallel to the ground. This is not accurate in the climb scenario because in a climb, the angle of attack is not large, and the incoming air is still almost entirely opposite of the direction of travel (parallel to the body x frame). Equation (5.10) claims that in a climb w is large, where in reality, it is still near zero. Figure 5.11 compares the assumed values of the variables determined in equations (5.7) and (5.10) with their true values during this simulation.

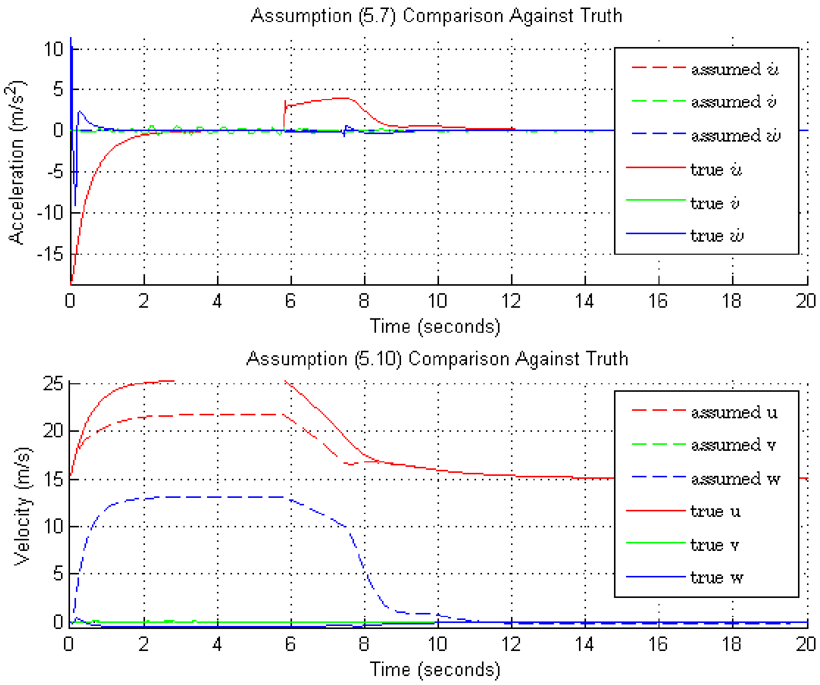


Figure 5.11: Method 1Assumption Validity for Benchmark 3 in Zero Wind

The effect of the errors introduced by these two assumptions can be seen directly in the results in Figure 5.9. However, the effect of each assumption can be measured separately by artificially providing method 1 the true values for u, v, and w. This can be done by replacing the measurement model of equation (5.30) with equation (5.57) below.

$$\begin{bmatrix} y_{a_x} \\ y_{a_y} \\ y_{a_z} \end{bmatrix} = \begin{bmatrix} \hat{q}w - \hat{r}v + g\sin\hat{\theta} \\ \hat{r}u - \hat{p}w - g\cos\hat{\theta}\sin\hat{\phi} \\ \hat{p}v - \hat{q}u - g\cos\hat{\theta}\cos\hat{\phi} \end{bmatrix} \tag{5.57}$$

Note that all variables in equation (5.57) are method 1's estimated values or inputs, and only u, v, and w are the true values. Also take note that this is the same as the full transport theorem equation (5.6), but still retaining the zero body frame acceleration assumption of equation (5.7). The results of a simulation of this modification are shown in Figure 5.12.

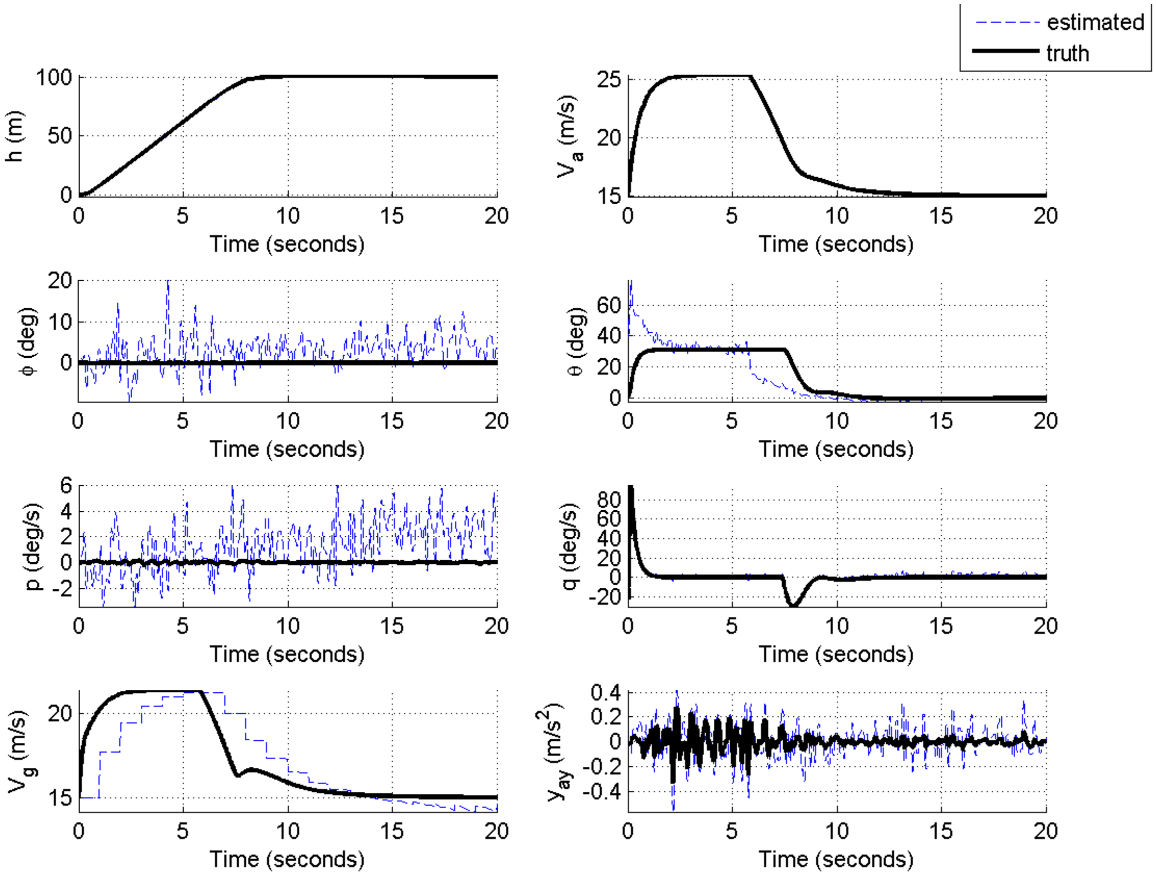


Figure 5.12: Method 1 State Estimation with True Inertial Velocity Values

Figure 5.12 seems to indicate that at least some of the error in the pitch estimate is due to the non-zero $\dot{u}$, as pitch angle errors only occur when $\dot{u}$ is non-zero. This makes sense, as the simplified transport theorem relationship (5.11) must account for excess acceleration in the x direction by adjusting pitch, as roll is not dependent on x axis acceleration. As shown in Figure 5.12, when the acceleration ceases about 3 seconds into the simulation, the estimate for the pitch angle begins to approach its true value. However, when airspeed begins to decrease as the aircraft approaches its target altitude, and before the pitch is decreased, the estimate for pitch becomes too low.

Now, it can be noticed that the spike in estimated bank angle that appears 8 seconds into the simulation occurs at the same time as an abrupt period of non-zero pitch rate q. It can also be noticed that the spike does not appear in the true u, v, w simulation in Figure 5.12, and that at 8 seconds in the simulation, there is an error in the estimated u, v, and w. It makes sense that this error goes away when the true inertial velocity components are used to calculate the centripetal acceleration terms, as the only time those terms are non-zero is when the angular rates are non-zero. With excess magnitude on the right hand side of equation (5.11), the change must be accounted for in the estimate of the attitude. Because it is the pitch rate that is now non-zero, the first equation in (5.11) is used to indicate that pitch should be much larger.

At the beginning of the simulation, the pitch rate is positive, and according to the first equation in (5.11), the subsequent pitch angle should be smaller. Interestingly, the change in estimated pitch angle due to the pitch rate is the opposite sign of the change in pitch due to inaccurate measurement of $\dot{u}$ both at both location in the simulation. This makes sense as both a non-zero X axis acceleration and a positive pitch rate are equivalent to an increase in pitch with respect to the gravity vector. When one of those equivalent increases in pitch are taken away, such as in Figure 5.12, the estimated pitch angle will be increased.

This behavior could be a key area for improving upon this state estimation method by future work. This analysis shows that the state estimates are very sensitive to maneuvers where the change in body frame velocity components are non-zero or there are significant angular rates that cause large centripetal acceleration terms in the transport theorem. This is a large part of the reason why this method performs so poorly in wind, which is about to be discussed.

5.2.2 Dryden Wind

Now that the performance of method 1 has been characterized and the causes of error found, the same simulations will be performed in a Dryden wind field to see how these errors compound and new sources of error appear. For more information on the Dryden wind field that is used in these simulations, see section 4.3.4.

5.2.2.1 Benchmark 1: Straight and Level

Figure 5.13 below contains the state estimation results of method 1 under straight and level flight conditions in Dryden wind.

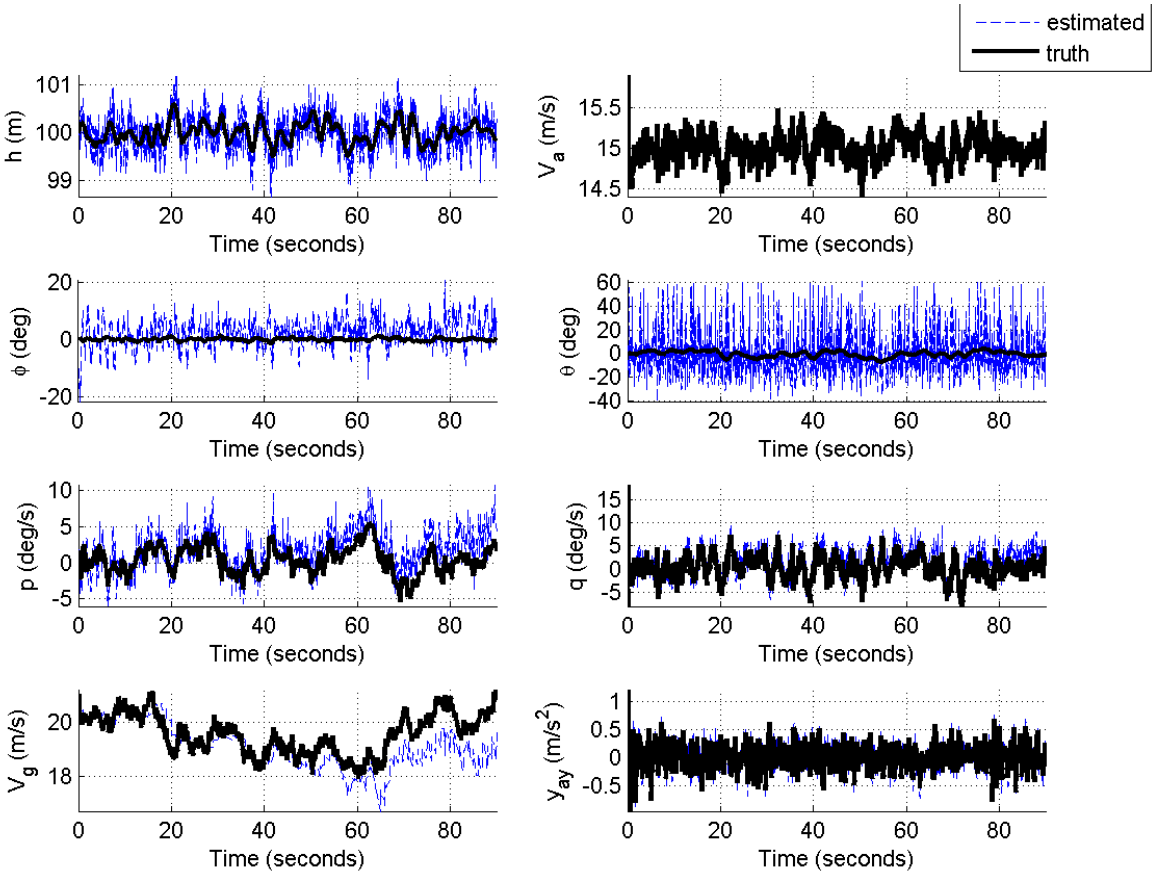


Figure 5.13: Method 1 Results for Benchmark 1 in Dryden Wind

Table 5.6: Error Statistics for Method 1 State Estimation for Benchmark 1 in Dryden Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	0.009912	0.263184	0.210142	0.158744	0.000028	1.017485
$V_a\ (m/s)$	-0.00013	0.031237	0.019976	0.024014	0.000002	0.280067
$\phi\ (deg)$	-2.48169	5.422106	4.67493	3.701529	0.00012	26.76654
$\theta\ (deg)$	-0.08141	6.245348	4.825204	3.965579	0.000603	29.17006
$p\ (deg/s)$	-1.54382	2.015224	2.046104	1.502586	0.000406	7.948609
$q\ (deg/s)$	-1.56719	1.967791	2.073747	1.423953	0.000015	7.773666
$V_g\ (m/s)$	0.422333	0.676247	0.602575	0.522072	0.000118	3.267293
$y_{a_y}\ (m/s^2)$	-0.00368	0.107917	0.082943	0.069133	0.000019	1.100337

When comparing these results to the zero wind results for benchmark 1 in section 5.2.1.2, it should be noticed that state estimates become significantly noisier. The mean of the errors for states other than the ones previously shown to be affected by the gyro bias drifts (h, V_a, θ, and y_{a_y}) remain close to zero.

The mean of the error of the gyro bias drift affected states (p, q, V_g, and ϕ) are similar to what was seen in the zero wind simulation for benchmark 1. These statistics show that in this case, the wind is only causing an increase in the standard deviation of the error.

To investigate why this occurs, it can first be recalled that the Dryden wind model produces disturbances at a spectrum of frequencies to both the wind speed and angular rate disturbances on the aircraft. The wind speed oscillations create translational aerodynamic forces on the aircraft that in turn affect the accelerations measured by the accelerometers. The rate disturbances directly put large oscillations on the angular rates of the aircraft, which affect both the estimated angular rates and the accelerometers due to the centripetal acceleration terms in the rigid body dynamics equations. As Figure 5.14 shows, these disturbances on the aircraft's motion create errors in assumption (5.7) for zero change in body frame inertial velocity components and in assumption (5.10) for the body frame inertial velocity components being strictly relative to the air velocity.

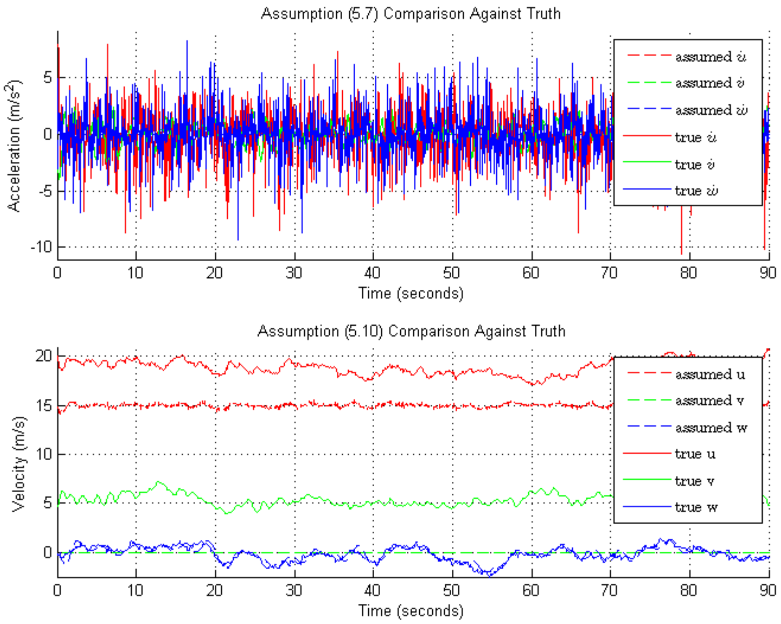


Figure 5.14: Assumption Validity for Benchmark 1 in Dryden Wind

Figure 5.14 indicates that the equation for w in (5.10) is actually fairly accurate. This makes sense because the wind experienced in the vertical direction is small, as shown in Figure 4.4, and because the aircraft is flying only in small disturbances about level flight. Because the changes in pitch due to the wind is small, and the dynamics of the aircraft point it into the wind, the pitch angle changes closely with the vertical wind component. However, Figure 5.14 indicates that the estimates for u and v are not accurate. This makes sense, because the aircraft is always pointed into the wind with zero sideslip, meaning that the aircraft is flying at an airspeed V_a relative to the wind, and not the ground. The error seen on u is due to the aircraft not travelling at V_a relative to the ground, and the constant error seen on v is due to the aircraft having a non-zero crab angle. If the aircraft were flying at a zero crab angle (directly into or away from the wind), then v would be close to its true value. Nonetheless, if the average values of the error are taken away from the u and v estimates, a highly variable component would remain. This is simply due to variations of the wind speed and its effect on the ground velocity of the aircraft. Finally, below are the results of using the true values of u, v, and w in method 1, as is depicted in equation (5.57), for benchmark 1 in Dryden wind.

Table 5.7: Error Statistics for Benchmark 1 in Dryden Wind with True Inertial Velocity Values

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	-0.00465	0.258631	0.203887	0.159175	0.000024	1.396879
$V_a\ (m/s)$	-0.00029	0.031291	0.020139	0.02395	0.000001	0.272728
$\phi\ (deg)$	-3.11832	6.752448	5.82259	4.62755	0.000946	32.69921
$\theta\ (deg)$	-0.72317	6.829383	5.289714	4.379421	0.002258	39.27843
$p\ (deg/s)$	-1.54382	2.015224	2.046104	1.502586	0.000406	7.948609
$q\ (deg/s)$	-1.56719	1.967791	2.073747	1.423953	0.000015	7.773666
$V_g\ \ (m/s)$	0.439593	0.706454	0.629366	0.544238	0.000004	2.95929
$y_{a_y}\ (m/s^2)$	-0.00368	0.107917	0.082943	0.069133	0.000019	1.100337

These results are almost identical to the results achieved with method 1's original estimates for u, v, and w. The only distinguishable difference is there is actually a small *increase* in the standard deviation of the error in pitch and bank. This makes sense, because it should be recalled from equation (5.11) that the pitch and bank angles are calculated assuming that the angle of the gravity vector and the centripetal acceleration terms are the only pieces of information that affect acceleration. When $\dot{u}$, $\dot{v}$, and $\dot{w}$ are not zero, as Figure 5.14 shows, the attitude estimates will suffer regardless of whether the centripetal acceleration terms are correct or not. The values of the centripetal acceleration terms determined by method 1 were smaller in magnitude than the actual estimates. When the true values were added back in, they only added more uncertainty into the relationship, resulting in a larger variance of the attitude estimates.

To further back the notion that the acceleration measurement errors are the source of the error, the instantaneous values of acceleration can be directly mapped to individual fluctuations in the estimated pitch and bank angles. Figure 5.15 is a zoomed in version of the original method 1 simulation in Dryden wind.

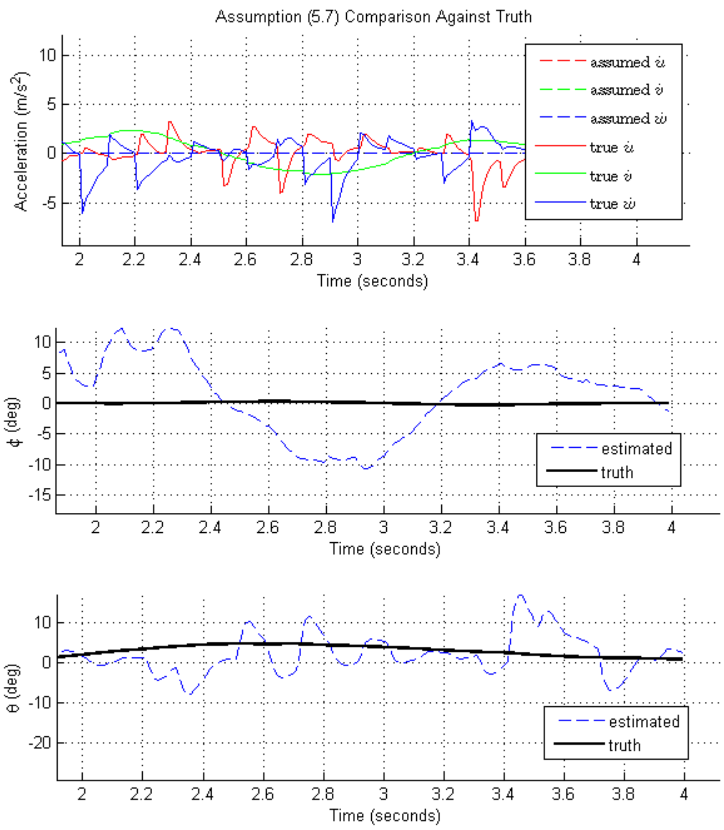


Figure 5.15: Assumption (5.7) to Estimated Attitude Zoom Comparison

Figure 5.15 highlights the direct correlation between the actual value for $\dot{v}$ and the bank angle ϕ. It also highlights the correlation between the combination of magnitude of acceleration in the body Z axis, $\dot{w}$, and the magnitude of acceleration in the -X axis, $-\dot{u}$, with the pitch angle θ. In conclusion, this illustrates that reasoning that assumption (5.7) is the main source of the increased error seen in this windy scenario.

As a tie in for method 2, and for the possibility of future work, this analysis shows that methods of finding information on the rate of change of the body frame velocity components would be very beneficial. Outside of this document, simple tweaks like artificially increasing expected accelerometer sensor noise and removing the low pass filter were tried, but neither had much effect on the quality of the estimates. Increasing the expected accelerometer noise only decreased the frequency of the random deviations in the attitude errors, and did not modify the magnitude.   Removing the low pass filters did not have any noticeable affect because, likely because the low pass filter cutoff frequencies used in this document were already tweaked by trial and error to attitude estimate with the least error. However, it was noticed that if the cutoff frequencies of the low pass filters were set very low, the error oscillations would decrease, but the error magnitude would increase.

5.2.2.2 Benchmark 2: Coordinated Turn

Figure 5.16 below contains the results of simulating benchmark 2 for a coordinated turn in the Dryden wind field.

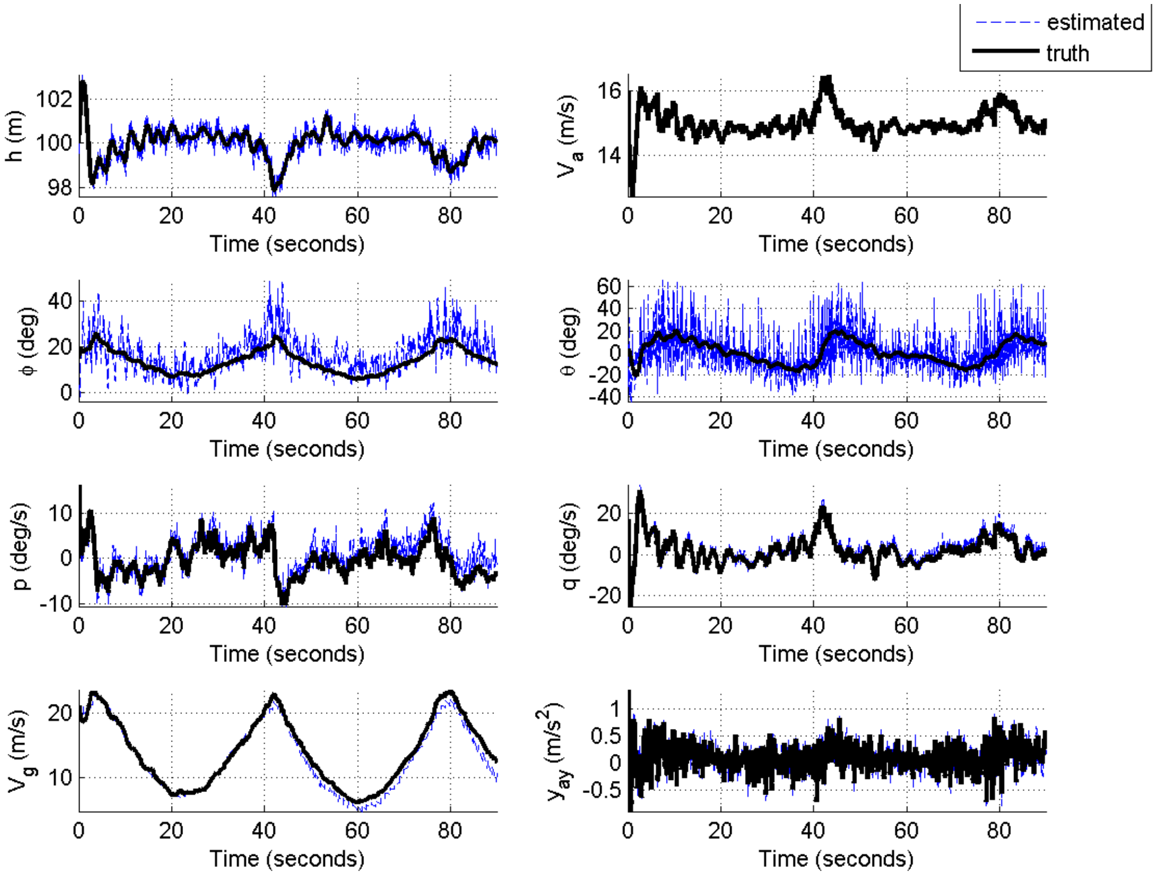


Figure 5.16: Method 1 Results under Benchmark 2 and Dryden Wind

Table 5.8: Error Statistics for Method 1 under Benchmark 2 and Dryden Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	0.008504	0.27609	0.22095	0.165751	0.000038	1.04847
$V_a\ (m/s)$	0.000026	0.027756	0.017566	0.02149	0	0.261154
$\phi\ (deg)$	-3.41256	5.397888	5.010868	3.958847	0.001032	25.33505
$\theta\ (deg)$	-0.08103	6.011813	4.219042	4.283242	0.000336	39.69093
$p\ (deg/s)$	-1.5443	1.881663	1.966788	1.434262	0.000155	10.47874
$q\ (deg/s)$	-1.56605	1.88061	2.018227	1.384117	0.000825	10.2309
$V_g\ \ (m/s)$	0.583431	0.783121	0.78459	0.581427	0.000096	2.861224
$y_{a_y}\ (m/s^2)$	-0.00369	0.106615	0.082276	0.067899	0.000003	1.447511

The first aspect of these results that should be noted is that the estimated bank angle ϕ and ground speed V_g still exhibit the signature of degraded estimates from gyro bias drift. In fact, the mean error that is listed in Table 5.8 for benchmark 2 in wind is almost exactly the same as the mean error observed in benchmark 1 for straight and level flight in wind, which is in Table 5.6. This indicates that a coordinated turn in wind does not create any more steady state error for bank angle.

The second thing that should be noticed from these results is the oscillation in the true bank angle. This oscillation in bank angle is due to the autopilot's reliance on the ground speed to determine the course angle rate as depicted in equation (4.4). Because the aircraft is kept at a constant airspeed, the ground speed varies as the aircraft performs its circular maneuver in the wind field. Because the ground speed is different at different locations in the turn, equation (4.4) dictates that in order to maintain a coordinated turn, the commanded bank angle must also different at different locations in the turn. This oscillation in bank angle is not perfectly sinusoidal because the autopilot will cause the aircraft to turn slower when the airspeed is lower. It can also be noted that the oscillation in pitch angle that was observed is due to the autopilot's inability to maintain altitude in a sharp turn where there is varying pitch rate. Studying in more detail and adjusting the autopilot's behavior in such a turn is outside of the scope of this research, and it also gives a variety of pitch angles to assess the performance of the state estimator at.

That being said, it can be noticed that steady state pitch error that was observed in benchmarks 2 in zero wind is not very detectable in these results. That error is still in these results, as just as in the zero wind condition in a coordinated turn, the acceleration measured opposite of the Z axis is larger than gravity. The measurement equation (5.11) captures this excess acceleration in the pitch angle because even if assumption (5.7) holds, the estimated vertical velocity w is inaccurate. This inaccuracy is depicted in Figure 5.17 below.

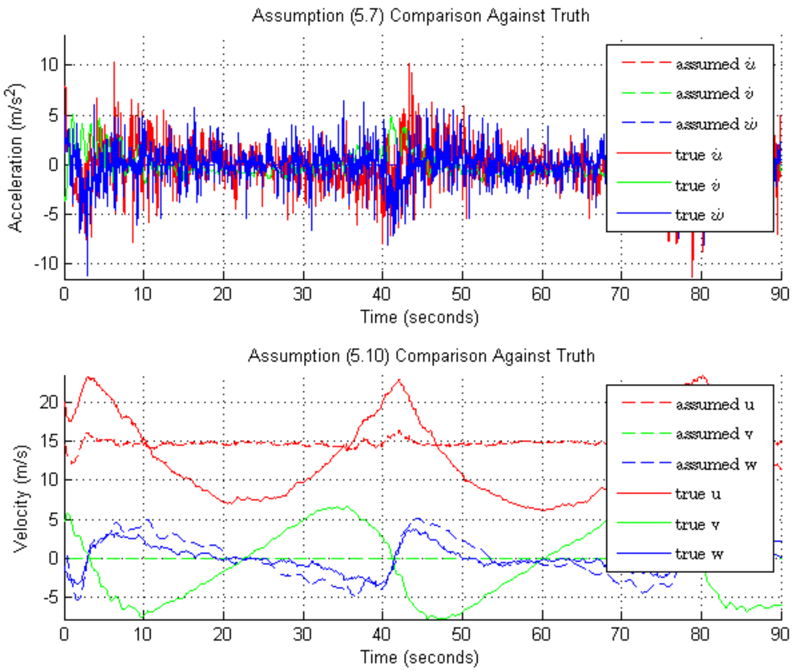


Figure 5.17: Method 1 Assumption Validity under Benchmark 2 in Dryden Wind

In addition to the assumption (5.10) error for w, Figure 5.17 shows how the true values for u and v change as the aircraft turns through the wind field. Just like discussed in the previous section, assumption (5.10) is only valid in the zero wind case. However, as also discussed in the previous section, the effects of these errors are miniscule in comparison to the errors in assumption (5.7).

The steady state error in pitch isn't very detectable here because of the large standard deviation of the error. The standard deviations of the attitude errors in benchmark 1 in wind is essentially identical to the standard deviations of the attitude here, but benchmark 1 did not have any steady state error in pitch, just like benchmark 1 in zero wind.

Next, the magnitude of the estimation error for benchmark 2 in wind are nearly identical to the state estimation error for benchmark 1 in wind. Similarly, the magnitude of the estimation error for benchmark 2 in zero wind was essentially identical to the magnitude of the estimation error for benchmark 1 in zero wind. While the increase in error in wind for both benchmarks has been attributed to assumption (5.7), it could be interesting to see what affect removing the angular rate biases from the wind would have on the estimation performance. Table 5.9 below contains statistics of such a simulation.

Table 5.9: Error Statistics for Method 1 in Benchmark 2 in Dryden Wind with No Angular Rate Disturbances

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	0.014131	0.273638	0.218413	0.165432	0.000003	1.170085
$V_a\ (m/s)$	-2.7E-05	0.028042	0.017927	0.021562	0.000001	0.263253
$\phi\ (deg)$	-3.71141	5.792228	5.395587	4.26736	0.001475	26.56745
$\theta\ (deg)$	-0.06719	6.169774	4.370888	4.354749	0.000866	37.16324
$p\ (deg/s)$	-1.54406	2.017114	2.047613	1.50331	0.000058	8.126008
$q\ (deg/s)$	-1.56714	1.964096	2.071286	1.422386	0.000097	7.632921
$V_g\ (m/s)$	0.492612	0.750946	0.69957	0.563171	0.000349	3.324215
$y_{a_y}\ (m/s^2)$	-0.00369	0.106183	0.081806	0.06779	0.000007	1.447511

It should be noticed that these results are almost identical to the results for benchmark 2 in Dryden wind with the angular rate disturbances included. This makes sense, as it was previously proven that the accuracy of the centripetal acceleration terms of the transport theorem don't have much relative bearing on the accuracy of the solution when compared to

While the rate disturbances do affect the true values for $\dot{u}, \dot{v}$, and $\dot{w}$ in the rigid body dynamics of the aircraft, this leads one to wonder why the changes in those quantities are what they are. Part of those values is the wind, but part of those values is also the autopilot's reaction to the wind and its reaction to follow its commanded course. Unfortunately, it's not really possible to separate the state estimation from the closed loop behavior, as fixing the aircraft's motion to a certain path would violate the reality that the aircraft is perturbed by the wind and would negate any of those effects on the state estimation. However, it can be recognized the more abrupt changes are that the autopilot makes, the more state estimation method 1 would be affected due to the subsequent change in the rate of change of the body components of the inertial velocity vector.

$$\begin{bmatrix} f_x/m \\ f\_y\ /m \\ f_z/m \end{bmatrix} = \begin{bmatrix} \dot{u} + qw - rv + g\sin\theta \\ \dot{v} + ru - pw - g\cos\theta\sin\phi \\ \dot{w} + pv - qu - g\cos\theta\cos\phi \end{bmatrix} \tag{5.58}$$

Equation (5.58) is the equation of motion for a rigid body aircraft. It shows that the rate of change of the body components of the inertial velocity vector, $\dot{u}, \dot{v}$, and $\dot{w}$, are functions of the translational force the aircraft is experiencing, the angular rates of the aircraft, and the attitude of the aircraft. All control mechanisms on an aircraft affect these directly, and so the design of an autopilot may exacerbate or mitigate some of the behaviors observed here. However, the design of the autopilot to mitigate these issues is not the scope of this work. The goal of this work is to bring forward these insights so the designers of a closed loop system can be aware of these concerns.

5.2.2.3 Benchmark 3: Wings Level Ascent

The final simulation of method 1 is for benchmark 3, a wings level ascent, in the same Dryden wind field as the previous simulations.

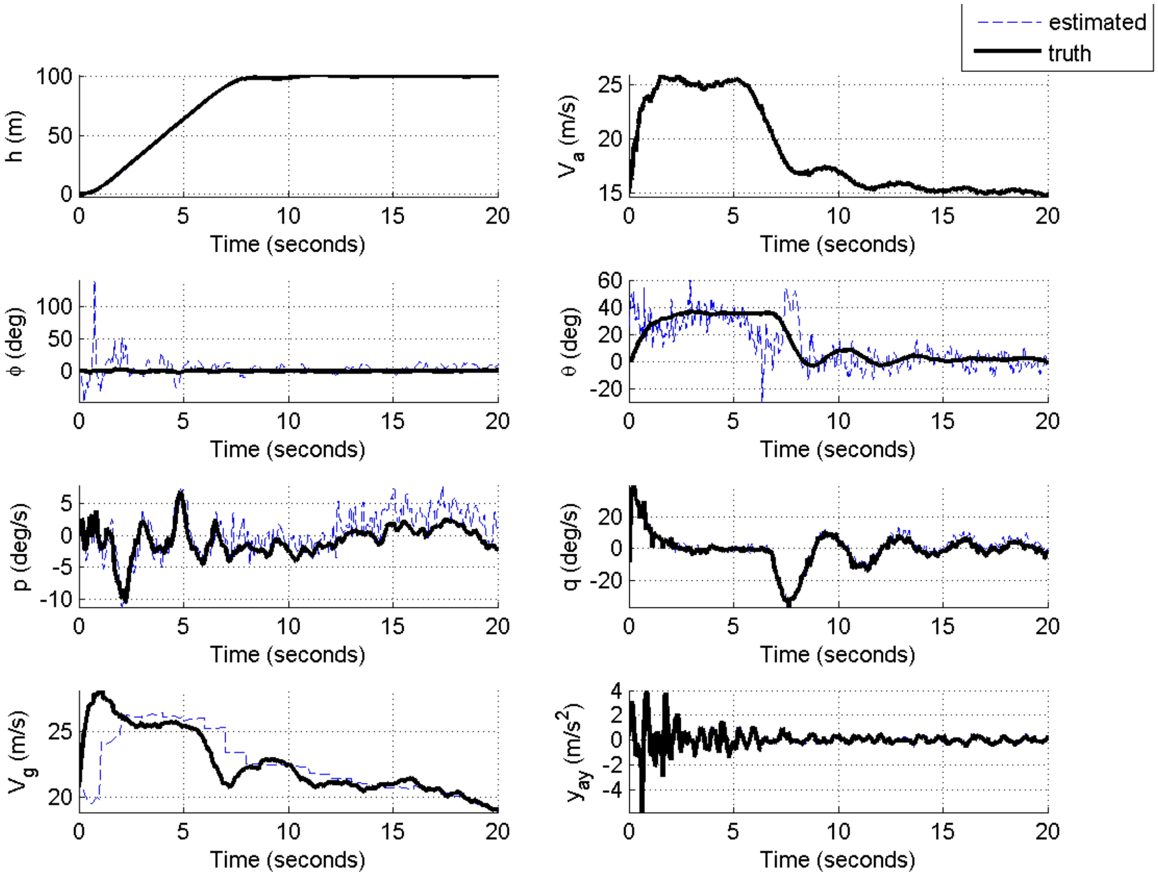


Figure 5.18: Method 1 Results for Benchmark 3 in Dryden Wind

Table 5.10: Error Statistics for Method 1 State Estimation for Benchmark 3 in Dryden Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	0.014131	0.273638	0.218413	0.165432	0.000003	1.170085
$V_a\ (m/s)$	-2.7E-05	0.028042	0.017927	0.021562	0.000001	0.263253
$\phi\ (deg)$	-3.71141	5.792228	5.395587	4.26736	0.001475	26.56745
$\theta\ (deg)$	-0.06719	6.169774	4.370888	4.354749	0.000866	37.16324
$p\ (deg/s)$	-1.54406	2.017114	2.047613	1.50331	0.000058	8.126008
$q\ (deg/s)$	-1.56714	1.964096	2.071286	1.422386	0.000097	7.632921
$V_g\ (m/s)$	0.492612	0.750946	0.69957	0.563171	0.000349	3.324215
$y_{a_y}\ (m/s^2)$	-0.00369	0.106183	0.081806	0.06779	0.000007	1.447511

In simple terms, these results are very similar to the results obtained for benchmark 3 in zero wind; they are just noisier. This state estimation method still exhibits the large pitch angle errors that were found to be associated with non-level flight in zero wind. The mean of the bank angle error in Table 5.10 still indicates a small magnitude drift of the bank angle estimate. The cause of this was determined previously to be the measurement update equations of EKF1. However, like the benchmark 3 simulation in zero wind, the estimate for the ground speed does not become too far off over the course of this short simulation. This makes sense, as over the course of this simulation, the yaw angle error only gets to about 45 degrees, and with GPS measurement updates once per second, the covariance matrix still places plenty of weight on the GPS ground speed measurement.

Last, the standard deviation of the error of the attitude angles is in family with the standard deviations seen in the other two benchmark maneuvers in wind. Figure 5.19 below shows the error of the two assumptions in method 1.

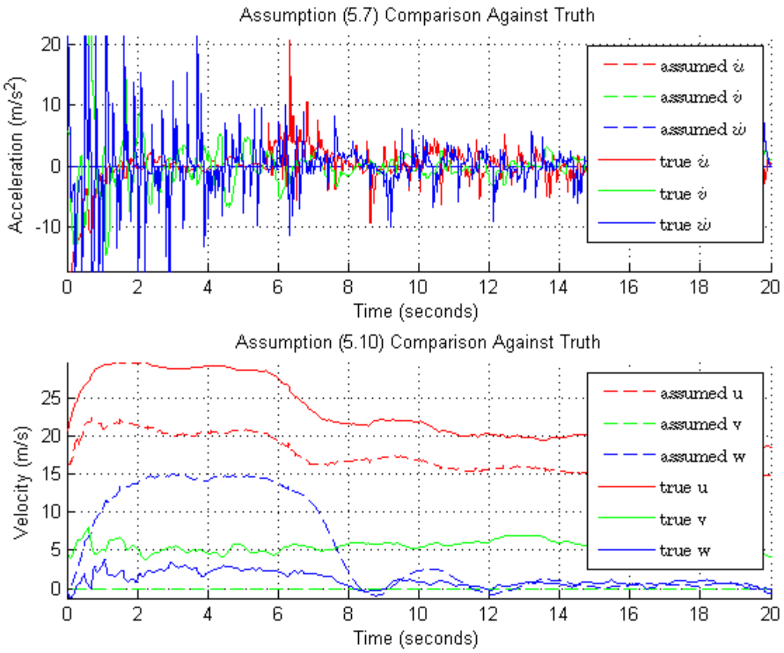


Figure 5.19: Method 1 Assumption Validity for Benchmark 3 in Dryden Wind

The error shown in Figure 5.19 for assumption (5.10) matches the error seen in benchmark 3 in zero wind in Figure 5.11, but has some additional error on u and v. As explained earlier, the error on u and v is due to the aircraft pointing into the wind and having a non-zero crab angle.

Interestingly, the error shown in Figure 5.19 for assumption (5.7) is larger at the beginning of the simulation than it was for both benchmarks 1 and 2 in Dryden wind. Since the wind field is the same, this increase in acceleration must be due to autopilot maneuvers.   By close inspection of the assumption error for benchmark 3 zero wind Figure 5.11, it can be seen that there is also a substantial fluctuation in $\dot{u}$, $\dot{v}$, and $\dot{w}$. This fluctuation is due to the autopilot initializing its PID controllers at the beginning of the simulation. To support that notion, Figure 5.20 below contains the inertial velocity components and their rate of change for a simulation of commanded straight and level flight, and then a 100 meter increase of commanded altitude at 6 seconds into the simulation.

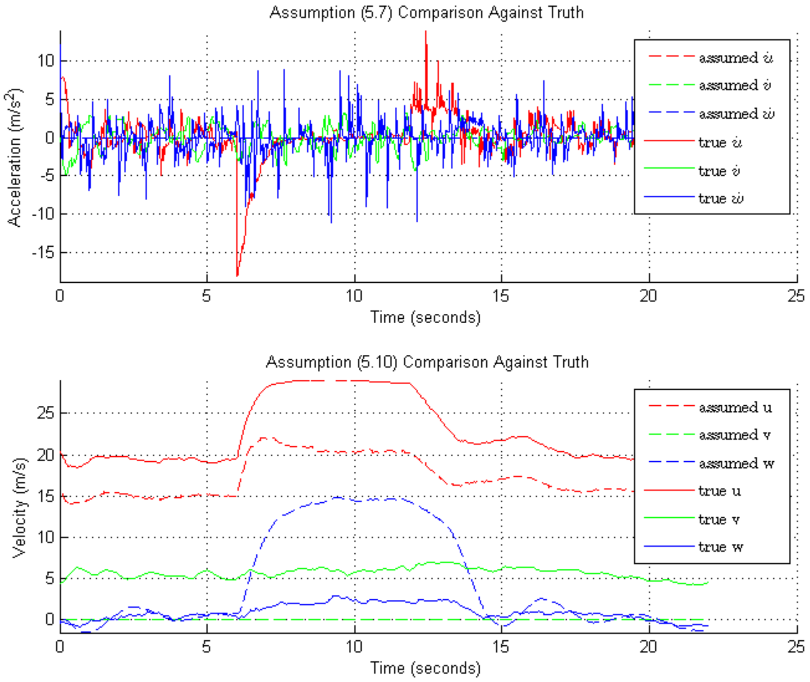
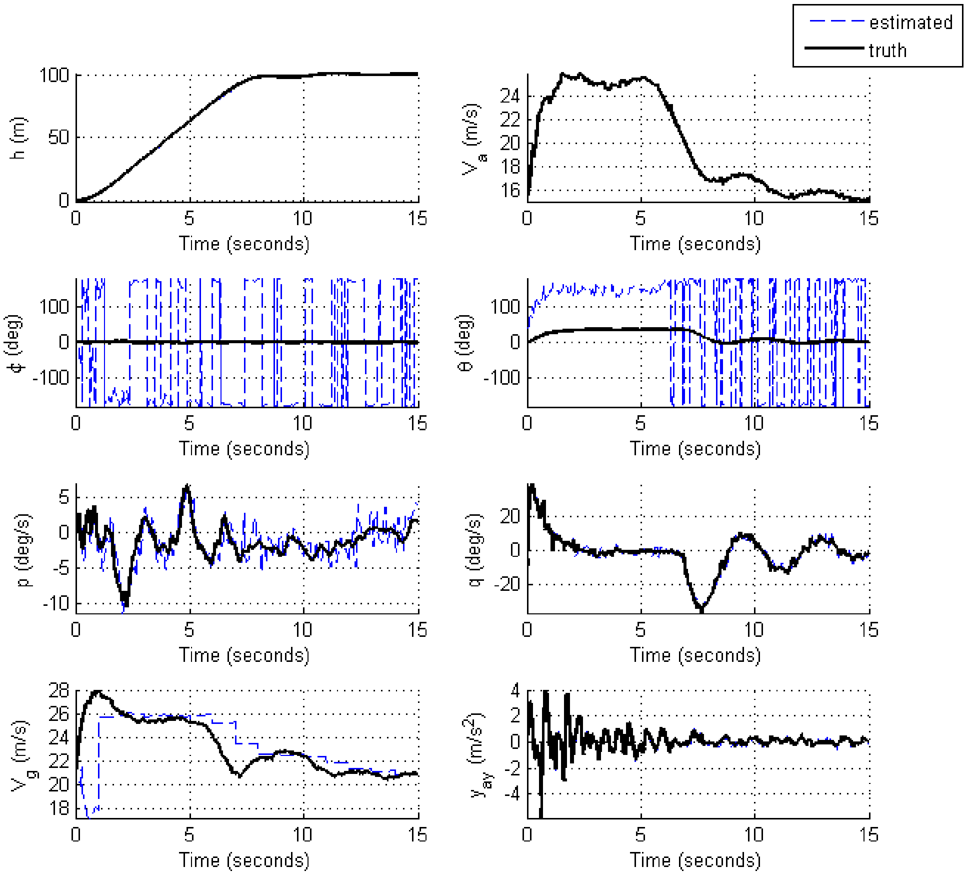


Figure 5.20: Method 1 Assumption Validity for Benchmark 3 in Dryden Wind and Delayed Climb

The accelerations experienced here, though more noisy, are roughly the same magnitude as those seen in Figure 5.11 for benchmark 3 in zero wind. Thus, the large accelerations seen in the original simulation are due to autopilot behavior at the beginning of the simulation.

It would make sense that the large fluctuations at the beginning simulation would cause the bank angle estimate of about 100 degrees, given the effect large errors in assumption (5.7) have been shown to have in the previous simulations. However, it should be noted that like in benchmark 3 in zero wind, if the true values for assumption (5.10) are used, as is depicted in equation (5.57), the estimates for the pitch and bank angles becomes large, that the estimates settle in on ± 180 degrees.



All this indicates is that like in benchmark 3 in zero wind, replacing the estimated inertial velocity components with their true ones cause the error in the attitude solution to become even larger. This makes sense, as like in zero wind, with large errors on both sides of the simplified transport theorem equation (5.11), the estimated pitch angle will be smaller. With the error on the right hand side gone, the pitch angle estimate will be larger. Unfortunately, in this case, EKF 1 isn't able to correct for that large of an error.

5.3 Method 1 Analysis Summary

This section has demonstrated the performance differences of state estimation method 1 between simulations in zero wind and in Dryden wind. It has shown that the performance of method 1 severely degrades in Dryden wind. As examined in detail in this section, the overall reason this state estimation method performs so poorly in wind is the assumptions that were made in developing the state estimation algorithm. A high level summary of the sources and their effects on the states necessary for autopilot control (4.11) is below in Table 5.11.

Table 5.11: Method 1 State Estimation Error Summary

Source of error	**Effects in Zero Wind**	**Effects in Dryden Wind**
Gyro Bias Drift	Drifts in estimates for ϕ and V_g	Drifts in estimates for ϕ and V_g
Assumption (5.7): $(\dot{u} = \dot{v} = \dot{w} = 0)$	Valid assumption in benchmarks 1 and 2. Pitch Errors in benchmark 3.	Increased standard deviation of attitude error in all benchmarks.
Assumption (5.10): $\begin{bmatrix} u \\ v \\ w \end{bmatrix} = V_a \begin{bmatrix} \cos\theta \\ 0 \\ \sin\theta \end{bmatrix}$	Valid assumption in benchmarks 1 and 2. Mixed effect on pitch error in benchmark 3.	No measured impacts in benchmarks 1 and 2. Mixed effect on pitch error in benchmark 3.

Pseudo-measurements (5.27) and (5.28)	EKF2 unable to account for yaw angle changes due to gyro rate bias. Affected desired state V_g.	EKF2 unable to account for yaw angle changes due to gyro rate bias. Affected desired state V_g.
Dryden Angular Rate Disturbances	N/A	No discernable effect on estimation of desired states.

These conclusions are revisited after the sources of error for state estimation method 2 are discovered.

6 Method 2: IMU and GPS Sensor Fusion

6.1 Description

In some ways, method 2 is conceptually simpler than method 1. However, it involves full sensor fusion of the IMU and GPS sensor inputs to form a combined full inertial state estimate in one sensor integration process. This process is demonstrated in [10], illustrated in [6] and [3], and derived in detail in [7] and [20]. However, an overview is given of it here. The method produces no estimate for wind, but it uses a minimal set of sensors to obtain inertial information that is necessary for autopilot control and lays the groundwork for wind estimators to be built on top of this method's state estimates, like is done in [10].

6.1.1 Direct IMU Numerical Integration

Method 2 is accomplished by recognizing that an IMU and GPS can in effect provide redundant measurements of position and velocity. The method works by first letting a strictly IMU based algorithm solve for the 9 states listed in (6.1).

$$\boldsymbol{x}_{IMU} = [p_n\ p_e\ p_d\ u\ v\ w\ \phi\ \theta\ \psi\ ]^T \tag{6.1}$$

Instead of the input-based EKF methodology used in method 1, method 2 solves for these states by using a priori knowledge of the states, which is the measurements of the states from the last estimator iteration. At the first iteration of the estimator, the initial conditions are used as the a priori information. With a priori knowledge, the accelerometer and gyro inputs can be directly numerically integrated to estimate the states $\boldsymbol{x}_{IMU}$. The subscript IMU on the state vector in (6.1) indicates that these are the states that are being determined directly from integration on the IMU sensor measurements.

So, using only gyro measurements, the Euler angle rates are directly calculated as follows in equations (6.2) and (6.3).

$$\begin{bmatrix} \hat{p}_k \\ \hat{q}_k \\ \hat{r}_k \end{bmatrix} = \begin{bmatrix} y_{p_{gyro,k}} + b_{g_p} \\ y_{p_{gyro,k}} + b_{g_q} \\ y_{p_{gyro,k}} + b_{g_r} \end{bmatrix} \tag{6.2}$$

$$\begin{bmatrix} \widehat{\dot{\phi}_k} \\ \widehat{\dot{\theta}_k} \\ \widehat{\dot{\psi}_k} \end{bmatrix} = \begin{bmatrix} 1 & \sin\hat{\phi}_{k-1}\tan\hat{\theta}_{k-1} & \cos\hat{\phi}_{k-1}\tan\hat{\theta}_{k-1} \\ 0 & \cos\hat{\phi}_{k-1} & -\sin\hat{\phi}_{k-1} \\ 0 & \sin\hat{\phi}_{k-1}\sec\hat{\theta}_{k-1} & \cos\hat{\phi}_{k-1}\sec\hat{\theta}_{k-1} \end{bmatrix} \begin{bmatrix} \hat{p}_k \\ \hat{q}_k \\ \hat{r}_k \end{bmatrix} \tag{6.3}$$

This relationship is the transport rate of 3-2-1 Euler angles as provided by equation (3.8). The subscript k indicates that this is the value for the current estimator iteration, and the subscript $k-1$ indicates the a priori value from the last estimator iteration. The values $b_{g_p}, b_{g_q}, b_{g_r}$ are the estimated biases of the gyros. The biases are actually determined later in this process and explained there. They are not obtained form an external calibration procedure. A trapezoidal rule numerical integration is then performed on these calculated Euler rates to get an updated Euler angle estimate. This integration is depicted in equation (6.4).

$$\begin{bmatrix} \hat{\phi}_k \\ \hat{\theta}_k \\ \hat{\psi}_k \end{bmatrix} = \begin{bmatrix} \hat{\phi}_{k-1} \\ \hat{\theta}_{k-1} \\ \hat{\psi}_{k-1} \end{bmatrix} + T_s \begin{bmatrix} \widehat{\dot{\phi}_k} \\ \widehat{\dot{\theta}_k} \\ \widehat{\dot{\psi}_k} \end{bmatrix} \tag{6.4}$$

T_s is the time that has elapsed since the last IMU sample. It could be desirable to use a higher order integration method here, but that would involve performing higher order derivatives of equation (3.8). That would be a problem when second or higher order derivatives of the attitude a priori $(k-1)$ attitude data is needed, as it is not available. Thus, a simple trapezoidal integration is performed, and as will be seen in the results, it is more than sufficient due to the high sampling frequency IMUs provide.

A similar method is performed to obtain the inertial velocities straight from the accelerometer measurements. The full (not simplified) transport theorem relationship derived in equation (5.6) is rearranged to become the relationship that gets the inertial velocity derivatives from a priori inertial velocities and the just determined attitude and angular rates, as depicted by equation (6.5).

$$\begin{bmatrix} \dot{u}_k \\ \dot{v}_k \\ \dot{w}_k \end{bmatrix} = \begin{bmatrix} y_{a_{x,k}} + b_{a_x} \\ y_{a_{y,k}} + b_{a_y} \\ y_{a_{z,k}} + b_{a_z} \end{bmatrix} - \begin{bmatrix} q_{k-1}w_{k-1} - r_{k-1}v_{k-1} + g\sin\theta_k \\ r_{k-1}u_{k-1} - p_{k-1}w_{k-1} - g\cos\theta_k\sin\phi_k \\ p_{k-1}v_{k-1} - q_{k-1}u_{k-1} - g\cos\theta_k\cos\phi_k \end{bmatrix} \tag{6.5}$$

Equation (6.5) can be numerically integrated to obtain the inertial velocities ${}^{B}[V_{B/N}]$ as is done in (6.6).

$$\begin{bmatrix} u_k \\ v_k \\ w_k \end{bmatrix} = \begin{bmatrix} u_{k-1} \\ v_{k-1} \\ w_{k-1} \end{bmatrix} + T_s \begin{bmatrix} \dot{u}_k \\ \dot{v}_k \\ \dot{w}_k \end{bmatrix} \tag{6.6}$$

It will again be seen that first order numerical integration is sufficient. In any case, derivatives of the angular rates are not available. Last, the position of the aircraft relative to the inertial frame $\boldsymbol{p}_{B/N}$ is just the integrated velocity of the aircraft relative to the inertial frame $\boldsymbol{V}_{B/N}$. Since $\boldsymbol{V}_{B/N}$ is currently written in body frame components u_k, v_k, w_k, the body frame components can just be rotated into the inertial frame prior to integration as in equation (6.7).

$$\begin{bmatrix} p_n \\ p_e \\ p_d \end{bmatrix} = \begin{bmatrix} p_{n_{k-1}} \\ p_{e_{k-1}} \\ p_{d_{k-1}} \end{bmatrix} + T_s \boldsymbol{R}_{NB} \begin{bmatrix} u_k \\ v_k \\ w_k \end{bmatrix} \tag{6.7}$$

Now, this obtains an IMU-only estimate of the 9 inertial states listed in (6.1). However, it is subject to integration errors and depends on known sensor biases. It should also be noted that the gyro and accelerometer inputs are not passed through a low pass filter prior to integrating them. This low pass filtering was only done in method 1 to help alleviate errors due to the assumptions made on the flight behavior of the aircraft. No assumptions on the aircraft's motion have been made here.

6.1.2 Sensor Fusion through Kalman Filtering

Now that numerical integration has provided inertial position, inertial velocity, and attitude information, it can be recognized that GPS measurements provide duplicate measurements of some of the states. Nonetheless, a pure IMU solution can never be used indefinitely due to inevitable buildup of errors from numerical integration and sensor noise. This is where sensor fusion comes in. After the IMU based estimation is complete, measurements of error between the IMU and GPS solution are found, as defined in equation (6.8).

$$\boldsymbol{\epsilon}_{IMU-GPS,k} = \begin{bmatrix} \left[\boldsymbol{p}_{B/N,k}\right]_{IMU} - \left[\boldsymbol{p}_{B/N,k}\right]_{GPS} \\ \left[\boldsymbol{V}_{B/N,k}\right]_{IMU} - \left[\boldsymbol{V}_{B/N,k}\right]_{GPS} \end{bmatrix} \tag{6.8}$$

The IMU solution provides the inertial velocities in body B frame components and positions in inertial N frame components. The GPS receiver provided inertial velocities and positions both in inertial N frame components. For easier use later, this error needs to have both the inertial positions and velocities written in the N frame. This conversion is performed by equation (6.9), and makes use of the rotation matrix $\boldsymbol{R}_{NB}$, which is defined in equation (3.3).

$$^{N}\boldsymbol{\epsilon}_{IMU-GPS,k} = \begin{bmatrix} ^{N}\left[\boldsymbol{p}_{B/N,k}\right]_{IMU} \\ \boldsymbol{R}_{NB}\left(\psi_{k,IMU}\right) {}^{B}\left[\boldsymbol{V}_{B/N,k}\right]_{IMU} \end{bmatrix} - \begin{bmatrix} ^{N}\left[\boldsymbol{p}_{B/N,k}\right]_{GPS} \\ ^{N}\left[\boldsymbol{V}_{B/N,k}\right]_{GPS} \end{bmatrix} \tag{6.9}$$

These simple GPS-IMU position and velocity errors are used as the measurements into an EKF that tracks the *errors* between the IMU solution and the perceived true solution. This measurement vector for this EKF is depicted by (6.10).

$$\boldsymbol{y}_{EKF3} = {}^{N}\boldsymbol{\epsilon}_{IMU-GPS} \tag{6.10}$$

The subscript $EKF3$ is used to denote this EKF. EKF 3 is the only EKF used in method 2, but labelling it 3 clearly distinguishes it apart from EKF 1 and EKF 2 from method 1. In sensor fusion, this vector of exact differences between the IMU and GPS solutions in (6.10) is typically referred to as the innovation matrix. Next, the angular rates found from the IMU's gyros are used as inputs to the EKF, as depicted by (6.11).

$$\boldsymbol{u}_{EKF3} = [p_k q_k r_k]^T \tag{6.11}$$

Last, the EKF estimates all of the inertial states shown below in equation (6.12).

$$\boldsymbol{x}_{EKF3} = \left[\delta p_n\ \delta p_e\ \delta p_d\ \delta V_n\ \delta V_e\ \delta V_d\ \delta\phi\ \delta\theta\ \delta\psi\ b_{a_X} b_{a_Y} b_{a_Z} b_{g_p} b_{g_q} b_{g_r}\right]^T \tag{6.12}$$

The δ modifier in the states of (6.12) indicates the estimated error of the specified state between the IMU solution for that state and the true value for that state. For example, δV_e represents the estimate error between the strictly IMU based solution for inertial velocity written in the inertial frame and the true value for inertial velocity. These error states are not the error between the GPS and IMU solutions, as that is the innovation matrix (6.10) that is used as the input to the EKF. GPS also does not provide attitude estimates, so there is no such thing as an attitude difference between IMU and GPS solutions.

The Kalman filter that creates this state vector processes the difference between GPS and IMU position and velocity measurements to also obtain the bias terms in state indices 10-15 of (6.12). These bias terms hold total estimated error in the gyro and accelerometer readings. These error terms can account for sensor imperfections other than just bias, to include time evolution of the bias based on a number of real sensor factors. It can also account for bias due to integration error. Most importantly however, these estimates of the sensor biases are fed back into the IMU solution to compensate for sensor errors. This is depicted in the flow diagram in Figure 6.1.

Even though this is an EKF, to some degree it acts as an Unscented Kalman Filter (UKF) in that it also keeps track of nonlinear error (or mean bias) of its states. Reference [6] alludes to this method uses this augmented state vector because it is “perhaps the correct way” to account for the biases. This could perhaps be implemented as a 9-state Unscented Kalman Filter, but comparing the performance of a UKF and an augmented EKF is outside the scope of this research.

Now, this EKF updates the state vector only at each GPS measurement update. This is done via the relationship in equation (3.12) using the equation (6.13).

$$\boldsymbol{H}(\boldsymbol{x}_{EKF3}, \boldsymbol{u}_{EKF3}) = [diag(\boldsymbol{x}_{EKF3})\ \boldsymbol{0}_{6x9}] \tag{6.13}$$

The function $diag$ builds a diagonal matrix with the components of $\boldsymbol{x}_{EKF3}$ as its diagonal components. This relationship is in fact this simple because the first six states of $\boldsymbol{x}_{EKF3}$ are treated to be the errors between the IMU solution and the GPS measurements. Like just explicitly stated, the states are the errors between the IMU solution and the truth, so here, the GPS measurements are treated as truth. Sensor fusion accomplishes estimating the error states by treating one of the sensor sources as truth, and then uses a Kalman filter to minimize the variance of the error of the error between the non-truth source (IMU) and the true values (GPS). The second six columns of the relation (6.13) are the null matrix (zero) because there is no truth (GPS) information for attitude and IMU sensor biases.

Equation (6.13) causes the Jacobian used in the measurement update equations (3.15) and (3.16) to become equation (6.14).

$$\frac{\partial \boldsymbol{H}_{EKF}}{\partial \boldsymbol{x}_{EKF}} = [\boldsymbol{I}_{6x6}\ \boldsymbol{0}_{6x9}] \tag{6.14}$$

Now, the measurement update relationship in equation (3.15) is used to generate the minimum $\boldsymbol{x}_{EKF3}$ error variance estimate $\boldsymbol{x}_{EKF3,k}$ and equation (3.16) is used to update the error covariance matrix.

However, this newly estimated $\boldsymbol{x}_{EKF3,k}$ is not the $\boldsymbol{x}_{EKF3,k-1}$ that is used in the next EKF 3 update. This is because the outputs of the measurement update of EKF 3, $\boldsymbol{x}_{EKF3,k}$, are used to *correct* the IMU solution. When an estimated minimum variance IMU error state $\boldsymbol{x}_{EFK3,k}$ is generated, it is applied to the IMU estimate by equation (6.15).

$$\begin{bmatrix} \delta u \\ \delta v \\ \delta w \end{bmatrix} = \boldsymbol{R}_{BN} \begin{bmatrix} \delta V_N \\ \delta V_E \\ \delta V_D \end{bmatrix} \tag{6.15a}$$

$$\hat{\boldsymbol{x}}_{fused} = \begin{bmatrix} \hat{p}_n \\ \hat{p}_e \\ \hat{p}_d \\ \hat{u} \\ \hat{v} \\ \widehat{w} \\ \widehat{\phi} \\ \hat{\theta} \\ \widehat{\psi} \end{bmatrix} = \begin{bmatrix} p_{n_{IMU}} - \delta p_n \\ p_{e_{IMU}} - \delta p_e \\ p_{d_{IMU}} - \delta p_d \\ u_{IMU} - \delta u \\ v_{IMU} - \delta v \\ w_{IMU} - \delta w \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \boldsymbol{0}_{6x1} \\ dcm2euler\left(\begin{bmatrix} 1 & -\delta\psi & \delta\theta \\ \delta\psi & 1 & -\delta\phi \\ -\delta\theta & \delta\phi & 1 \end{bmatrix} \times \boldsymbol{R}_{NB}(\boldsymbol{\psi}_{IMU}) \right) \end{bmatrix} \tag{6.15b}$$

Note that $dcm2eueler$ mus be a quadrant safe function that backs the roll, pitch, and yaw angles out of the $\boldsymbol{R}_{NB}$ direction cosine matrix (DCM). Also note that the skew symmetric method of applying the attitude errors to the DCM in equation (6.15) assumes that the angle errors are small. Instead of multiplying the original IMU DCM $\boldsymbol{R}_{NB}$ matrix by a skew symmetric matrix of the angle errors (plus the identity matrix), a full DCM could be created using equation (3.3) in place of the skew symmetric matrix (plus the identity matrix). Later, simulations show that the attitude angle errors can become large. Thus future work could use this full method of accounting for the attitude error states.

Now, this corrected IMU estimate $\hat{\boldsymbol{x}}_{fused}$ is both sent to the autopilot and fed back to the numerical integrators in section 6.1.1 as the a priori data. This is illustrated in the flow chart in Figure 6.1. Because the corrected state vector $\hat{\boldsymbol{x}}_{fused}$ is fed back to the IMU, the error state of EKF 3 $\boldsymbol{x}_{EKF3,k}$ is no longer valid when the next EKF 3 time or measurement update is reached. After the error state that the EKF generated is applied to the IMU solution, the expected error in the IMU solution is by definition zero, as the error has been subtracted out by (6.13). In mathematical terms, see equation (6.16).

$$\hat{\boldsymbol{x}}_{fused}\Big|_{k} = \boldsymbol{x}_{IMU,k-1}\Big|_{k+1} \tag{6.16}$$

Equation (6.16) means that the error corrected, fused IMU-GPS solution at the current measurement update k is the data that is used for the a priori information $\boldsymbol{x}_{IMU,k-1}$ at the next sample of the IMU, which is indexed at the current update as $k+1$. So, after each measurement update of EKF3, the errors in the IMU estimated states must be set to zero, as in equation (6.17).

$$\boldsymbol{x}_{EKF3,k+1} = \begin{bmatrix} \mathbf{0}_{9x1} \\ b_{a_x} \\ b_{a_y} \\ b_{a_z} \\ b_{g_x} \\ b_{g_y} \\ b_{g_z} \end{bmatrix} \tag{6.17}$$

Note that this keeps in memory the currently estimated sensor biases. Thus, in algorithm implementation, it is easier to simplify the measurement update equation to equation (6.18) and hold onto the memory of the estimated biases separately, as shown in Figure 6.1.

$$\boldsymbol{x}_{EKF3,k} = \boldsymbol{K}\boldsymbol{y}_{EKF3} \tag{6.18}$$

This simplification only works because the first 9 elements of the previous measurement update's state vector is zero and the last 6 columns of the Jacobian (6.14) are zero. Reference [X] shows this derivation. The covariance matrix does not go to zero after a measurement update, and must continue to be updated at each measurement update via equation (6.18).

Now, although the error state is updated only at each GPS measurement update, the covariance matrix must be updated at each time update of the EKF. Similar to method 1, a time update will be performed in EKF 3 each time the IMU is sampled. This time update is performed by the state transition matrix formulation in equation (3.21). Here, $\boldsymbol{\Phi}_{\mathrm{EKF3}}$ is the state transition matrix of the expected error between the IMU and GPS solutions from the previous IMU update to the current one, and $\boldsymbol{Q}_{EKF3}$ is the expected process noise from this discretized system.

The state transition matrix $\boldsymbol{\Phi}_{\mathrm{EKF3}}$ is derived from two sources of expected error: the expected error induced on the states by errors in other states, and the error in the IMU solution due to the non-inertial N-E-D frame due to the rotation of the Earth. These dependencies of some state errors on other state errors are expressed by formulating the Jacobian $\boldsymbol{F}$ matrix that is defined in (3.11). This Jacobian matrix is defined for this EKF in equation (6.19).

$$\frac{\partial \boldsymbol{F}_{EKF3}(\boldsymbol{x}_{EKF3},\boldsymbol{u}_{EKF3})}{\partial \boldsymbol{x}_{EKF3}} = \begin{bmatrix} 0_{3x3} & I_{3x3} & 0_{3x3} & 0_{3x3} & 0_{3x3} \\ \frac{\partial \delta \dot{\boldsymbol{V}}}{\partial \delta \boldsymbol{p}} & 0_{3x3} & \frac{\partial \delta \dot{\boldsymbol{V}}}{\partial \delta \boldsymbol{\psi}} & \frac{\partial \delta \dot{\boldsymbol{V}}}{\partial \boldsymbol{b}_{\boldsymbol{a}}} & 0_{3x3} \\ 0_{3x3} & 0_{3x3} & \frac{\partial \delta \dot{\boldsymbol{\psi}}}{\partial \delta \boldsymbol{\psi}} & 0_{3x3} & \frac{\partial \delta \dot{\boldsymbol{\psi}}}{\partial \boldsymbol{b}_{\boldsymbol{g}}} \\ 0_{3x3} & 0_{3x3} & 0_{3x3} & \frac{\partial \dot{\boldsymbol{b}}_{\boldsymbol{a}}}{\partial \boldsymbol{b}_{\boldsymbol{a}}} & 0_{3x3} \\ 0_{3x3} & 0_{3x3} & 0_{3x3} & 0_{3x3} & \frac{\partial \dot{\boldsymbol{b}}_{\boldsymbol{g}}}{\partial \boldsymbol{b}_{\boldsymbol{g}}} \end{bmatrix} \tag{6.19}$$

This matrix is used to derive the state transition matrix by solving the differential equation (3.18), which is depicted by equation (6.20).

$$\boldsymbol{\Phi}_{\mathrm{EKF3}} = e^{T_s \boldsymbol{F}_{EF3}} \tag{6.20}$$

T_s in equation (6.20) is the time between samples of the IMU.

Now, the derivatives in equation (6.19) indicate the expected change in the *rates* of the states of EKF 3 per change in the *values* of the states of EKF 3. In terms of the states of EKF3, the derivatives in (6.19) relate the change in the error rate of a measurement per change in the error of another measurement. For example, $\frac{\partial\delta\dot{\boldsymbol{p}}}{\partial\delta\boldsymbol{V}}$ represents the change in the rate of IMU-GPS position vector error per change in IMU-GPS error of the velocity vector. In the matrix above, for short hand, $\boldsymbol{p}$ is the position vector, $\boldsymbol{V}$ is the velocity vector, $\boldsymbol{\psi}$ is the vector of Euler angles, $\boldsymbol{b_a}$ is the vector of accelerometer biases, and $\boldsymbol{b_g}$ is the vector of gyro biases. Last, recall that these derivatives are functions of the EKF 3 states $\boldsymbol{x}_{EKF3}$ and inputs $\boldsymbol{u}_{EKF3}$. The definitions for these derivatives are below.

$$\frac{\partial\delta\dot{\boldsymbol{V}}}{\partial\delta\boldsymbol{p}} = -skew(\boldsymbol{\omega}_{Earth,nav}) \tag{6.21}$$

$$\frac{\partial\delta\dot{\boldsymbol{V}}}{\partial\delta\boldsymbol{\psi}} = skew\left(\boldsymbol{R}_{NB}(\phi_k,\theta_k,\psi_k)\begin{bmatrix} y_{a_{x,k}} + b_{a_x} \\ y_{a_{y,k}} + b_{a_y} \\ y_{a_{z,k}} + b_{a_z} \end{bmatrix}\right) \tag{6.22}$$

$$\frac{\partial\delta\dot{\boldsymbol{V}}}{\partial\boldsymbol{b_a}} = \boldsymbol{R}_{NB}(\phi_k,\theta_k,\psi_k) \tag{6.23}$$

$$\frac{\partial\delta\dot{\boldsymbol{\psi}}}{\partial\delta\boldsymbol{\psi}} = -skew(\boldsymbol{\omega}_{Earth,inetial} + \boldsymbol{\omega}_{Earth,nav}) \tag{6.24}$$

$$\frac{\partial\delta\dot{\boldsymbol{\psi}}}{\partial\boldsymbol{b_g}} = -\boldsymbol{R}_{NB}(\phi_k,\theta_k,\psi_k) \tag{6.25}$$

$$\frac{\partial\dot{\boldsymbol{b}_g}}{d\boldsymbol{b}_g} = \boldsymbol{I}_{3x3}/\tau_{\text{gyro}} \tag{6.26}$$

$$\boldsymbol{\omega}_{Earth,inetial} = 7.292115\times 10^{-5}\,\frac{rad}{s}\begin{bmatrix} \cos(latitude) \\ 0 \\ -\sin(latitude) \end{bmatrix} \tag{6.27}$$

$$\boldsymbol{\omega}_{Earth,nav} = \begin{bmatrix} v_k/(r_{Earth} - p_{d,IMU,k}) \\ -u_k/(r_{Earth} - p_{d,IMU,k}) \\ -v_k\tan(latitude)\,/(r_{Earth} - p_{d,IMU,k}) \end{bmatrix} \tag{6.28}$$

For the purposes of simulation here, the latitude was assumed to always 40° North. These derivatives are explained in detail in [6] and derived in [7] [20].

Last, the $\boldsymbol{Q}$ matrix is determinable from the expected IMU sensor noise as follows in equations (6.29) through (6.31).

$$\boldsymbol{G} = \begin{bmatrix} 0_{3x3} & 0_{3x3} & 0_{3x3} & 0_{3x3} \\ \boldsymbol{R}_{NB} & 0_{3x3} & 0_{3x3} & 0_{3x3} \\ 0_{3x3} & -\boldsymbol{R}_{NB} & 0_{3x3} & 0_{3x3} \\ 0_{3x3} & 0_{3x3} & I_{3x3} & 0_{3x3} \\ 0_{3x3} & 0_{3x3} & 0_{3x3} & I_{3x3} \end{bmatrix} \tag{6.29}$$

$$\boldsymbol{R_w} = diag\left(\left[\sigma_{a,w_x}\ \sigma_{a,w_y}\ \sigma_{a,w_z}\ \sigma_{g,w_x}\ \sigma_{g,w_y}\ \sigma_{g,w_z}\ \sigma_{a,d_x}\ \sigma_{a,d_y}\ \sigma_{a,d_z}\ \sigma_{g,d_x}\ \sigma_{g,d_y}\ \sigma_{g,d_z}\right]^T\right) \tag{6.30}$$

$$\boldsymbol{Q} = (I_{15x15} + T_s\boldsymbol{F})\left(T_s\boldsymbol{G}\boldsymbol{R_w}\boldsymbol{G^T}\right) \tag{6.31}$$

The standard deviation of accelerometer and gyro sensors, given above as σ_a and σ_g respectively, are typically modelled as a Gauss-Markov process with properties of noise correlated to time and noise uncorrelated to time, $\sigma_{(sensor),d}$ and $\sigma_{(sensor),w}$ respectively. Again, these equations are explained in detail in [6] and derived in [7] [20].

Now, it should be noted that while this method does provide a way to map the effects of some states' errors on other states' errors, in practice, the performance of this filter is strongly impacted by the initial covariance matrix $\boldsymbol{P_0}$. In fact, during simulation of this estimator, it was observed that if the initial covariance of the yaw angle error was set too high, the estimator would favor excess yaw rate bias in place of using the correct IMU supplied yaw angle. This resulted in the yaw estimate being terribly off, and thus causing the inertial velocities u and v to take the place of each other, depending on the current yaw angle. However, with a manually adjusted initial covariance, stable estimation was achieved. A future study could look to augment this filter with an inertial attitude sensor (such as EMI-prone magnetometers) to provide a more stable bias estimation.

Last, once the IMU-GPS fused estimative $\boldsymbol{x}_{fused,k}$ is obtained, the course angle χ is found from the inertial velocity through the relationships (4.13). Further, the ground speed V_g is found by first determining the inertial frame components of the inertial velocity through equation (6.32), and then determining the magnitude of the earth-level north and east components via equation (6.33).

$$[V_N \quad V_E \quad V_D]^T = R_{NB}(\hat{\phi}_k, \hat{\theta}_k, \hat{\theta}_k)[\hat{u}_k \quad \hat{v}_k \quad \hat{w}_k]^T \tag{6.32}$$

$$V_g = \sqrt{V_N^2 + V_E^2} \tag{6.33}$$

6.1.3 Method 2 Summary

In summary, this method performs numerical integrations on unfiltered IMU gyro and accelerometer inputs to obtain a strictly IMU based solution for the states $\boldsymbol{x}_{IMU}$. The IMU solution is then passed through an EKF that determines the minimum variance estimate of the error of the difference between the GPS and IMU solutions for their shared inertial velocity and inertial position states. This process also estimates the error in attitude of the IMU solution as well as dynamically estimates the total gyro and accelerometer biases. The estimated error of the IMU solution is used to correct the IMU solution and the estimated biases are fed back to the IMU estimator. A flow chart that summarizes the equations discussed in this section is in Figure 6.1.

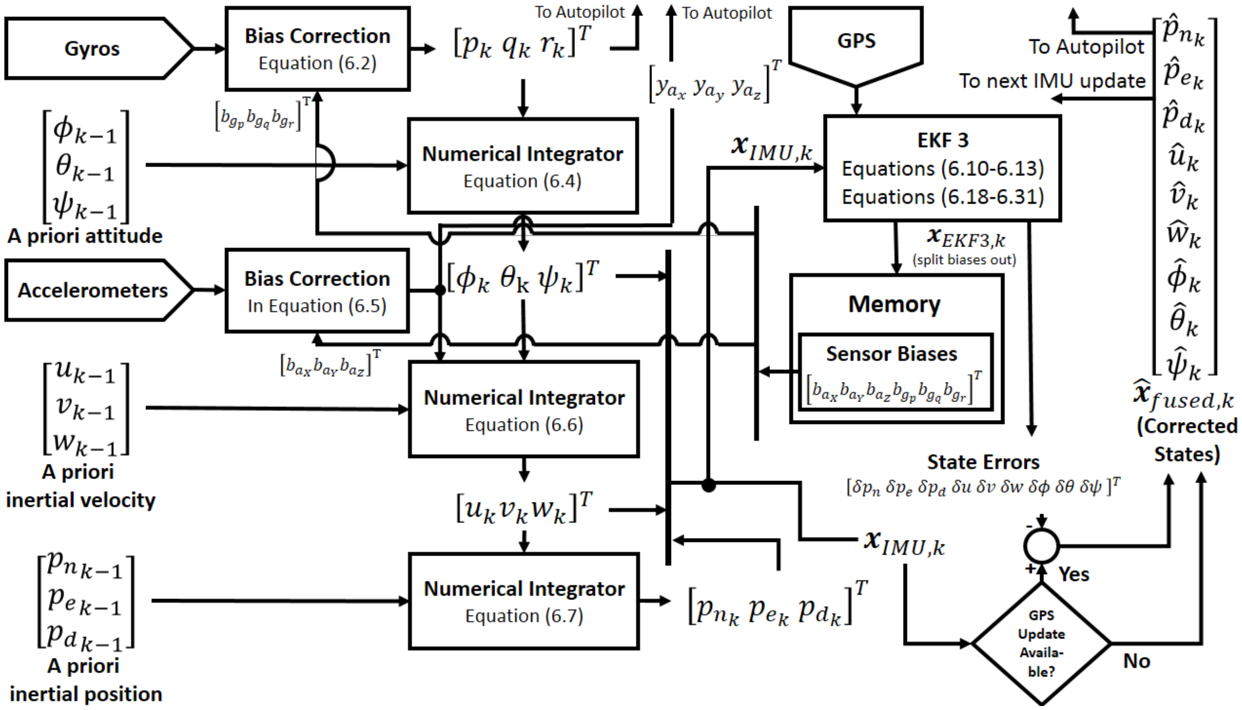

Gyros
Bias Correction
Equation (6.2)
$[p_k\ q_k\ r_k]^T$
To Autopilot
To Autopilot
$[y_{a_x}\ y_{a_y}\ y_{a_z}]^T$
GPS
To Autopilot
To next IMU update
$[b_{g_p} b_{g_q} b_{g_r}]^T$
$[\phi_{k-1}\ \theta_{k-1}\ \psi_{k-1}]^T$
A priori attitude
Numerical Integrator
Equation (6.4)
$x_{IMU,k}$
EKF 3
Equations (6.10-6.13)
Equations (6.18-6.31)
$x_{EKF3,k}$
(split biases out)
Accelerometers
Bias Correction
In Equation (6.5)
$[\phi_k\ \theta_k\ \psi_k]^T$
Memory
Sensor Biases
$[b_{a_x} b_{a_y} b_{a_z} b_{g_p} b_{g_q} b_{g_r}]^T$
$[b_{a_x} b_{a_y} b_{a_z}]^T$
$[u_{k-1}\ v_{k-1}\ w_{k-1}]^T$
A priori
inertial velocity
Numerical Integrator
Equation (6.6)
$[u_k v_k w_k]^T$
State Errors
$[\delta p_n\ \delta p_e\ \delta p_d\ \delta u\ \delta v\ \delta w\ \delta\phi\ \delta\theta\ \delta\psi\ ]^T$
$x_{IMU,k}$
$[p_{n_{k-1}}\ p_{e_{k-1}}\ p_{d_{k-1}}]^T$
A priori
inertial position
Numerical Integrator
Equation (6.7)
$[p_{n_k}\ p_{e_k}\ p_{d_k}]^T$
-
+
Yes
GPS Update Availa-ble?
No
$[\hat{p}_{n_k}\ \hat{p}_{e_k}\ \hat{p}_{d_k}\ \hat{u}_k\ \hat{v}_k\ \hat{w}_k\ \hat{\phi}_k\ \hat{\theta}_k\ \hat{\psi}_k]$
$\hat{x}_{fused,k}$
(Corrected
States)

Figure 6.1: Method 2 Flow Chart

Last, it should be noticed that this estimator does not provide an estimate for airspeed. Only inertial velocity information is obtained, which provides no insight into the airspeed. So, airspeed is separately obtained through a low pass filter on a pitot – static tube airspeed sensor, as depicted in Figure 6.2.

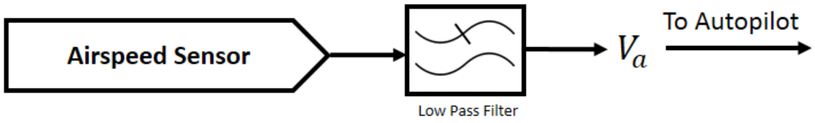


Figure 6.2: Airspeed Estimation for Method 2

6.2 Detailed Simulation Results

This section contains simulation results for the performance of state estimation method 2 under varying levels of wind and the benchmark maneuvers presented in section 4.3. Like the results presented for state estimation method 1 in section 5.2, the autopilot was fed the true states throughout these simulations to keep the aircraft from diverging form the benchmark path during simulations and thus enable direct analysis of the state estimation method during different flight conditions.

6.2.1 IMU Only Solution (No GPS/EKF3 Corrections)

Because state estimation method 2 is built around the notion of propagating the IMU solution for the aircraft's position and attitude over time, and then correcting the IMU errors with GPS data, it is important to first characterize the errors in the states obtained without any corrections to the IMU solutions. With a baseline of the IMU estimator's performance, the effectiveness of EKF3 can be properly analyzed.

6.2.1.1 Benchmark 1 without Bias Drift

To thoroughly analyze the performance of the IMU solution, the performance of the IMU will first be assessed without any gyro rate bias drift. Simulation of such a scenario enables analysis of the behavior of individual state estimates. Figure 6.3 below contains the results of this simulation.

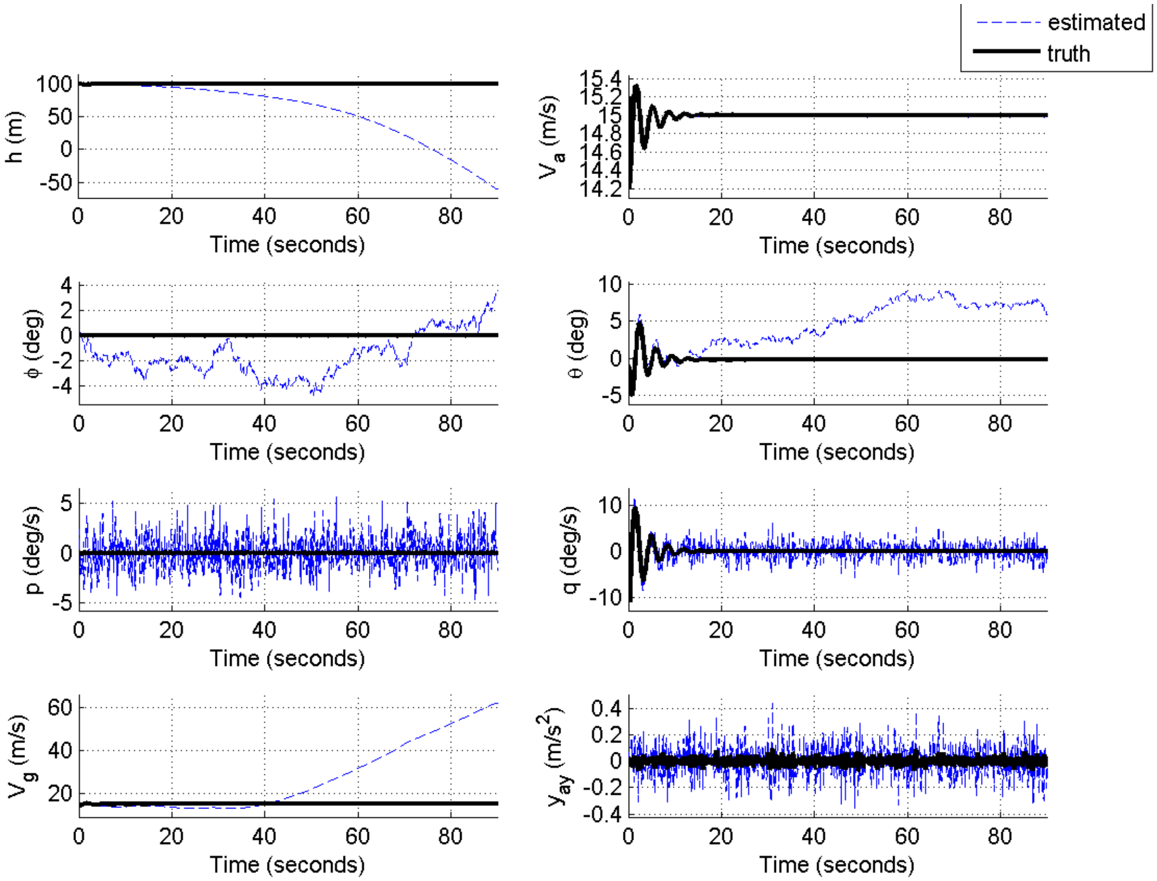


Figure 6.3: State Estimation Results of Method 2 without EFK3 in Benchmark 1 in Zero Wind

Table 6.1: Error Statistics of Method 2 without EKF3 in Benchmark 1 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	42.72204	45.02032	42.72204	45.02032	0.003516	160.7358
$V_a\ (m/s)$	-1.9E-05	0.008112	0.005828	0.005643	0.000002	0.120962
$\phi\ (deg)$	1.593074	1.670424	2.021349	1.114529	0.000277	4.780944
$\theta\ (deg)$	-4.75249	2.927823	4.774749	2.891371	0.000342	9.408488
$p\ (deg/s)$	-0.04432	1.752646	1.38712	1.072108	0.006639	5.641793
$q\ (deg/s)$	-0.068	1.776449	1.401344	1.093806	0.001934	6.239603
$V_g\ \ (m/s)$	-11.8473	15.97751	13.04485	15.01559	0.000121	47.42488

These results show that variations of the error in the angular rates, side specific force y_{a_y}, and airspeed are nearly identical to the variations in their errors in all of the simulations in method 1. The standard deviations of the side specific force in Table 6.1 above and in Table 5.2 for method 1 in benchmark 1 and zero wind are both 0.096 m/s^2. The standard deviations of the airspeed force in Table 6.1 above and in Table 5.2 for method 1 in benchmark 1 and zero wind are both 0.0072 m/s. This makes sense, as the same exact set of low pass filter was used for y_{a_y} and V_a in method 2 as in method 1. On the other hand, the standard deviations of the roll and pitch rates in Table 6.1 for method 2 above are slightly higher than in Table 5.2 for method 1. This is because although the same sensors are used here, the angular rates are no longer passed through a low pass filter. Nonetheless, their values are close, as is indicative of the same sensor model and the high cutoff frequency that is used for the low pass filter for method 1, as is discussed in section 5.2.2.2.

Now, the errors in the remaining states (h, ϕ, θ, V_g) do not closely match the errors in any of the method 1 simulations. The estimates for the bank and pitch angles exhibit a random walk, and the error of the estimates for the altitude and ground speed steadily grow. This behavior is typical of IMU-only solutions. Method 2 directly integrates two sources of Gaussian distributed sensor noise to obtain these four states – the gyros and accelerometers. The angular rate sensor inputs are integrated once in equation (6.4) to obtain the aircraft attitude, and the accelerometer inputs are integrated twice, once in equation (6.6) and again in equation (6.7) to obtain the inertial position. The single integration on the angular rates produces the random walk of the bank and pitch estimates seen in Figure 6.3. This random walk of angle error is often termed angle random walk, and is often specified as a noise bound on gyro manufacturers' data sheets [21]. While the random walk on attitude is direct result of integrating the rate gyros, the transport theorem based equation (6.6) shows that the estimates for the body-frame inertial velocity components u, v, w are integrations of a combination of the accelerometers and the already integrated attitude solution. The attitude is used by equation (5.5) to determine the body-frame components of the gravity vector.

Figure 6.4 below shows these inertial velocity components that are also estimated as part of method 2. The figure also shows the estimate for the yaw angle ψ, which like the bank and pitch angles also exhibits a random walk, but is not shown in the above results.

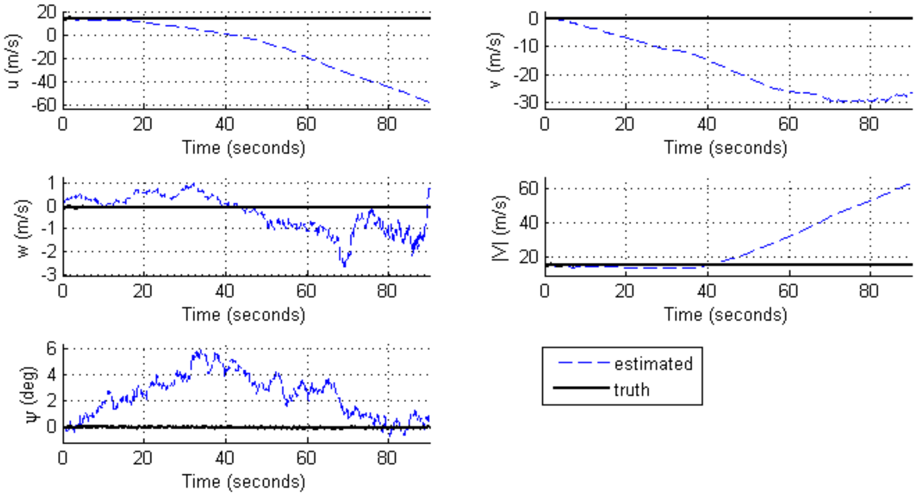


Figure 6.4: Extra State Estimation Results of Method 2 without EFK3 in Benchmark 1 in Zero Wind

This shows that without EKF3, the error of the estimates for the body-frame inertial velocity components u, v, w become very large. Because the desired states h and V_g are direct computations from these values, per equation (6.7) and equation (XX) respectively, it makes sense how much error is seen in these states without EKF3, as depicted in Figure 6.3. However, it isn't immediately clear from Figure 6.4 how much of the error in the inertial velocity components is due to the aggregate effect of bad attitude estimates on the accelerations of equation (6.5), and how much is due to random walk from integration of accelerometer measurements that also have a Gaussian noise distribution. To analyze that, below is the results of a simulation where the true attitude angles were used in method 2 instead of the attitude that is obtained from integrating the rate gyros.

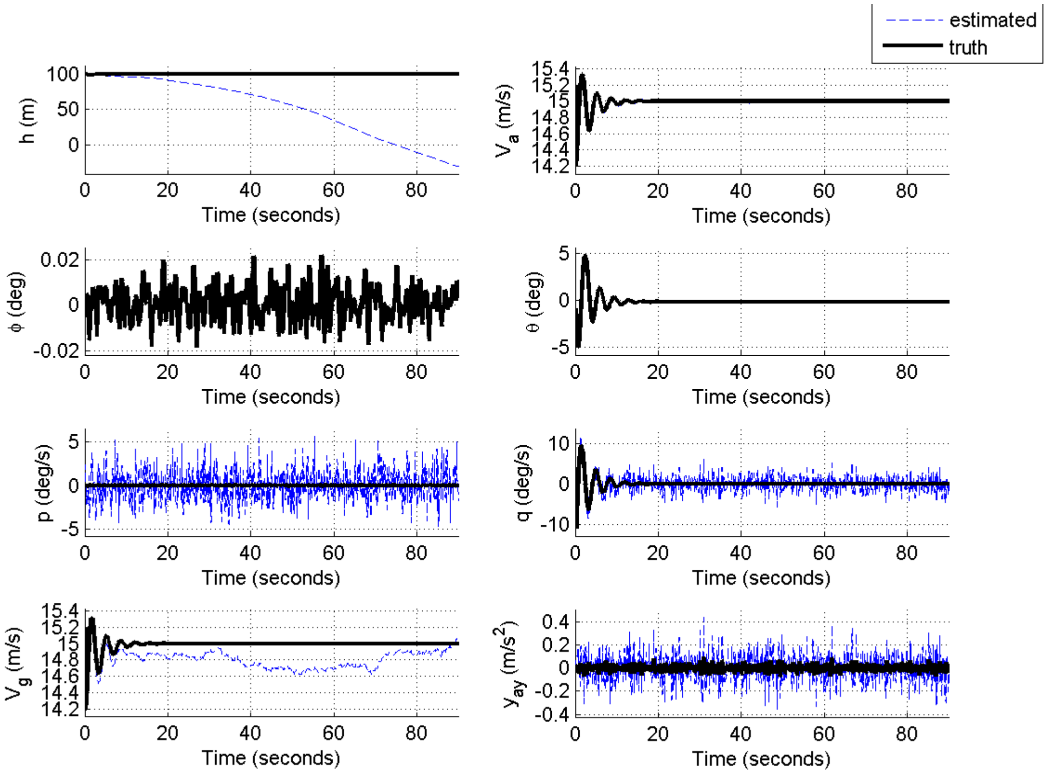


Figure 6.5: State Estimation Results of Method 2 Benchmark 1 without EFK3 and with True Attitude Knowledge

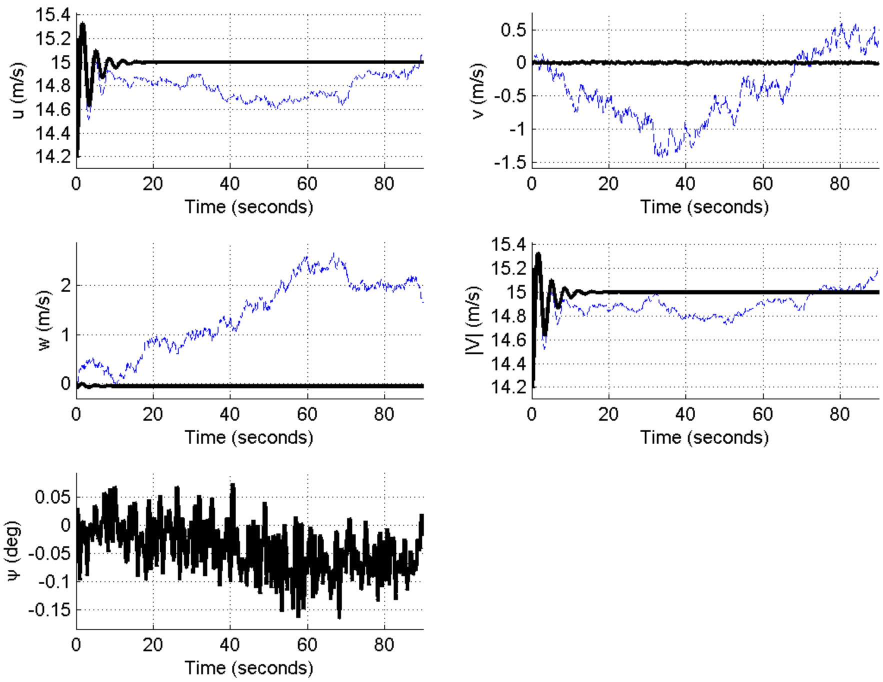


Figure 6.6: Extra State Estimation Results of Method 2 Benchmark 1 without EFK3 and with True Attitude Knowledge

These results show that with zero attitude estimate error, the error in all of the inertial velocity components become small random walks instead of quickly diverging from the true values. Consequently, the estimate for the ground speed V_g also now has a small random walk. This makes sense because V_g is a direct function of these components and the now zero error attitude solution. However, the estimate for the altitude h still quickly diverges from the true altitude. This makes sense because the inertial position is still an integration of this random walk.

In summary, without GPS corrections, random walk of the attitude estimates cause a quickly diverging estimate for the ground speed, but do not have much bearing on the behavior of the altitude estimate. The double integration of the accelerometers causes a random walk in the body-frame inertial velocity components, which in turn causes a random walk in the ground speed V_g estimate, and through a second integration, a quickly diverging value in the altitude estimate. This shows that the ability of GPS corrections through EKF3 to correct the estimates for altitude h and ground speed V_g is paramount for successful estimation of those states. This also shows that it must also be the responsibility of EKF3 to keep the random walk of the attitude solution bounded over time, while the estimates for p, q, y_{a_y}, and $V_a$are not affected by the integration error of the strictly IMU based solution.

6.2.1.2 Benchmark 1

Now that the performance of the raw IMU solution has been analyzed, the added effects of introducing a rate gyro bias drift to the IMU solution can be characterized. The gyro rate bias drift that is introduced here is identical to the bias drift that is used in the simulations for method 1, which is $\frac{1}{30}\frac{deg}{s^2}$ on each gyro axis. Figure 6.7 below contains the results of such a simulation.

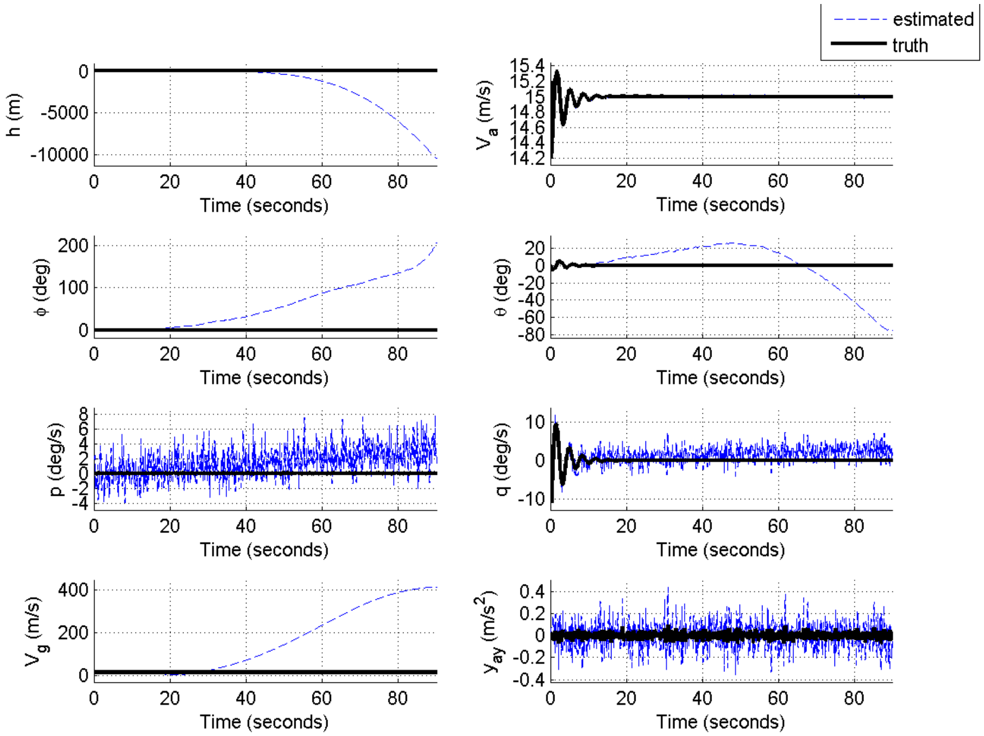


Figure 6.7: State Estimation Results for Method 2 without EKF3 under Benchmark 1 in Zero Wind

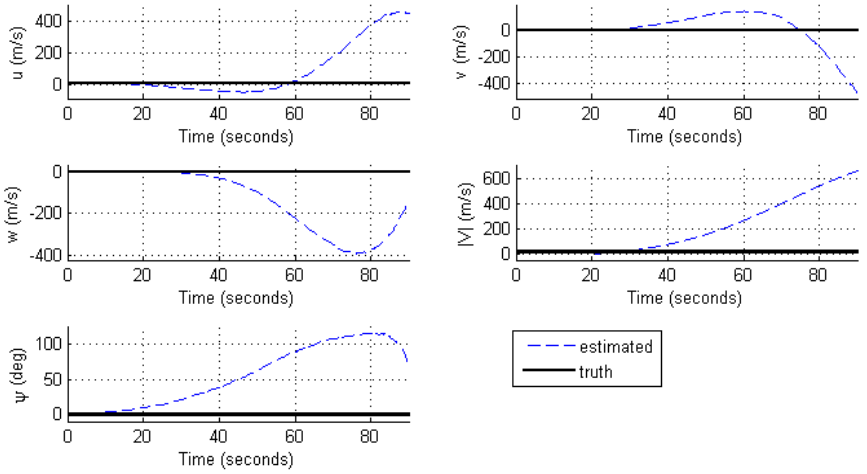


Figure 6.8: Extra State Estimation Results for Method 2 without EKF3 under Benchmark 1 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	1763.6	2726.97	1763.6	2726.97	0.003516	10580.86
$V_a\ (m/s)$	0.000119	0.008135	0.005861	0.005642	0	0.119165
$\phi\ (deg)$	-58.7161	54.72433	58.90604	54.51982	0.000141	206.3099
$\theta\ (deg)$	-0.19626	25.9485	19.23697	17.4144	0.003929	75.1807
$p\ (deg/s)$	-1.5439	2.013943	2.045339	1.50199	0.001062	7.887263
$q\ (deg/s)$	-1.56757	1.948588	2.060914	1.416582	0.007491	7.347993
$V_g\ (m/s)$	-142.625	149.3058	145.5383	146.4675	0.000121	399.9502
$y_{a_y}\ (m/s^2)$	-0.00376	0.09608	0.07568	0.059307	0.000003	0.34756

These results illustrate that when gyro rate bias drifts are added to the strictly IMU based solution, the estimates for the attitude angles become unboundedly large. This is a direct result of the gyro rate bias drift, as equation (6.3), which is based on the angular rate inputs, is the sole source of attitude information. Now, it should be recalled from the previous section that with accurate attitude estimates, the estimate for the ground speed V_g only exhibits a small random walk due accelerometer noise, and although the estimate for the altitude h quickly becomes very large, it only reaches an error of about 150 meters over the course of the 90 second simulation. In these results for the IMU solution with a non-zero gyro bias drift, the magnitude of the error in altitude is much larger, and the ground speed estimate no longer exhibits a small random walk. Figure 6.7 also shows that all of the angular velocity components exhibit very large errors. Thus, these increases in error are strictly due to the substantially increased error in the attitude estimates.

This simulation shows that if the gyro bias drifts are not accounted for, in addition to incurring substantial attitude estimate error, the IMU solutions for the ground speed V_g and altitude h will also be significantly adversely affected.

6.2.2 Zero Winds

Now that the performance of IMU only portion of method 2 has been assessed, the IMU estimate corrections that are supplied by EKF3 with GPS inputs can be added.

6.2.2.1 Benchmark 1 without Gyro Bias Drift

The first simulation that is performed with the entirety of method 2, including EKF3, is the straight and level flight scenario with zero gyro bias drift.

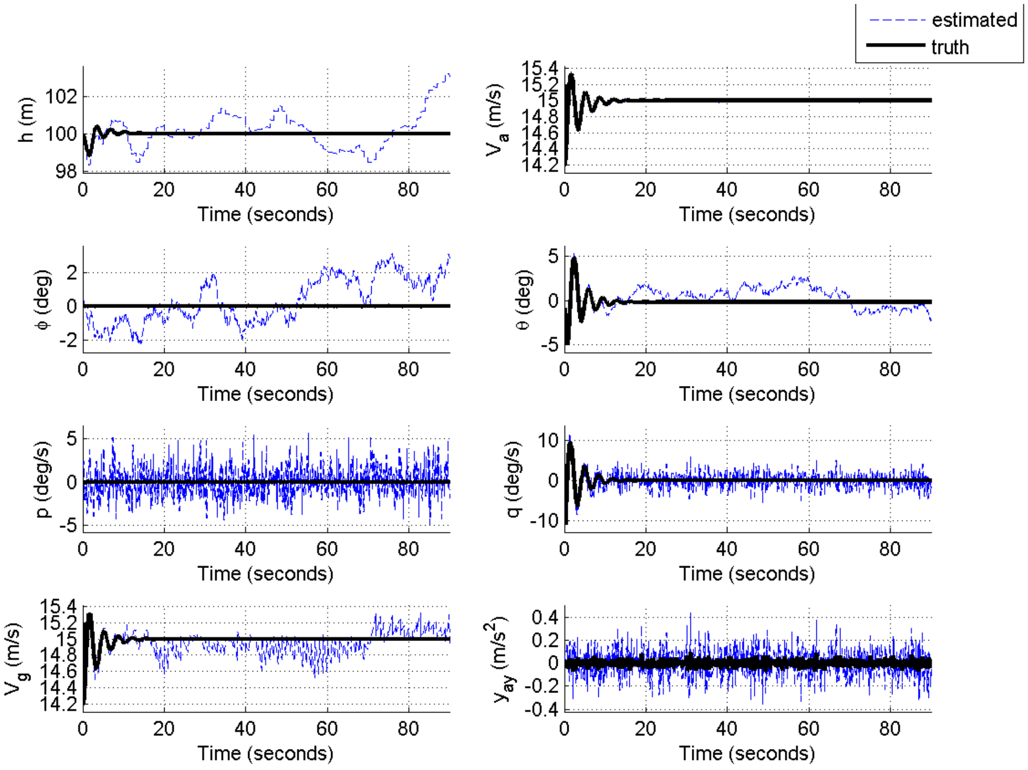


Figure 6.9: Method 2 State Estimation for Benchmark 1 in Zero Wind and Zero Gyro Bias Drift

Table 6.2: State Error Statistics for Benchmark 1 in Zero Wind and Zero Gyro Bias Drift

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	-0.19165	0.948101	0.73011	0.634436	0.000375	3.211446
$V_a\ (m/s)$	-9.7E-05	0.00818	0.005913	0.005653	0.000001	0.121002
$\phi\ (deg)$	-0.32496	1.338593	1.171662	0.724223	0.000019	3.147065
$\theta\ (deg)$	-0.58559	1.065025	1.043912	0.62238	0.000005	2.947473
$p\ (deg/s)$	-0.06213	1.752315	1.386524	1.073222	0.000395	5.683305
$q\ (deg/s)$	-0.02871	1.777905	1.401115	1.094728	0.002865	6.225272
$V_g\ \ (m/s)$	0.042695	0.128317	0.105566	0.084515	0.000003	0.481267

These results can be directly compared to the results obtained in section 6.2.1.1, which was performed under the same conditions but without EKF3. This comparison yields a good measure of the performance of EKF3. The first thing that should be noticed about the results in Figure 6.9 is that estimates for altitude h and ground speed V_g no longer diverge away from the true values like they do in Figure 6.3. Errors in the estimates for the components of the ground speed are still present, as indicated by Figure 6.10, but EKF3 corrects those errors to the GPS provided measurements, which don't accumulate as much error, as is indicated in Figure 6.9.

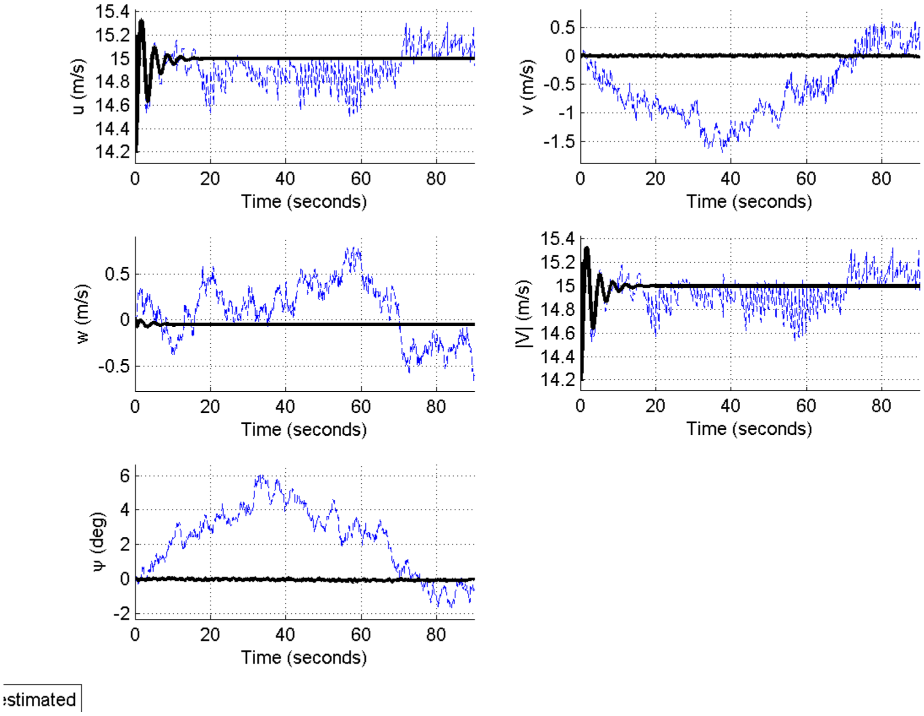


Figure 6.10: Extra Estimated States from Method 2 for Benchmark 1 with Zero Wind and Zero Gyro Rate Bias

While Figure 6.10 shows that the estimates for the body frame inertial velocity component u, v, w still have error, the estimate for the component u is significantly less than without EKF3 as in Figure 6.4. This makes sense because assuming the attitude error is small, EKF3 is able to directly map the GPS inertial frame velocity vector measurement to the body frame velocity vector through equation (6.9) and thus get accurate measurements for u, v, and w. However, as will be seen more later in this document, large attitude estimate errors can drastically degrade the estimates for u, v, and w. The cause of this degradation of the velocity estimates, which is even seen here with small attitude error, is the fact that attitude knowledge is required to map the GPS inertial frame velocity vector to the body frame, again in equation (6.9). When the attitude estimates are not accurate, equation (6.9) provides an inaccurate velocity error measurement $\boldsymbol{\delta V}$, even if the current IMU provided velocity estimate is in fact close to truth.

In this simulation, the pitch and bank angles become up to about 2 degrees off, and the yaw angle becomes up to about 6 degrees off. While these errors in attitude are small, they do still affect the estimates for u, v, and w. For example, the 6 degree error can be visually correlated to the error in v, which is nominally zero in zero wind. Both errors peak at about 40 seconds into the simulation, and go to zero again around 80 seconds into the simulation. It makes sense that the error in ψ directly shows in the estimate for v, as in straight and level flight can, the estimated horizontal velocity v that results from mapping the GPS inertial velocity vector into body coordinates is proportional to $\sin(\psi)$. For small angles, $\sin(\psi)$ is often approximated by ψ, meaning that v is directly proportional to ψ. Oppositely, the inertial velocity component u in straight and level flight is proportional to $\cos(\psi)$, which for small angles is often approximated as 1 and means that u does not change with changes in ψ. Thus, for any small error in ψ in straight and level flight, v will be much more affected than u. Similarly, in level flight where the pitch angle is nominally zero, the vertical velocity w is proportional to $\sin(\theta)$. For small angles of θ, this is approximated as θ, meaning that w is directly proportional to θ. Again, oppositely, u being proportional to $\cos\theta$ in level flight means that u is essentially independent of θ. In conclusion, errors in the attitude in straight and level flight will affect the mapping of the GPS inertial velocity vector into the body frame by larger errors in v and w than u. This is why in Figure 6.10 the estimate for u is seen to be corrected at every GPS measurement interval of 1 second to essentially the true value, while the estimates v and w are corrected to a value that has a small error. The measurement model of (6.9) induces errors in v and w when attitude errors are present, but not in u, and thus is able to correct any IMU error in u.

Now, it can also be noted from Figure 6.9 that the estimates for the pitch and bank angles still visually appear to randomly walk. While this appears to be hardly different from the attitude estimates without EKF3 in Figure 6.3, it should be noticed that the standard deviation and mean of the attitude errors are in fact less with EKF3. This brings up an important recurring notion about attitude error observability that will be highlighted in each of the following simulations. This notion is as follows.

The measurement model of method 2 only relates the differences between the GPS and IMU position and velocity to the position error states and velocity error states, respectively. The measurement model does not relate the measured IMU-GPS differences to the attitude error states, accelerometer bias states, or gyro bias states. The only way these three sets of states are modified in EKF3 is via the use of the covariance matrix P_{EKF3} in the Kalman gain K_{EKF3}. Thus, the GPS-IMU difference measurements only affect the attitude error, accelerometer bias, and gyro bias states through the system model as is described in equation (6.19) through equation (6.31) and is propagated into the covariance matrix by equation (3.21). In the system model, the only equation relating the attitude error to the directly measured position and velocity errors is equation (6.22). The attitude error is also related to the gyro bias via equation (6.25), but that will be discussed in the next section. Equation (6.22) is derived from an analysis of the difference between the IMU and GPS solutions for the time rate of change of velocity, and is derived in detail in [7] and highlighted in [6]. The most important aspect of this sole relationship between attitude error and measured velocity and position error is that it does not make assumptions about the typical wind frame aligned motion of an aircraft. The only assumptions are that the system is an object with body frame attached accelerometers. That means that, for example in the case of zero wind and level flight, the aircraft could have a large body frame axis horizontal velocity v and zero longitudinal velocity u, which indicates a 90 degree sideslip, and that would have no bearing on this relationship. Instead of relating attitude error to typical motions of aircraft, this relationship (6.22) compares measured accelerations along body frame axes to the time rate of change of velocity error.

This concept is most easily illustrated in a 2D scenario, as depicted in the aircraft's horizontal plane in the figure below.

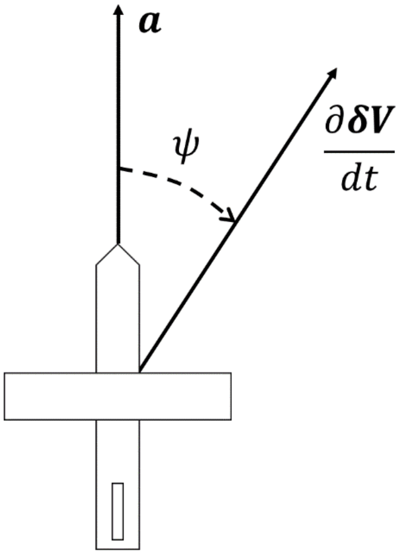


Figure 6.11: Dependence of Difference between Accelerometer Measurement and Velocity Errror Growth on Yaw Growth

This illustrates that difference between the vector of increasing velocity error can be related to the measured acceleration through the error in the yaw angle. In this case, if the velocity error is growing in the X-Y plane, it can be concluded that there is an error in the estimated yaw angle if the error in the velocity is growing in a different direction than the acceleration vector. In other words, if the measurement model of (6.9) provides a non-zero velocity error, it can be assumed that there is some error in the yaw angle if the aircraft has only accelerated in another direction.

Expanding this notion to all three axes for the straight and level scenario can be performed mathematically by solving equation (6.22) around the straight and level flight assumptions of $\theta = 0$ and $\phi = 0$. Making these substitutions into (6.22) results in the following system of equations that relate time rate of change of the components of the velocity error state $\boldsymbol{\delta V}$ to the measured accelerations, current attitude, and the attitude error state $\boldsymbol{\delta\psi}$.

$$\frac{\partial \delta \dot{u}}{\partial \delta \theta} = -\frac{\partial \delta \dot{v}}{\partial \delta \phi} = -\left(y_{a_z} + b_{a_z}\right) \tag{6.34}$$

$$\frac{\partial \delta \dot{u}}{\partial \delta \psi} = -\frac{\partial \delta \dot{w}}{\partial \delta \phi} = \left(y_{a_x} + b_{a_x}\right)\sin\psi - \left(y_{a_y} + b_{a_y}\right)\cos\psi \tag{6.35}$$

$$\frac{\partial \delta \dot{v}}{\partial \delta \psi} = -\frac{\partial \delta \dot{w}}{\partial \delta \theta} = \left(y_{a_y} + b_{a_y}\right) sin\,\psi - \left(y_{a_x} + b_{a_x}\right) cos\,\psi \tag{6.36}$$

An important notion can be drawn from these simplified relationships. When these equations are inverted, which is done numerically by the Kalman filter, it can be seen that every component of the attitude error state $\boldsymbol{\delta\psi}$ is not always affected by every axis' accelerometer input. The dependence of the attitude error states can be determined by inspection of the above equations and can be summarized by the below table.

Table 6.3: Dependence of Attitude Error Observability on Accelerations for Benchmark 1

State that is Observed	Observability *Enabled* by Acceleration along Axes
$\delta\phi$	X,Y,Z
$\delta\theta$	X,Y,Z
$\delta\psi$	X,Y

This table describes the skew symmetric relationship in equation (6.22) of the measured acceleration to the expected change in velocity error due to attitude error. This notion is a qualitative description of the dependence of attitude error observability on the maneuvers of the aircraft. However, it is extremely important to note that accelerations along these axes do not guarantee observability of the corresponding attitude angle; these accelerations simply *enable* these attitude errors to be observed. In other words, it is impossible for an attitude angle error to be observable if there is no acceleration in any of the corresponding axes listed in the above table.

True observability of the above attitude angles involves a much more in-depth analysis, and such an analysis is performed in great mathematical detail in [22] from both a time-invariant approach and a time-varying approach that uses a concept of instantaneous observability. The research presented in [22] concludes that introducing time varying accelerations improves the observability of the attitude angles, but the yaw angle ψ remains less observable than the others. A similar conclusion can be drawn here by realizing that under normal circumstances of aircraft flight, it is desired to minimize the X and Y axes accelerations, while the Z-axis specific force is always present due to gravity. Mapping this knowledge of accelerations to the table above produces the conclusion the roll and pitch attitude angle errors will always be much more observable than the yaw angle error. In order to measure yaw error, accelerations in the X or Y axes are need, and are not typically present. On the other hand, accelerations in the Z axis (gravity) enable measuring of the roll and pitch errors.

Last, it is important to note that the accelerations used here are the accelerometer inputs after applying the estimated accelerometer biases. Large errors in accelerometer bias estimates do not occur in these simulations due to the lack of accelerometer bias drift. Analyzing the effects of accelerometer bias errors could be a topic of future study.

In conclusion, the attitude angles still randomly walk because EKF3 does not correct for them directly, and instead relies on differences between velocity error growth and measured accelerations to correlate measured velocity error to attitude error through the covariance matrix. Further, the reason the pitch and bank angle estimates have less error with EKF3, whereas the error in estimated yaw is about the same with and without EKF3, is that yaw and bank angles are somewhat observable due to a constant acceleration due to gravity and yaw is not observable unless accelerations in the X-Y plane are provided.

6.2.2.2 Benchmark 1

Now that the performance of EKF3 has been analyzed without the presence of gyro bias drift, the same gyro bias drift of $\frac{1}{30}\frac{deg}{s^2}$ on each axis can be added to the simulation. The below results are for a simulation of straight and level flight in zero wind, but now including these gyro bias drifts.

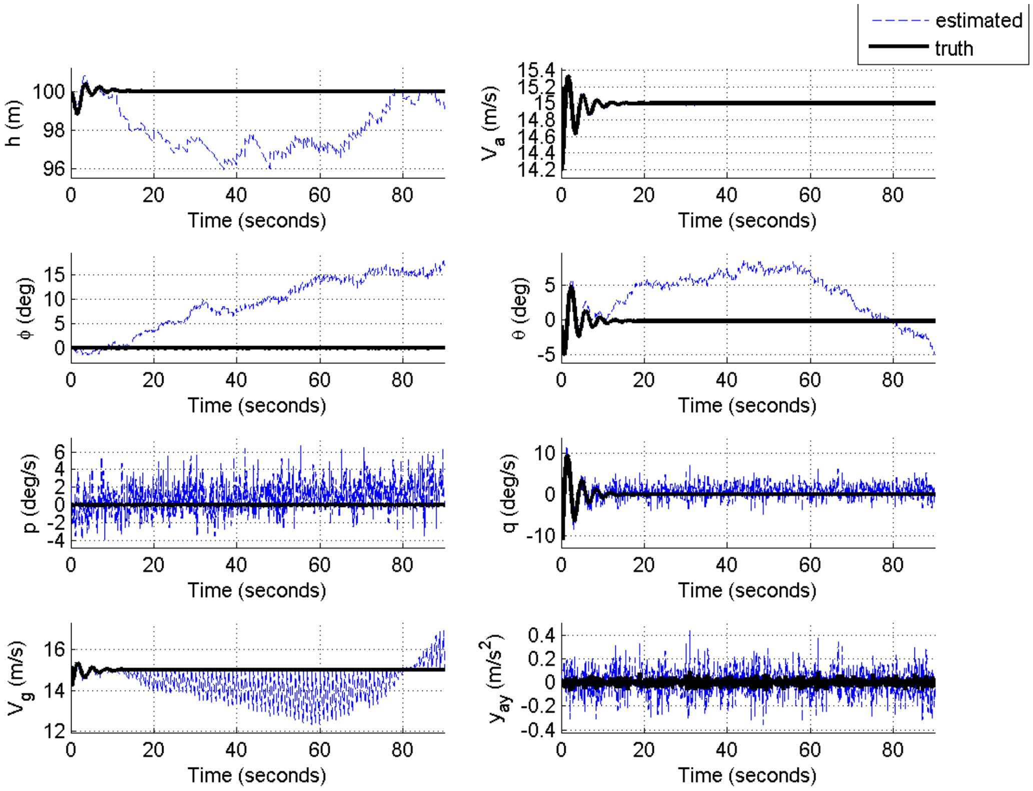


Figure 6.12: Method 2 State Estimates for Benchmark 1 in Zero Wind

Table 6.4: State Estimation Error Statistics for Method 2 Benchmark 1 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	-1.31604	0.903431	1.374919	0.811	0.000674	2.885288
$V_a\ (m/s)$	-3.6E-05	0.007888	0.005789	0.005357	0	0.119036
$\phi\ (deg)$	-9.07726	5.689311	9.229268	5.439224	0.001429	18.08639
$\theta\ (deg)$	-3.75936	3.214824	4.210033	2.596742	0.002526	8.772794
$p\ (deg/s)$	-0.97087	1.821175	1.630213	1.265496	0.003296	6.753102
$q\ (deg/s)$	-0.85567	1.794001	1.602877	1.17524	0.000111	7.016067
$V_g\ \ (m/s)$	0.576728	0.713256	0.690267	0.60404	0.000011	2.70545

At first glance, it is immediately apparent that EKF3 does not eliminate all of the effects of a gyro bias drift. While the estimates for pitch θ and roll ϕ do not grow to the extent they do without EKF3 as shown in section 6.2.1.2, they still contain significantly more error than without bias drift as shown in section 6.2.2.1. Below are the results for the remaining estimates of the IMU states.

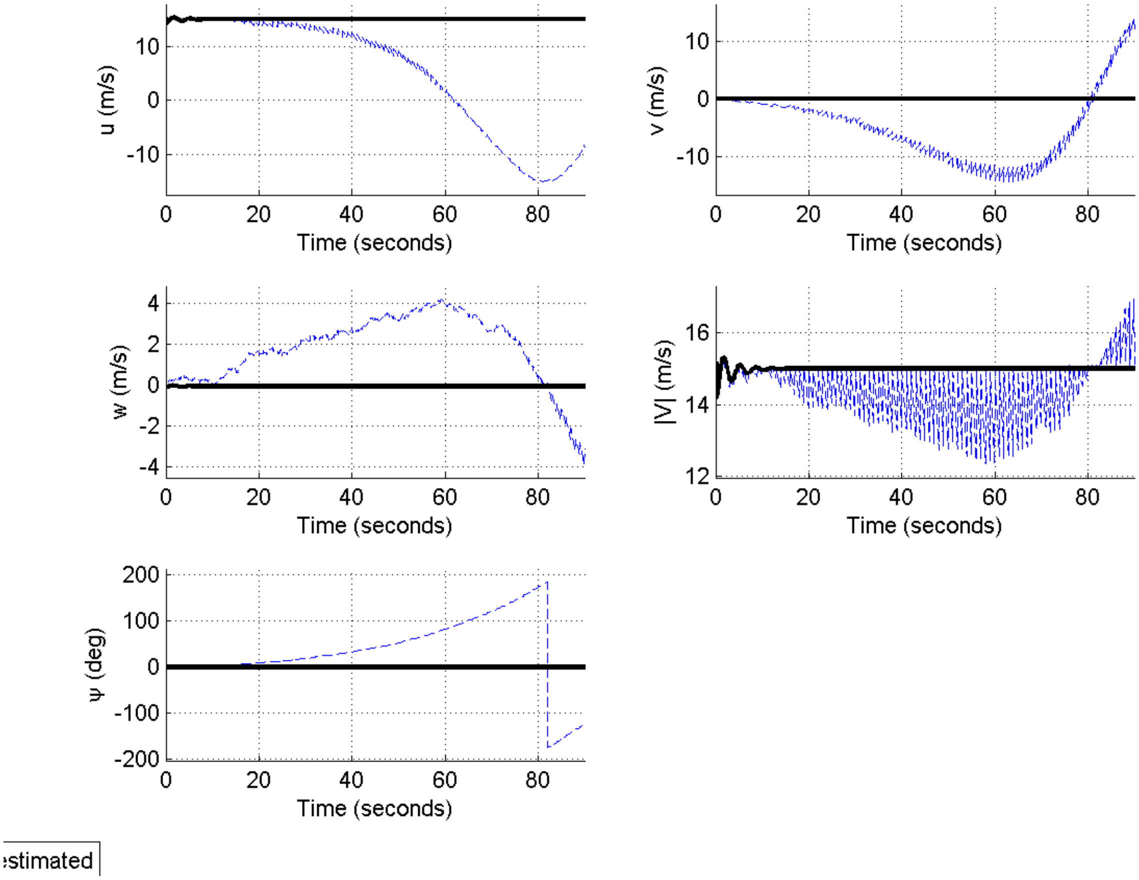


Figure 6.13: Extra State Estimates for Benchmark 1 in Zero Wind

First, it should be noted that just as in the results without gyro rate bias in Figure 6.9, the estimate for V_g is essentially being 100% corrected at each GPS measurement update. This is because the components of Kalman gain for the velocity errors is always near 1, which is due to the fact that the expected measurement noise on GPS measurements is very low. The estimate for V_g is determined by rotating the body frame velocity vector into the inertial frame, as in equation (6.22), and then by taking the magnitude of the first two components, as in equation (6.33). The net result of this computation and the Kalman gain being near 1 is that at each GPS measurement update, the GPS measurements are mapped into the body frame and then directly back to the inertial frame to compute the ground speed. This essentially error-free mapping and immediate inverse mapping causes the ground speed estimate to be corrected to the ground speed directly computed from the GPS provided north and east velocities.

Next, as discussed in the previous section, in straight and level flight and zero wind there is zero measured acceleration in the horizontal plane and measured acceleration due to gravity in the vertical X-Z and Y-Z planes. Thus, it can be expected that roll and pitch angle errors are corrected much more than yaw angle errors. The results here reflect this. The roll and pitch angles stay close to their true values, while the yaw estimate steadily increases in error with time.

Nonetheless, it should be recalled that EKF3 also estimates the gyro sensor bias and accelerometer sensor bias. The estimate for the gyro sensor bias comes from the relationship (6.25) between the time rate of change of the attitude error estimates $\delta\boldsymbol{\psi}$ and the gyro biases $\boldsymbol{b}_g$. The two quantities are directly related through the body-to-inertial transformation matrix $\boldsymbol{R}_{NB}$. In straight in level flight, the transformation matrix is well characterized by single rotation about the Z axis of an amount equal to the yaw angle. This means that the roll and pitch gyro bias states are linear combinations of the time rate of change of the roll and pitch attitude error states, and the yaw bias state is directly mapped to the time rate of change of the yaw error state. This relationship is summarized by simplifying equation (6.25) with the assumption that in straight and level flight, $\phi = \theta = 0$.

$$\begin{bmatrix} \frac{\partial \delta\dot{\phi}}{\partial b_{g_x}} & \frac{\partial \delta\dot{\phi}}{\partial b_y} & \frac{\partial \delta\dot{\phi}}{\partial b_z} \\ \frac{\partial \delta\dot{\theta}}{\partial b_{g_x}} & \frac{\partial \delta\dot{\theta}}{\partial b_{g_y}} & \frac{\partial \delta\dot{\theta}}{\partial b_{g_z}} \\ \frac{\partial \delta\dot{\psi}}{\partial b_{g_x}} & \frac{\partial \delta\dot{\psi}}{\partial b_{g_y}} & \frac{\partial \delta\dot{\psi}}{\partial b_{g_z}} \end{bmatrix} = -\begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Since it was shown in the previous section that in straight and level flight, roll and pitch attitude errors are observable and yaw errors are not, it is logical to conclude that because of this relationship, in straight and level flight, the roll error and pitch gyro biases are able to be observed, and the yaw bias is not. This notion is supported by the gyro bias states that are estimated in this simulation, which are shown below.

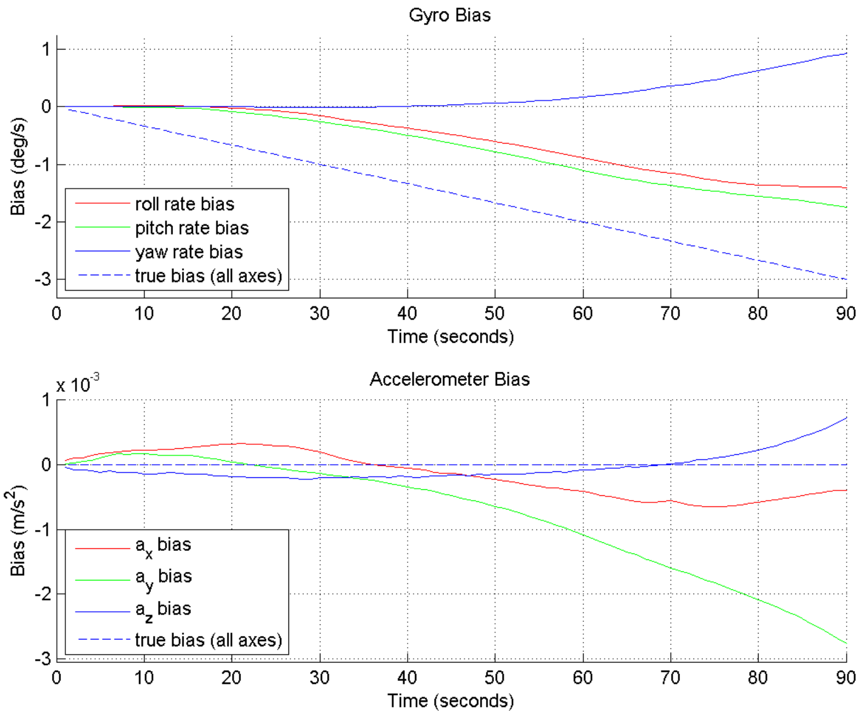


Figure 6.14: Bias Estimates for Method 2 in Benchmark 1 in Zero Wind

These results show that as expected, the estimated roll and pitch gyro bias tracks their true values much closer than the estimate for the yaw bias does. A lag still exists in the bias estimates for the other angles, but that is also to be expected as the system model indicates a zero time rate of change of both the accelerometer and gyro biases. Again, the only relationship affecting the estimate of the gyro bias is (6.25), relating attitude error to the gyro bias.

Last, the estimates for the body-frame components of the inertial velocity can be seen to correlate to the errors in attitude. The sign and relative magnitude of the yaw error can be seen in the varying estimates for u and v, and the variation of the pitch estimate can be seen in the varying estimate for the vertical velocity w. As explained previously, this is due to the relationship (6.9) which requires attitude information to map the inertial frame GPS measurements into the body frame.

The derivation of the propagation of the attitude error, as is performed in [7], omits a term relating the time rate of change of attitude error to the estimated angular velocity of the aircraft.   This term is omitted simply because it is not part of the EKF 3's states. Inclusion of such a term could perhaps improve IMU-GPS attitude error tracking, but would require estimating the angular velocity as part of the Kalman filter.

6.2.2.3 Benchmark 2

Now that the effects of adding gyro bias drift on the state estimates have been analyzed, a simulation of a coordinated turn in zero wind can be performed and analyzed. Below are the method 2 state estimation results for such a simulation.

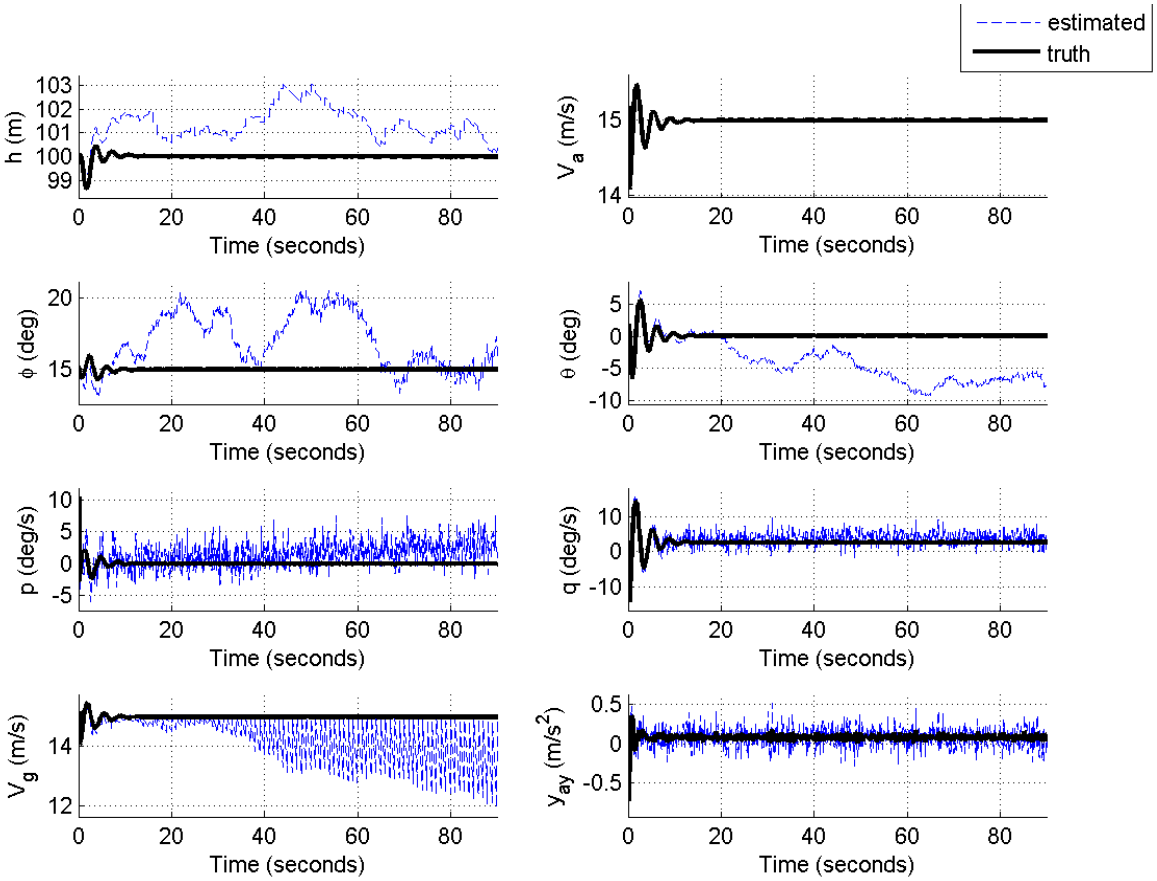


Figure 6.15: State Estimation Results for Benchmark 2 in Zero Wind

Table 6.5: State Estimation Error Statistics for Benchmark 2 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	-1.31604	0.903431	1.374919	0.811	0.000674	2.885288
$V_a\ (m/s)$	-3.6E-05	0.007888	0.005789	0.005357	0	0.119036
$\phi\ (deg)$	-9.07726	5.689311	9.229268	5.439224	0.001429	18.08639
$\theta\ (deg)$	-3.75936	3.214824	4.210033	2.596742	0.002526	8.772794
$p\ (deg/s)$	-0.97087	1.821175	1.630213	1.265496	0.003296	6.753102
$q\ (deg/s)$	-0.85567	1.794001	1.602877	1.17524	0.000111	7.016067
$V_g\ \ (m/s)$	0.576728	0.713256	0.690267	0.60404	0.000011	2.70545

It is immediately apparent from the plot of the estimates for roll rate p in Figure 6.15 that EKF3 is doing a much poorer job of accounting for the roll rate bias drift than in the previous simulation, as the estimated roll rates error steadily increases at a faster rate than in benchmark 1. This is also represented in the plot of the gyro bias estimate below.

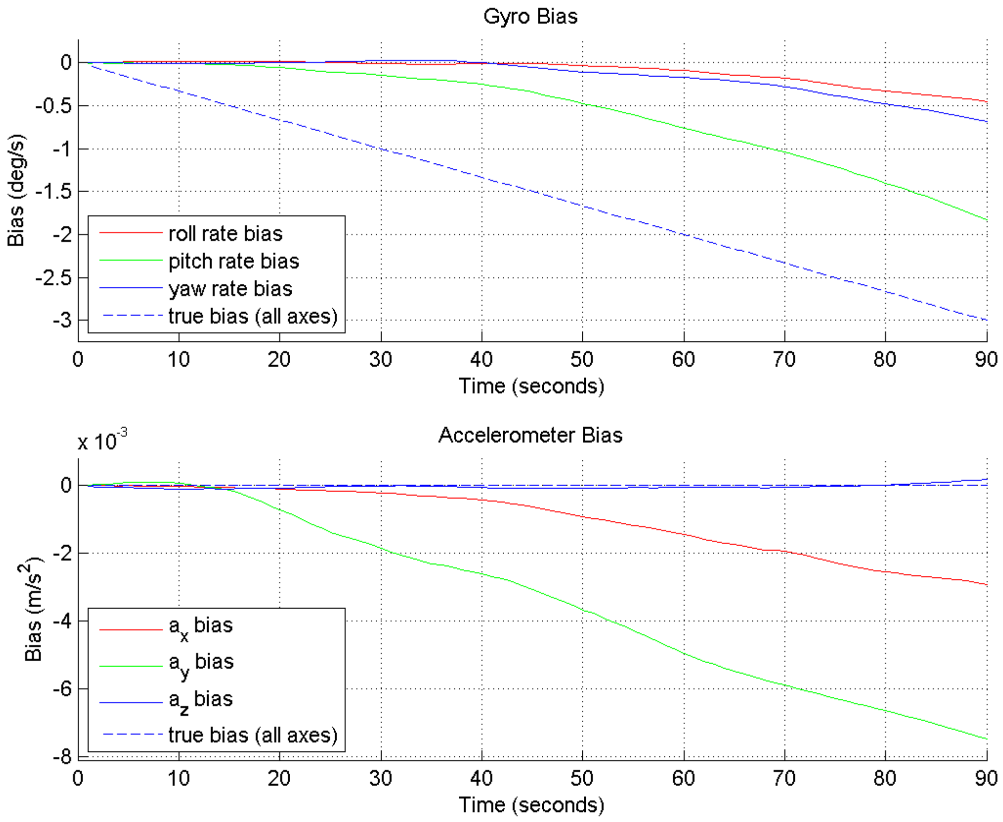


Figure 6.16: Bias Estimation for Benchmark 2 in Zero Wind

This plot shows that in addition to a more significant roll rate bias error, there yaw rate bias error that, unlike in benchmark 1, tracks along with the roll rate error. The effects of these roll and yaw rate bias estimates are also apparent in the estimate for the yaw angle and body frame components of inertial velocity, which are illustrated below.

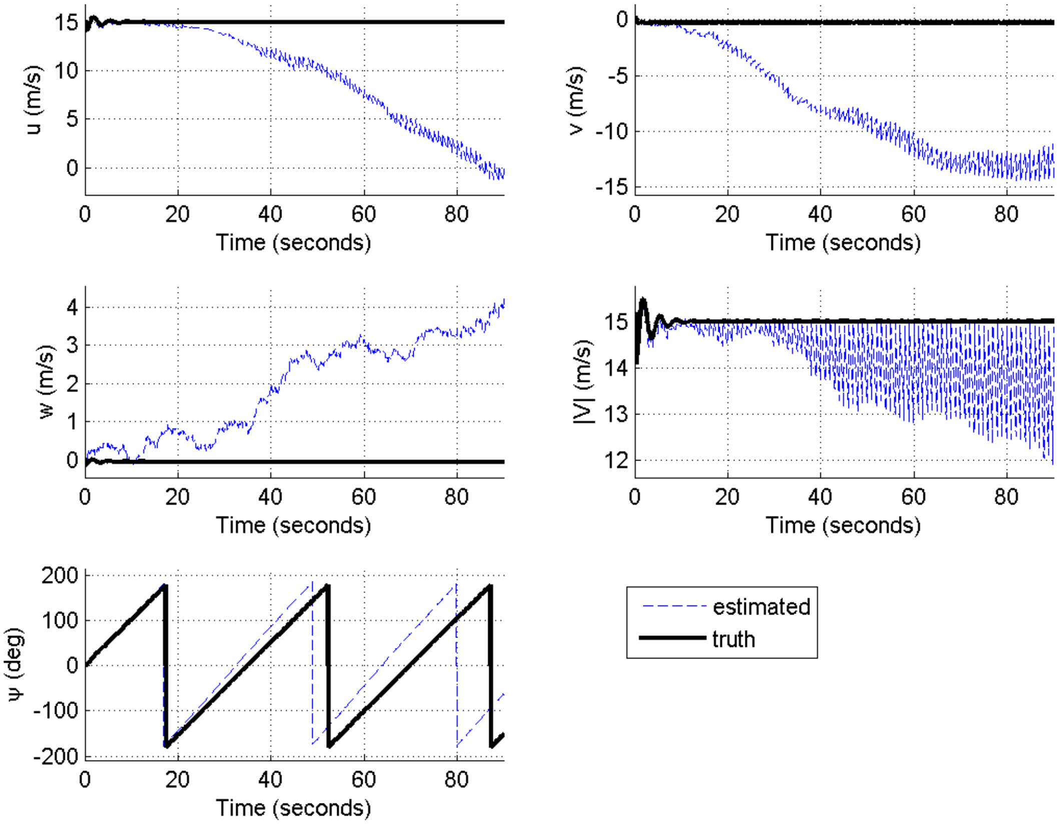


Figure 6.17: Extra State Estimation Results for Benchmark 2 in Zero Wind

This poor estimate for the roll rate gyro bias brings the discussion back to observability. Table 6.3 shows the enabling of observability of attitude angles when acceleration is observed during straight and level flight. Unfortunately, these relationships are only true for straight and level flight, so in order to find relationships for a coordinated turn, equation (6.22) needs to be re-solved with a single assumption of $\theta = 0$. This results in the following relationships.

$$\frac{\partial \delta \dot{u}}{\partial \delta \theta} = -\frac{\partial \delta \dot{v}}{\partial \delta \phi} = -\left(y_{a_y} + b_{a_y}\right)\sin\phi - \left(y_{a_z} + b_{a_z}\right)\cos\phi \qquad (6.37)$$

$$\frac{\partial \delta \dot{u}}{\partial \delta \psi} = -\frac{\partial \delta \dot{w}}{\partial \delta \phi} = \left(y_{a_x} + b_{a_x}\right)\sin\psi + \left(y_{a_y} + b_{a_y}\right)\cos\psi\cos\phi - \left(y_{a_z} + b_{a_z}\right)\cos\psi\sin\phi \qquad (6.38)$$

$$\frac{\partial\delta\dot{v}}{\partial\delta\psi} = -\frac{\partial\delta\dot{w}}{\partial\delta\theta} = -\left(y_{a_x} + b_{a_x}\right)\cos\psi + \left(y_{a_y} + b_{a_y}\right)\cos\phi\sin\psi - \left(y_{a_z} + b_{a_z}\right)\sin\psi\sin\phi \quad (6.39)$$

These relationships show that the yaw angle error $\delta\psi$ is now also affected by accelerations along the Z axis. In fact, all attitude angle errors are affected by accelerations along any of the axes. This is a problem when it comes to estimating the attitude error states because in method 2, there are never more than 2 observable attitude angle modes. The below table contains an output of calculating the rank of several different observability matrices. These matrices were determined by selecting the columns and rows of the system and measurement models for EKF3 that correspond to the states listed in the “Composing States” column.

Table 6.6: Observability of Method 2 States for All Simulations

Observability Matrix	Composing States	Rank	Unobservable Modes
$\mathbf{O}$	All 15 states	13	2
$\boldsymbol{O}_{\boldsymbol{all-position}}$	$\delta u\ \delta v\ \delta w\ \delta\phi\ \delta\theta\ \delta\psi\ b_{a_X} b_{a_Y}\ b_{a_Z}\ b_{g_X} b_{g_Y} b_{g_Z}$	10	2
$\boldsymbol{O}_{\boldsymbol{gps,attitude}}$	$\delta p_n\ \delta p_e\ \delta p_d\ \delta u\ \delta v\ \delta w\ \delta\phi\ \delta\theta\ \delta\psi$	8	1
$\boldsymbol{O}_{\boldsymbol{gps,accel}}$	$\delta p_n\ \delta p_e\ \delta p_d\ \delta u\ \delta v\ \delta w b_{a_X} b_{a_Y} b_{a_Z}$	9	0
$\boldsymbol{O}_{\boldsymbol{gps,gyro}}$	$\delta p_n\ \delta p_e\ \delta p_d\ \delta u\ \delta v\ \delta w b_{g_X} b_{g_Y} b_{g_Z}$	8	1
$\boldsymbol{O}_{\boldsymbol{attitude,gyro}}$	$\delta\phi\ \delta\theta\ \delta\psi\ b_{g_p} b_{g_q} b_{g_r}$	4	2
$\boldsymbol{O}_{\boldsymbol{accel,gyro}}$	$\delta\phi\ \delta\theta\ \delta\psi\ b_{g_p} b_{g_q} b_{g_r}$	5	1
$\boldsymbol{O}_{\boldsymbol{attitude,accel}}$	$\delta\phi\ \delta\theta\ \delta\psi\ b_{a_X} b_{a_Y} b_{a_Z}$	5	1

Interestingly, the rank of theses observability matrices never changes throughout the different benchmarks or even in the Dryden wind conditions that method 2 is simulated under. The research performed in [22] shows that under zero specific force conditions (zero gravity and inertial Earth frame accelerations), the system has 6 unobservable modes. It then shows that when any acceleration is added (which can include gravity), the system only haves 3 unobservable modes. The above table shows that in these simulations, the number of unobservable modes never exceeds 2. In its instantaneous observability analysis, [22] finds that either if the aircraft undergoes a coordinated turn or it experiences time varying accelerations, the number of unobservable modes is decreased from 3 to 2, with the unobservable modes being associated with the angle about the jerk vector. It makes sense that even in straight and level flight, only 2 unobservable modes would be observed because even very small non-zero accelerations will increase the rank of the system model and thus the observability matrix. Both the accelerometer sensor noise and small oscillating command tracking errors of the autopilot can be sources of such small acceleration.

Now, both the $\boldsymbol{O}_{\boldsymbol{gps,gyro}}$ matrix and the $\boldsymbol{O}_{\boldsymbol{gps,attitude}}$ matrix have one unobservable mode, and the $\boldsymbol{O}_{\boldsymbol{gps,accel}}$ matrix has zero unobservable modes. Thus, the two unobservable modes correspond to one attitude measurement and one gyro bias measurement.   In the results for this benchmark, both the bank and pitch angle estimates are shown to be corrected towards zero error, while the yaw angle error grows unboundedly. Thus, it is logical to conclude that the yaw angle error is the unobservable mode and that small jerk vectors exist in the Z-axis of the aircraft. With that knowledge, the system model relationship (6.25) that determines the dependence of gyro bias on attitude error can be examined to determine what the unobservable gyro bias mode is. Below is equation (6.25) simplified around the single assumption of $\theta = 0$.

$$\begin{bmatrix} \frac{\partial \delta\dot{\phi}}{\partial b_{g_x}} & \frac{\partial \delta\dot{\phi}}{\partial b_y} & \frac{\partial \delta\dot{\phi}}{\partial b_z} \\ \frac{\partial \delta\dot{\theta}}{\partial b_{g_x}} & \frac{\partial \delta\dot{\theta}}{\partial b_{g_y}} & \frac{\partial \delta\dot{\theta}}{\partial b_{g_z}} \\ \frac{\partial \delta\dot{\psi}}{\partial b_{g_x}} & \frac{\partial \delta\dot{\psi}}{\partial b_{g_y}} & \frac{\partial \delta\dot{\psi}}{\partial b_{g_z}} \end{bmatrix} = -\begin{bmatrix} \cos\psi & -\cos\phi\sin\psi & \sin\psi\sin\phi \\ \sin\psi & \cos\phi\cos\psi & -\cos\psi\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}$$

If $\delta\psi$ is not observable ,it is not immediately clear from this relationship what the unobservable gyro bias mode is, as all three biases are dependent on the pitch and roll error states. At this point, it should be recalled that all of the gyro bias estimates here, unlike in benchmark 1, track the true gyro bias estimate at least somewhat. This leads to the notion that even though there are always at most 2 observable gyro bias modes, the unobservable mode does not have to be exactly aligned with a particular axis. The results here indicate that the unobservable mode contains part of the yaw rate bias estimate and the roll rate bias estimate, as those track their true values slower than the pitch bias estimate.

Nonetheless, this discovery of partially observable gyro bias states is best explored by providing a large time-varying jerk vector. By providing accelerations that change in magnitude and direction, the attitude angle errors that the unobservable modes affect could be made time-varying. Thus, each attitude angle error would be observable at different points and time, and consequently, all three gyro biases could share time of being observable. In other words, it would spread the unobservable gyro bias mode around between the gyro bias states. To simulate this, a $\frac{15}{90}\ Hz$ sine wave signal was applied to the rudder with an amplitude of 8 degree deflection, and a $\frac{25}{90}$ Hz sine wave signal was applied to the elevator with an amplitude of 8 degree deflection. The results of the bias estimation of this simulation are shown below.

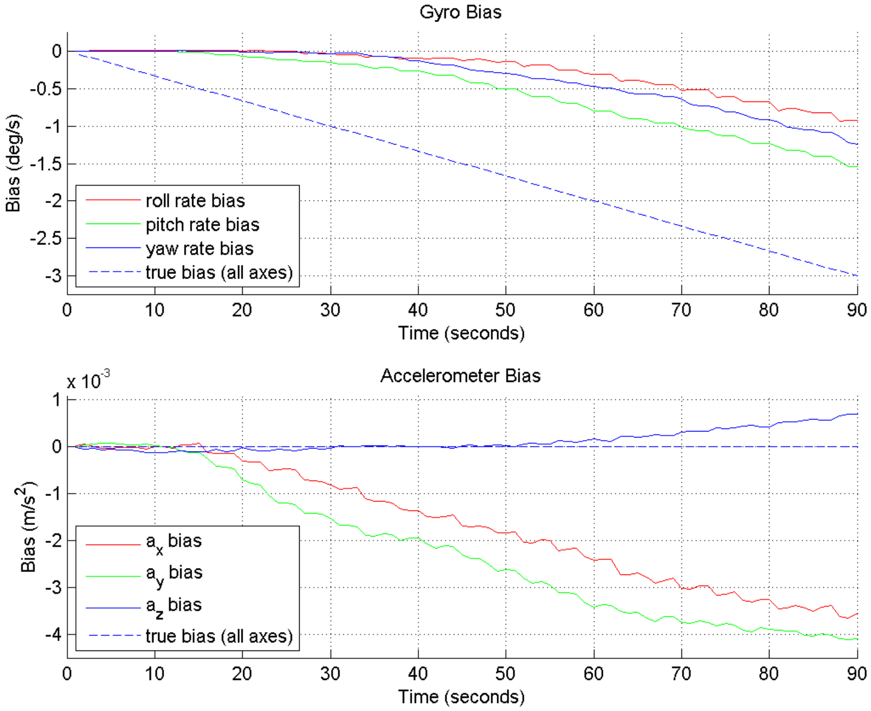


Figure 6.18: Bias Estimates for Benchmark 2 with Time-Varying Jerk Vector

This is the best performance for gyro bias estimation that is seen in any of the simulations without wind. It is comparable to benchmark 2 in wind, which will be seen shortly. By creating a jerk vector that varies its direction, the gyro bias states tracked their true values fairly equally. Unfortunately, however, the estimates still significantly lag behind their true values. As mentioned previously, this is due to the fact that the expected rate of change of the gyro bias states is zero, and thus continuous attitude error is necessary to update these states. Also unfortunately, a significant time-varying jerk vector is not a desired way to fly most aircraft.

In conclusion, the results seen for benchmark 2 match closely to benchmark 1 with the exception that the roll bias estimate does not track the true roll bias as well. Errors in the other states are the same as in previous simulations and their causes are previously explained. The cause for the poorer tracking of the roll bias and the subsequent effect on the roll angle estimate is due to the fact that there are only 2 observable attitude angle error modes and 2 observable gyro rate bias modes. In a coordinated turn, the yaw angle estimate, and subsequently the yaw rate bias, are now determinable, and their inclusion in the estimate without 3 separate modes being observable, is a degradation of one or more of the other attitude angles and their gyro rate biases when compared to straight and level flight.

6.2.2.4 Benchmark 3

The last simulation performed in zero wind is benchmark 3 for a wings level ascent. Below are the results of such a simulation.

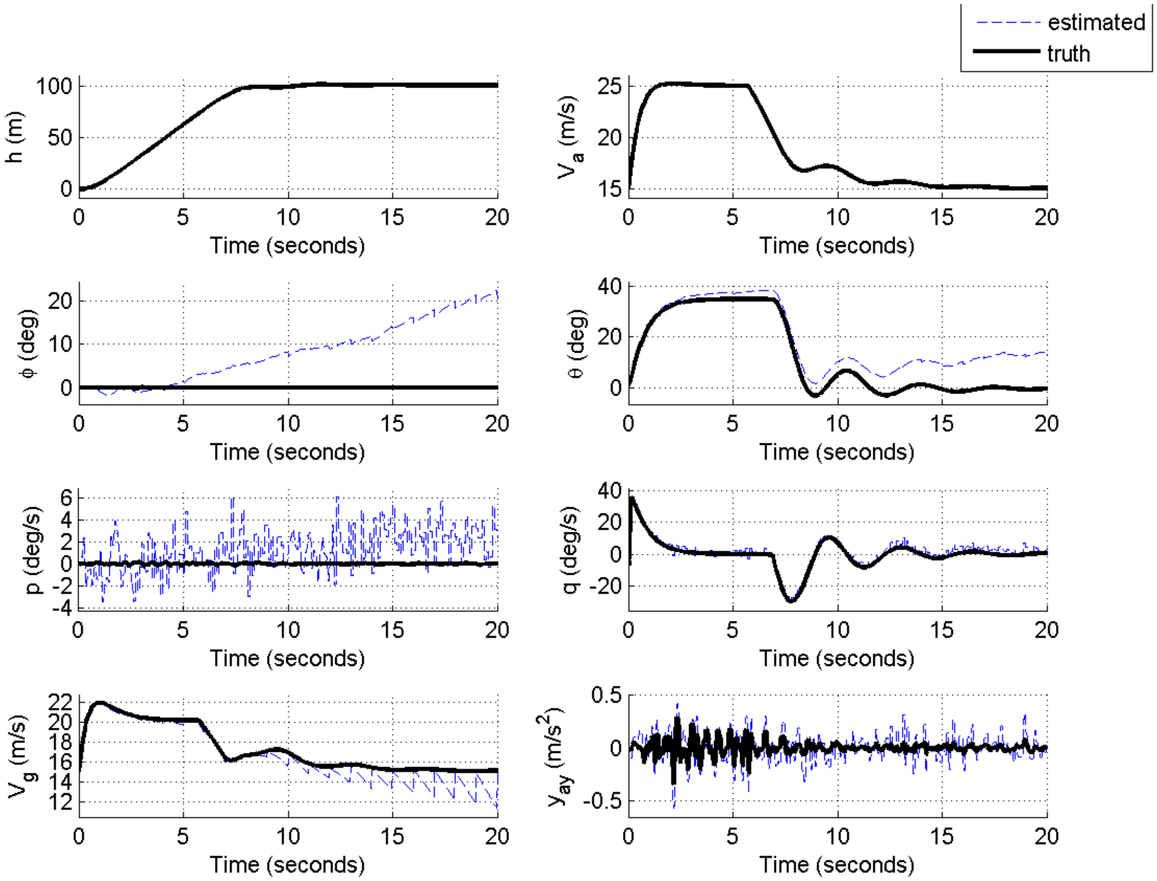


Figure 6.19: State Estimation Results for Benchmark 3 in Zero Wind

Table 6.7: State Estimation Error Statistics for Benchmark 3 in Zero Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	-0.24503	0.759922	0.692816	0.39663	0.000483	1.607249
$V_a\ (m/s)$	-0.00028	0.04083	0.018931	0.036175	0.000003	0.266219
$\phi\ (deg)$	-8.20853	6.993724	8.420343	6.737071	0.00221	22.51239
$\theta\ (deg)$	-6.48061	4.345809	6.49162	4.329342	0.003681	14.74268
$p\ (deg/s)$	-1.32631	1.950928	1.919849	1.370559	0.00043	6.221454
$q\ (deg/s)$	-1.54433	1.969362	2.08712	1.380686	0.004861	7.200226
$V_g\ \ (m/s)$	0.683986	0.78211	0.688272	0.778339	0.000137	3.797198

The first thing that should be noticed about these results is that though it does suffer from a pitch rate bias drift, the estimate for the pitch angle estimate tracks its true value much better than method 1 does. Method 2 makes no assumptions about the maneuvers an aircraft is undergoing in order to perform its state estimation. Because of this, there is little distinction between benchmarks 1, 2, and 3 in terms of the method's ability to track its states. There is no abnormal or unique behavior seen in any of these maneuvers, while method 1 introduces a steady state error in attitude in benchmark 2 and true pitch angle dependent pitch errors in benchmark 3. In all of these maneuvers, the estimate for the ground speed V_g is corrected 100% to its true value at each GPS / EKF3 measurement update. The altitude, airspeed, and side specific force track their true states with a mean zero error. And last, the angular rates p and q and the attitude angles θ and ϕ only incur steady error from bias drift, which is about to be shown.

Below are plots of the extra estimated states from method 2.

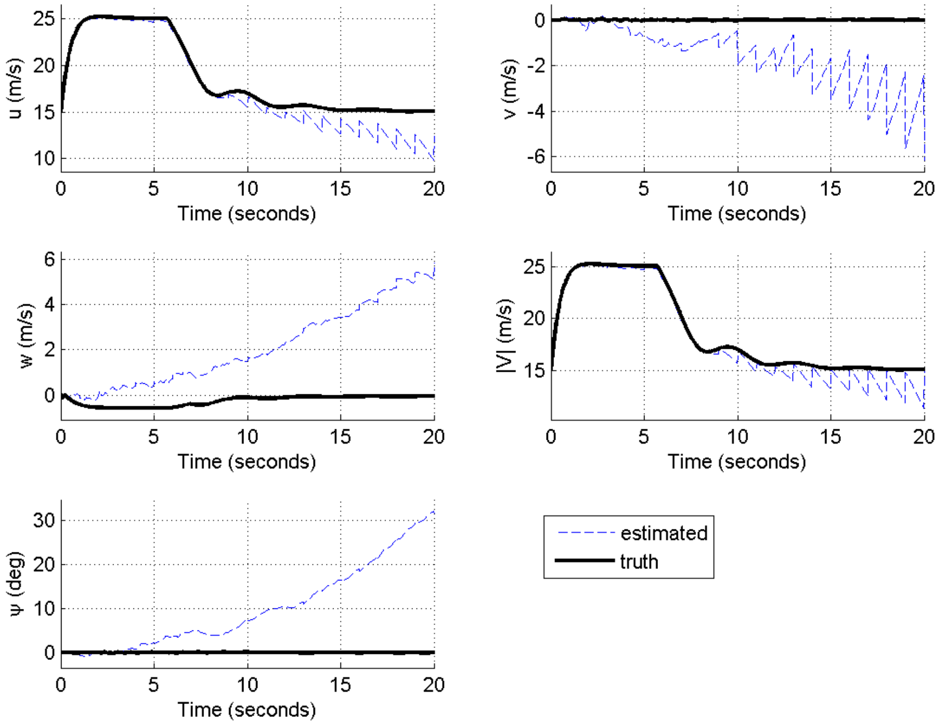


Figure 6.20: Extra Estimated States for Method 2 in Benchmark 3 and Zero Wind

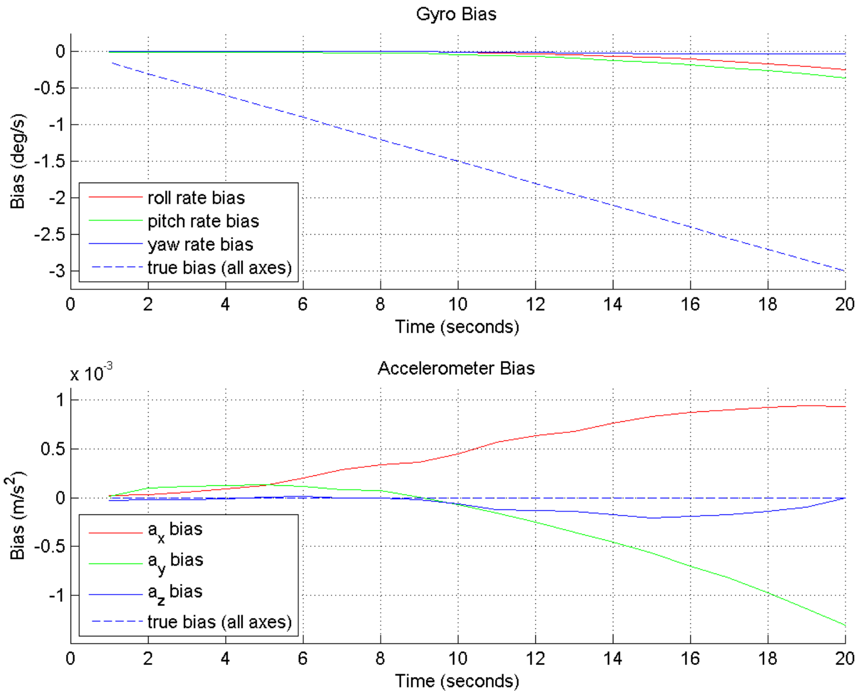


Figure 6.21: Estimated Bias States for Method 2 in Benchmark 3 and Zero Wind

Now, it should be noted that for benchmark 3, like in the method 1 simulations, the bias drift rate was increased from $\frac{3}{90}\frac{degrees}{second^2}$ to $\frac{3}{20}\frac{degrees}{second^2}$ so that the same 3 $\frac{degrees}{second}$ gyro bias is obtained by the end of the simulation. It can be seen in the above bias estimate plot that EKF3 does a poor job of estimating the bias states, as it just begins to change its bias estimate for the pitch and roll biases at the end of the simulation. By visually comparing the gyro bias estimates from benchmark 1 in Figure 6.14 and the estimates here in Figure 6.21, it can be seen that the bias estimation here looks very similar to the first 30 seconds of the benchmark 1 simulation. There is a small difference, however, in that the bias estimates here begin moving away from zero sooner than in benchmark 1. Despite the increased rate of gyro bias drift, this makes sense for a number of reasons.

First, the bias drift rate is still assumed to be zero by the system model, meaning that the gyro bias still does not change naturally, and is dependent on attitude errors to update it. It can be noticed that the attitude errors are in fact larger here than in benchmark 1 – both pitch and bank estimates grow about 5 degrees larger. This is explained by realizing that as the gyro bias estimate error grows, the attitude error grows, and thus the more the gyro bias states are changed. The gyro bias drift is faster in benchmark 3 than benchmark 1, thus the attitude error grows faster, and thus the gyro bias estimates grow faster.

Second, it makes sense that the gyro bias estimates only begin to track the pitch and bank angle rate biases. This behavior is explained by first, like in the previous benchmarks, simplifying (6.22) about the maneuver conditions. In this case that is only the assumption $\phi = 0$. This results in the following relationships in the system model:

$$\frac{\partial \delta \dot{u}}{\partial \delta \theta} = -\frac{\partial \delta \dot{v}}{\partial \delta \phi} = \left(y_{a_x} + b_{a_x}\right)\sin\theta - \left(y_{a_z} + b_{a_z}\right)\cos\theta \tag{6.40}$$

$$\frac{\partial \delta \dot{u}}{\partial \delta \psi} = -\frac{\partial \delta \dot{w}}{\partial \delta \phi} = \left(y_{a_x} + b_{a_x}\right)\sin\psi\cos\theta + \left(y_{a_y} + b_{a_y}\right)\cos\psi + \left(y_{a_z} + b_{a_z}\right)\sin\psi\sin\theta \tag{6.41}$$

$$\frac{\partial \delta \dot{v}}{\partial \delta \psi} = -\frac{\partial \delta \dot{w}}{\partial \delta \theta} = -\left(y_{a_x} + b_{a_x}\right)\cos\phi\cos\theta - \left(y_{a_y} + b_{a_y}\right)\sin\psi + \left(y_{a_z} + b_{a_z}\right)\cos\psi\sin\theta \tag{6.42}$$

This shows that, like benchmark 2, observability of each attitude angle error state is enabled by acceleration on any of the axes. However, it should be recalled that there are always at most 2 observable attitude error modes, and the unobservable mode is the angle about the jerk vector. In a rapid climb like is performed here, and in zero wind, the only substantial jerk vector is about the body Z-axis. Because the climb of the aircraft reaches only about 30 degrees in pitch, and spends much of its time at smaller pitch angles than that, the yaw angle is the angle that is less observable than the others. Because of this, it makes sense that the attitude error state never gets very large and that the subsequently derived yaw bias estimate never really grows inside the simulation duration. Figure Figure 6.20 also backs this up, as it appears that the yaw angle estimate is growing exponentially, unlike the pitch angle and bank angle estimates. Perhaps if a climb was simulated for a much lengthier amount of time, interesting dynamics between the different bias estimate states could be seen, but such a simulation is not part of this research. Most small UAVs usually do not need to climb for very large amounts of time because their operating altitude is usually not very high off of the ground.

In conclusion, like all other simulations performed thus far, no unique steady state estimates of any of the states were observed other than the large attitude errors in modelling the gyro bias drifts. Method 2 does not suffer the same pitch estimate behavior under benchmark 3 as method 1 does. However, errors in gyro bias state estimation still greatly hinder the accuracy of the attitude estimation in this benchmark.

6.2.3 Dryden Wind

Now that the raw performance of method 2 without any wind has been analyzed, Dryden wind can be added to the simulations so that the effects of wind on method 2's performance can be assessed.

6.2.3.1 Benchmark 1 without Gyro Bias Drift

The first simulation of method 2 in wind that is performed is benchmark 1, straight and level flight, but without gyro bias drift. Exclusion of gyro bias enables analysis of the direct effects of wind, as it removes the effects of gyro bias. Below are the results of this simulation.

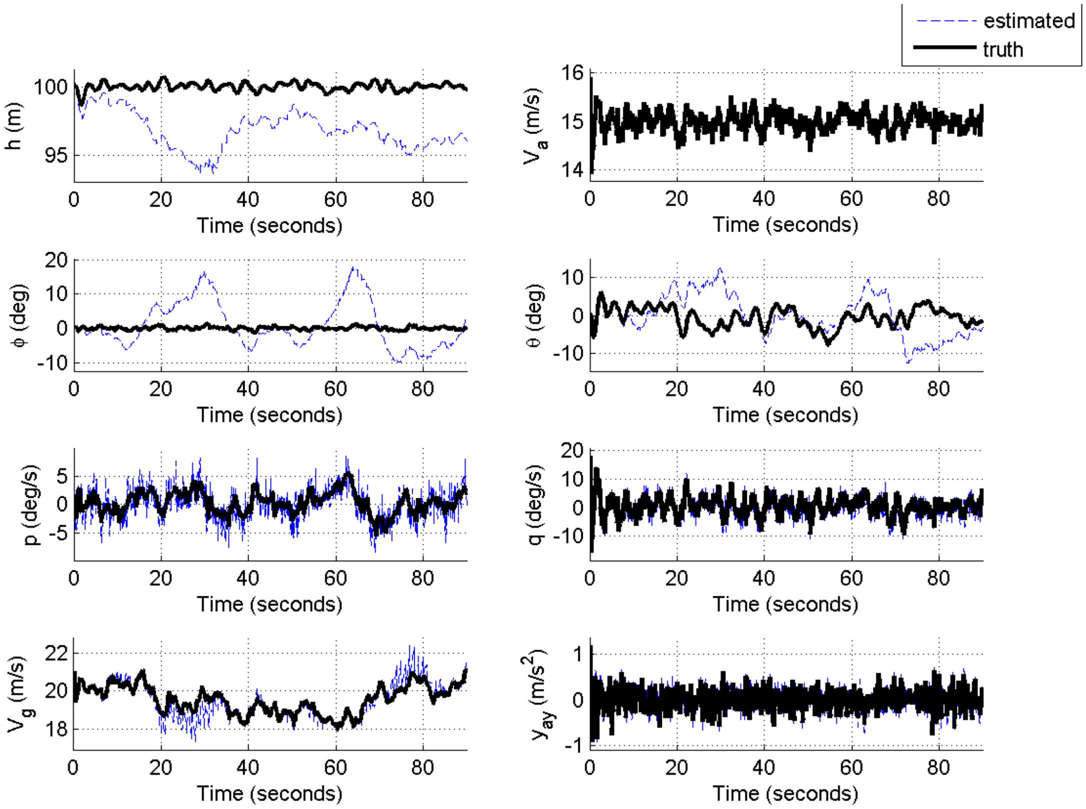


Figure 6.22: State Estimation Results for Method 2 Benchmark 1 in Dryden Wind and Zero Bias Drift

Table 6.8: State Estimation Error Statistics for Method 2 Benchmark 1 in Dryden Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	-0.21503	1.40745	1.12608	0.871184	0.000011	2.941845
$V_a\ (m/s)$	-0.00033	0.030879	0.019375	0.024045	0.000007	0.265249
$\phi\ (deg)$	-0.95443	6.605894	5.21015	4.171354	0.000301	17.27692
$\theta\ (deg)$	-0.5144	5.852534	4.524938	3.746925	0.000066	14.61207
$p\ (deg/s)$	0.19336	1.756825	1.407942	1.068318	0.000424	5.3913
$q\ (deg/s)$	0.179876	1.807966	1.428433	1.122698	0.000165	7.73369
$V_g\ (m/s)$	0.036958	0.409217	0.27293	0.307125	0.000001	1.704375

These results show the effects of gyro bias are not present, as there is no drift in the estimated pitch and roll rates. Unlike the previous method 2 simulations, there are now significant oscillations in the true values for all components. These oscillations in the true values are direct result of the force and moment disturbances imparted on the aircraft from the Dryden wind model and the autopilot's reaction to these forces and moments. Since this is the same wind model and autopilot used in method 1's simulations, these variations are very similar, but not identical, as the wind uses an unseeded randomly generated noise model.

Now, the first thing that should be noticed when comparing these results to the zero wind results in section 6.2.2.1 is that there is a much larger variation in the estimated pitch and bank angles. Here, the bank angle estimate error becomes as large as 17 degrees, and without wind it only reaches an error of 3 degrees. Similarly, here the pitch angle estimate error becomes as large 14 degrees, and without wind it only reaches an error of just under 3 degrees. In order to find why this is happening, the additional source of error should first be determined.

To determine the source of the extra attitude error, it should be recalled that the attitude angle estimates are a direct output of the IMU integration and is only corrected by EKF3 measurements when it can be. At each IMU measurement update, which is the same time as EKF3's time update to the covariance matrix, the attitude estimate is updated by the simple integration depicted in equations (6.3) and (6.4). In those equations, the current attitude estimate $\widehat{\boldsymbol{\psi}}_k$ is computed from the a-priori attitude estimate $\widehat{\boldsymbol{\psi}}_{k-1}$ and the current angular rate measurements $\widehat{\boldsymbol{\omega}}_k$. It could be expected that this integration would produce more error in the IMU attitude estimate in wind than without wind due to the more variable nature of the aircraft's attitude $\boldsymbol{\psi}$ and angular rates $\boldsymbol{\omega}$.

The effects of wind on integration error can be measured directly in the simulations, as the true known states are available. To measure the integration error, the absolute value of the difference between the true attitude from the previous time step and the current true attitude can be written as equation (6.34).

$$\mathbf{\Delta\psi}_{\mathrm{k}} = \begin{bmatrix} \Delta\phi_k \\ \Delta\theta_k \\ \Delta\psi_k \end{bmatrix} = \left\| \begin{bmatrix} \phi_k \\ \theta_k \\ \psi_k \end{bmatrix} - \begin{bmatrix} \phi_{k-1} \\ \theta_{k-1} \\ \psi_{k-1} \end{bmatrix} \right\| \tag{6.43}$$

The capital delta (Δ) symbol is used here to designate the difference between IMU updates so that these quantities will not be confused with EKF3's error states, where the lower case delta symbol (δ) is used.

Now that the true delta across the IMU attitude states $\boldsymbol{\psi}_{k-1}$ and $\boldsymbol{\psi}_k$ has been determined, the error between the true difference $\mathbf{\Delta\psi}$ and the estimated difference can be found. The difference between the true attitude change and the estimated attitude change can be found by rewriting equations (6.3) and (6.4) as equation (6.44).

$$\widehat{\boldsymbol{\Delta\psi}}_k = \begin{bmatrix} \widehat{\Delta\phi}_k \\ \widehat{\Delta\theta}_k \\ \widehat{\Delta\psi}_k \end{bmatrix} = \left| \begin{bmatrix} \Delta\phi_k \\ \Delta\theta_k \\ \Delta\psi_k \end{bmatrix} - T_s \left(\begin{bmatrix} 1 & \sin\hat{\phi}_{k-1}\tan\hat{\theta}_{k-1} & \cos\hat{\phi}_{k-1}\tan\hat{\theta}_{k-1} \\ 0 & \cos\hat{\phi}_{k-1} & -\sin\hat{\phi}_{k-1} \\ 0 & \sin\hat{\phi}_{k-1}\sec\hat{\theta}_{k-1} & \cos\hat{\phi}_{k-1}\sec\hat{\theta}_{k-1} \end{bmatrix} \begin{bmatrix} \hat{p}_k \\ \hat{q}_k \\ \hat{r}_k \end{bmatrix} \right) \right| \tag{6.44}$$

While comparing this value between wind scenarios and zero wind scenarios will yield insights on the effects of wind, it by itself will not yield insights on the individual effects of bad estimates for attitude and estimates for the angular rates. Thus, two more descriptions of the change across the IMU can be defied:

$$\overbrace{\boldsymbol{\Delta\psi}}_k = \begin{bmatrix} \overbrace{\Delta\phi}_k \\ \overbrace{\Delta\theta}_k \\ \overbrace{\Delta\psi}_k \end{bmatrix} = \left| \begin{bmatrix} \Delta\phi_k \\ \Delta\theta_k \\ \Delta\psi_k \end{bmatrix} - T_s \left(\begin{bmatrix} 1 & \sin\phi_{k-1}\tan\theta_{k-1} & \cos\phi_{k-1}\tan\theta_{k-1} \\ 0 & \cos\phi_{k-1} & -\sin\phi_{k-1} \\ 0 & \sin\phi_{k-1}\sec\theta_{k-1} & \cos\phi_{k-1}\sec\theta_{k-1} \end{bmatrix} \begin{bmatrix} \hat{p}_k \\ \hat{q}_k \\ \hat{r}_k \end{bmatrix} \right) \right| \tag{6.45}$$

$$\overline{\boldsymbol{\Delta\psi}}_k = \begin{bmatrix} \overline{\Delta\phi}_k \\ \overline{\Delta\theta}_k \\ \overline{\Delta\psi}_k \end{bmatrix} = \left| \begin{bmatrix} \Delta\phi_k \\ \Delta\theta_k \\ \Delta\psi_k \end{bmatrix} - T_s \left(\begin{bmatrix} 1 & \sin\phi_{k-1}\tan\theta_{k-1} & \cos\phi_{k-1}\tan\theta_{k-1} \\ 0 & \cos\phi_{k-1} & -\sin\phi_{k-1} \\ 0 & \sin\phi_{k-1}\sec\theta_{k-1} & \cos\phi_{k-1}\sec\theta_{k-1} \end{bmatrix} \begin{bmatrix} p_k \\ q_k \\ r_k \end{bmatrix} \right) \right| \tag{6.46}$$

Equation (6.45) computes the IMU change in attitude estimate $\boldsymbol{\widetilde{\Delta\psi}}$ as if there was zero attitude error, but still uses the estimated angular rates that is determined from the current gyro measurements and EKF3 gyro bias states. Equation (6.46) computes the IMU change in attitude estimate $\boldsymbol{\overline{\Delta\psi}}$ as if there was zero error in both of these measurements. By using 100% true values as inputs, the output of equation (6.46) provides insights into the accuracy of this integration method, and by comparing its value to the other versions of the IMU change in attitude estimate, the error from the inaccuracy of the integration method can be compared to attitude error generated from a-priori attitude error and current angular rate error.

The table below contains the average value of these three IMU update magnitudes over the course of the 90 second benchmark 1 with zero bias drift simulation. The first three columns of the table are the components of the measured average error, and the last column is the sum of these three components.

Table 6.9: IMU Attitude Propagation Error for Benchmark 1 in Dryden Wind and Zero Rate Bias

	$\boldsymbol{\Delta\phi}$ (degrees)	$\boldsymbol{\Delta\theta}$ (degrees)	$\boldsymbol{\Delta\psi}$ (degrees)	$\boldsymbol{\Delta\phi+\Delta\phi+\Delta\phi}$ (degrees)
Mean Error with True Attitude and Angular Rates $(\boldsymbol{\Sigma\overline{\Delta\psi}_k/n})$	0.01687	0.017239	0.018454	0.052563
Mean Error with True Attitude $\left(\boldsymbol{\Sigma\,\widetilde{\Delta\psi}_k\,/n}\right)$	0.022329	0.022629	0.023645	0.068603
Mean Estimate Error $\left(\boldsymbol{\Sigma\widehat{\Delta\psi}_k/n}\right)$	0.022413	0.022542	0.024028	0.068983

The variable n is the number of IMU updates in the simulation, which is the duration of the simulation multiplied by the sample rate of the IMU in *Hz*. In all simulations in this document, the IMU sample rate is $100\ Hz$. Thus, n in this simulation is 9,000.

This information provided in the above table yields a very insightful conclusion. There is only a marginal difference between the total of the components of the mean estimate error and the mean error of the estimate that used only the true attitude. Then, there is only a 24% drop in the estimate error when both true attitude states and true angular rates are used. This indicates that around 76% of the error of the IMU attitude propagation actually comes from the integration method itself. Even with true states are used, 76% of the error in the attitude estimate is still present. Thus, a future effort could endeavor to address this integration problem, perhaps by using higher orders of integration or by using a more appropriate mix of a-priori and current data. Interpolation between data points could perhaps be used in conjunction with a higher order integrator to reduce this percentage. However, such a study is outside the scope of this research. That said, it is still unreasonable to expect 100% accuracy from an integration method when only discrete values are provided. In order to capture all of the dynamics of the angular rates in wind, the maximum frequency of the change of the rates would have to been know, and the sample rate would have meets Nyquist criterion (twice that frequency). Unfortunately, the noise used in the Dryden model is wide band, so this knowledge is not really possible.

Now, it is important to note that this high percentage of error being due to the integration method is only the case for the Dryden wind scenario. Below is a table of these same measures for Benchmark 1 with zero bias drift in zero wind.

Table 6.10: IMU Attitude Propagation Error for Benchmark 1 in Zero Wind and Zero Rate Bias

	$\boldsymbol{\Delta\phi}$ (degrees)	$\boldsymbol{\Delta\theta}$ (degrees)	$\boldsymbol{\Delta\psi}$ (degrees)	$\boldsymbol{\Delta\phi + \Delta\phi + \Delta\phi}$ (degrees)
Mean Error with True Attitude and Angular Rates $\left(\boldsymbol{\Sigma\,\overbrace{\widehat{\Delta\psi}}\,/n}\right)$	0.000012	0.000073	0.000061	0.000146
Mean Error with True Attitude $(\boldsymbol{\Sigma\overline{\Delta\psi}/n})$	0.013861	0.014003	0.014243	0.042108
Mean Estimate Error $\left(\boldsymbol{\Sigma\widehat{\Delta\psi}/n}\right)$	0.013875	0.014	0.01426	0.042136

This shows that in zero wind, the error of each IMU integration step is roughly 50% smaller than error in windy conditions. More importantly however, it shows that the error of the estimate when the true attitude and angular rates are used is essentially zero, and the error when only true attitude used is essentially identical to the actual estimate's error. This means that essentially 100% of the error in the wind-free scenario is due to errors in the angular rate estimates. The angular rate estimates are affected by both the gyro sensor noise and the accuracy of the current gyro rate bias estimate.

It could also be noticed that in the straight and level scenario, there was comparatively very little error in the attitude estimates. One could postulate that with more attitude error, more error could be attributed the attitude solution. However, in benchmark 2, the effect of the larger attitude error on the IMU propagation error is still negligible. This makes sense, as the Euler angle propagation equation of (3.8) in straight and level flight directly maps the angular rates one to one to the Euler angles. Some attitude errors creates small cross-coupling terms, but for the most part, the equation remains a one-to-one mapping. This behavior is different in other scenarios, and is addressed in the explanation of those simulations.

Now that the source of the extra attitude error that is encountered with wind has been identified, how that error is accounted for by EK3 needs to be addressed. To support that analysis, attitude error observability must be re-addressed. Thus, below are the results of the estimation of the remaining EKF3 states.

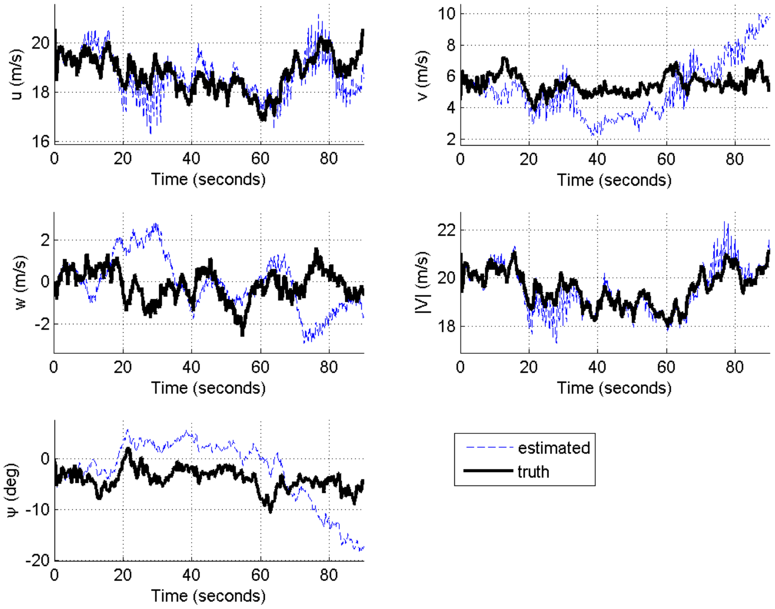


Figure 6.23: Extra Estimated States for Method 2 Benchmark 1 in Wind with Zero Rate Bias

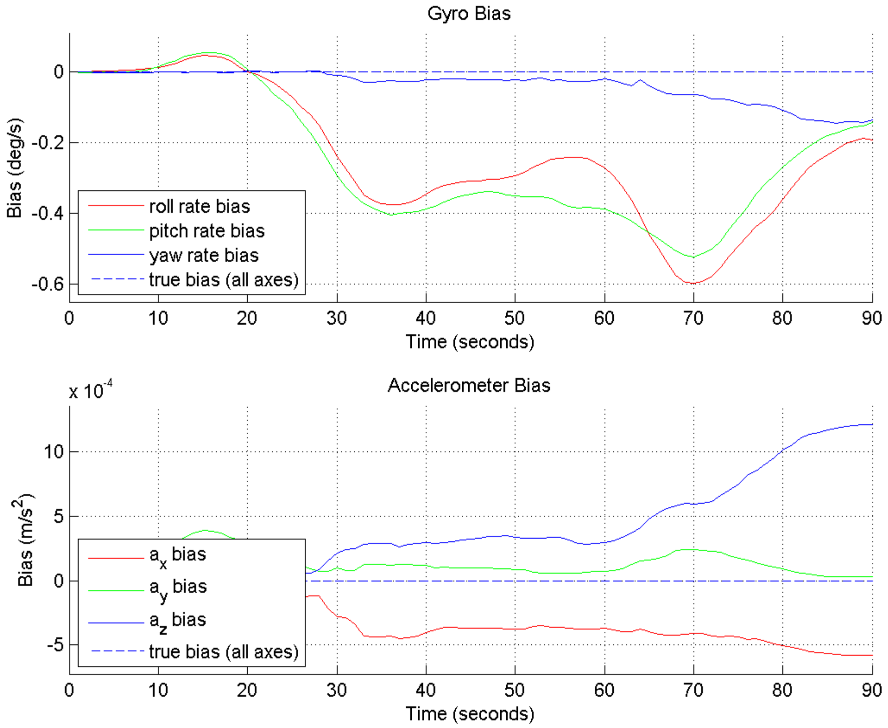


Figure 6.24: Sensor Bias Estimates for Method 2 Benchmark 1 in Wind with Zero Rate Bias

The sensor bias estimate plot above indicates that unlike in the zero wind and zero bias scenario, the estimate for all three bias axes varies significantly. It makes sense that this happens because continually large attitude errors, which are seen here in wind but not in the zero wind scenarios, will manifest in the attitude errors per equation (6.25). Without wind, attitude errors are quickly corrected out, and thus the only reason that the bias states would be populated is if there is a constant attitude error, as occurs from gyro sensor biases. With wind, the IMU solution has shown to cause large errors that are not immediately corrected out. They are corrected out over time, but this continual attitude error builds up into the gyro biases. Fortunately though, when the large attitude errors are corrected out, the biases also go back to zero, and thus the bias estimates stay around zero.

Now, the question remains of why aren't these attitude errors corrected out just as fast as without wind. To understand that, the average Kalman gain that adjusts the attitude states can be measured. The below table contains the average value of the sum of the 7th, 8th, and 9th row of the Kalman gain matrix, which correspond to the amount that the bank, pitch, and yaw angles, respectively, are updated at each EKF3 measurement update. Further, it is only the sum of the 4th, 5th, and 6th columns of each row, because those are the columns that correspond to the velocity error measurement. The units of these average sums are radians per meter per second, as they are the gain that maps the measured velocity error to the attitude error state. The other first three columns in the table are always very close to zero, as position error is not directly related to the attitude error states in the system model, and any non-zero value hat is in those columns is due to the propagation of the covariance between the attitude error and velocity error states through the system model relation between the velocity error and the position error states.

Table 6.11: Kalman Gain Averages for Attitude Error States from Benchmark 1 in Dryden Wind with Zero Rate Bias

	$\boldsymbol{\phi}$ Gain: $\overline{\boldsymbol{\Sigma K(7,4:6)}}$	$\boldsymbol{\theta}$ **Gain**: $\overline{\boldsymbol{\Sigma K(8,4:6)}}$	$\boldsymbol{\psi}$ Gain: $\overline{\boldsymbol{\Sigma K(9,4:6)}}$
Average Kalman Gain (radians/meter/s)	0.01063	0.010652	0.016765

These values for the benchmark 1 in wind simulation can be compared to the same values for the zero wind simulation, which are shown below:

Table 6.12: Kalman Gain Averages for Attitude Error States from Benchmark 1 in Zero Wind with Zero Rate Bias

	$\boldsymbol{\phi}$ Gain: $\overline{\boldsymbol{\Sigma K(7,:)}}$	$\boldsymbol{\theta}$ **Gain**: $\overline{\boldsymbol{\Sigma K(8,:)}}$	$\boldsymbol{\psi}$ Gain: $\overline{\boldsymbol{\Sigma K(9,:)}}$
Average Kalman Gain (radians/meter/s)	0.009577	0.009592	0.005249

It can be noticed that in wind, there is a slight increase in the Kalman gain for the bank and pitch angles and a substantial increase in the Kalman gain for the yaw angle. This makes sense because as previously discussed, in straight and level flight the yaw angle is much less observable because there are ideally no X-Y plane accelerations. However, in wind, the aircraft experiences significant accelerations from all directions.   Table 6.3 shows that when accelerations in the X-Y plane are added, the yaw angle error is able to be observed. In fact, the Kalman gains presented here show that all three attitude angles are corrected more in wind than they are in zero wind.   This notion agrees with the observability research performed in [22], as in windy conditions, the jerk vector varies significantly in both magnitude and direction.

So, in conclusion, wind assists in the observability of all of the attitude error states. While it has been shown that the attitude estimates are corrected more in wind than without wind, integration errors that arise from the integration method cause a net growth in attitude estimate error when Dryden wind is added to the simulation. Further, it has been shown that the estimates of the states that are input into the IMU integration equation (6.3) are not the reason for the majority of the error, it is in fact the integration method itself, which provides a nice segue for future work.

6.2.3.2 Benchmark 1

Now that the performance of state estimation method 2 in Dryden in wind has been analyzed without gyro bias drift, gyro bias drift can be added back into the simulation so that additional effects of the gyro bias may be observed.

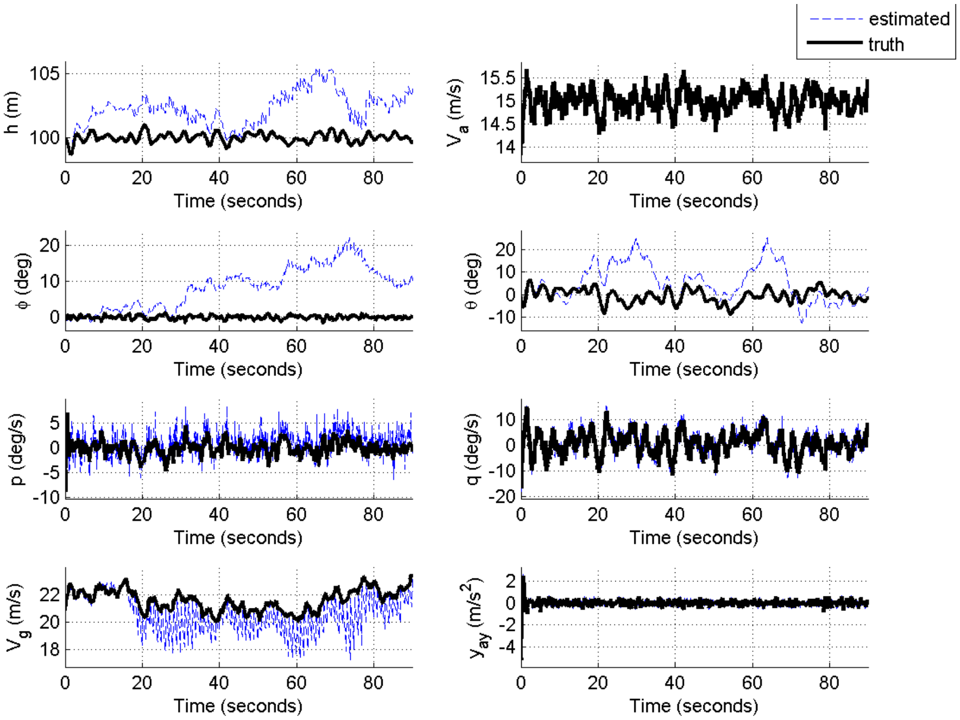


Figure 6.25: State Estimation Results for Method 2 in Benchmark 1 and Dryden Wind

Table 6.13: State Estimation Error Statistics for Method 2 in Benchmark 1 and Dryden Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}

$h\ (m)$	-2.29423	1.207648	2.295253	1.205711	0.004419	5.014218
$V_a\ (m/s)$	0.000185	0.032004	0.020298	0.024743	0.000007	0.283372
$\phi\ (deg)$	-8.46745	5.769247	8.522576	5.68748	0.000181	21.90765
$\theta\ (deg)$	-5.77646	8.164708	7.734534	6.340662	0.000182	25.50788
$p\ (deg/s)$	-0.99568	1.816456	1.636558	1.269792	0.00028	6.935281
$q\ (deg/s)$	-0.51144	1.796622	1.49002	1.126518	0.000272	7.005333
$V_g\ \ (m/s)$	0.971821	0.891811	0.99825	0.862122	0.000085	4.969819

The estimates here for every state except the pitch and bank angles and a barely visible drift in the estimates for the angular rates p and q are very similar to the benchmark 1 simulation without gyro rate bias. This makes sense, as the gyro bias states affect the attitude angles the most, and are used to calculate V_g and h at every IMU update, but V_g and h are essentially 100% corrected at each GPS update as discussed previously.

There is no unexpected extra behavior in these results, as in the increase in attitude error can be related to the addition of gyro bias drift. The below plots contain the extra states that method 2 produces, including the bias drifts.

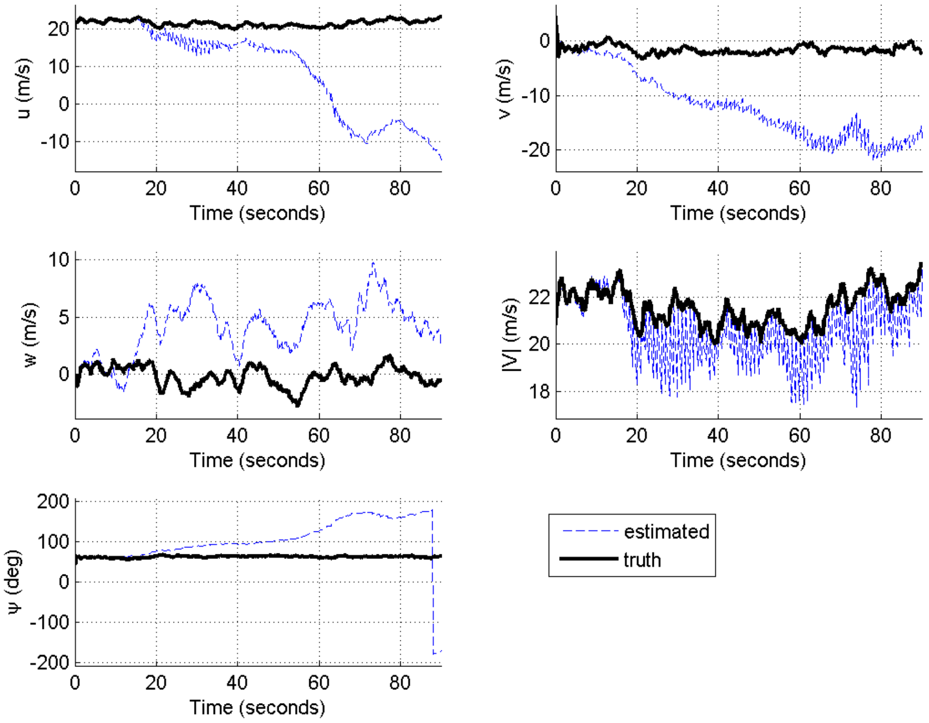


Figure 6.26: Extra State Estimates for Method 2 in Benchmark 1 and Dryden Wind

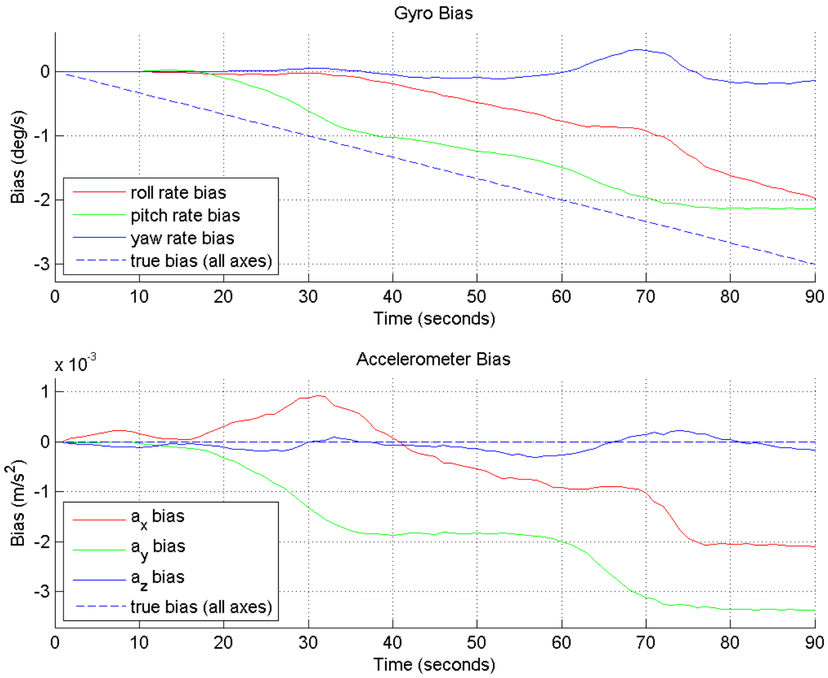


Figure 6.27: Bias State Estimates for Method 2 in Benchmark 1 and Dryden Wind

The unbounded behavior of the yaw angle estimate and the fact that the yaw bias state remains close to zero indicates that the yaw angle is the unobservable mode in this scenario. This matches all the other benchmark 1 simulations, and makes sense because even though there are now disturbances that give accelerations in the X-Y plane that enable the yaw error to be observed, the majority of the acceleration still comes from gravity. Because in straight and level flight, the pitch and bank angles are coupled to all three axes' acceleration where the yaw angle is only coupled to X and Y axes accelerations, the yaw angle error is less observable than the other angle errors.

Now, it should be recalled that without gyro bias drift, only 24% of the IMU propagation error arose from errors in estimates for the angular rates. The below table shows the components of error in the attitude propagation for this simulation with gyro bias drift added.

Table 6.14: IMU Attitude Propagation Error for Benchmark 1 in Dryden Wind

c	$\boldsymbol{\Delta\phi}$ (degrees)	$\boldsymbol{\Delta\theta}$ (degrees)	$\boldsymbol{\Delta\psi}$ (degrees)	$\boldsymbol{\Delta\phi + \Delta\phi + \Delta\phi}$ (degrees)
Mean Error with True Attitude and Angular Rates $(\boldsymbol{\Sigma\overline{\Delta\psi_k}/n})$	0.01687	0.017239	0.018454	0.052563
Mean Error with True Attitude $\left(\boldsymbol{\Sigma\,\widetilde{\Delta\psi_k}\,/n}\right)$	0.022329	0.022629	0.023645	0.068603
Mean Estimate Error $\left(\boldsymbol{\Sigma\widehat{\Delta\psi_k}/n}\right)$	0.022413	0.022542	0.024028	0.068983

These results show that with bias drift, 30% of the attitude error comes from poor knowledge of the angular rate estimates. It makes sense that this percentage is higher than without bias drift because now the bias estimates how significant error. However, the above table shows that the integration method is still responsible for 70% of the attitude error. Nonetheless, poor gyro rate bias estimation can still cause long-term biases in the attitude states, as is shown in the estimate for the bank angle in this simulation. Fortunately though, those biases do not always last since their errors are observable, as is shown in the estimate for the pitch angle in this simulation, and is shown better next in benchmark 2.

In conclusion, the only change when gyro rate bias drift was added to the simulation of benchmark 1 was an increase in attitude error. While the attitude error increased, the majority of the attitude error can still be blamed on the integration method and not the gyro rate bias estimation, as the attitude error generated by the rate bias can still be accounted for by EFK3. However, that error can only be removed if those attitude angles are observable. In this benchmark, the yaw angle is the unobservable angle, so the pitch and bank angles are effectively corrected by EFK3.

6.2.3.3 Benchmark 2

This section contains the results of simulating method 2 state estimation under a coordinated turn maneuver in Dryden wind.

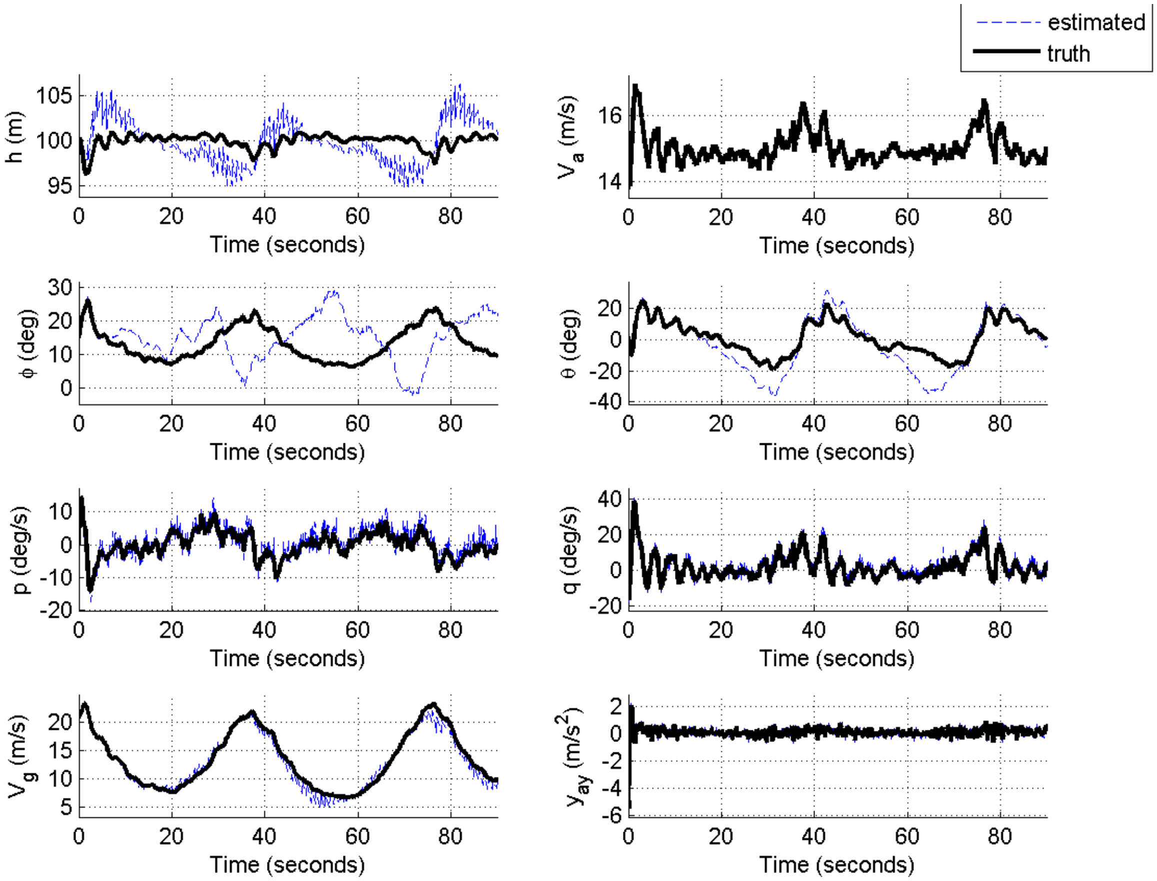


Figure 6.28: State Estimation Results for Method 2 in Benchmark 2 in Dryden Wind

Table 6.15: State Estimation Error Statistics for Method 2 in Benchmark 2 in Dryden Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	0.030457	2.323744	1.929098	1.295716	0.000046	6.508412
$V_a\ (m/s)$	0.000179	0.026681	0.017143	0.020445	0.000002	0.285108
$\phi\ (deg)$	-1.79709	9.909461	8.241748	5.787345	0.000292	22.06134
$\theta\ (deg)$	3.551419	8.174121	6.504409	6.092471	0.00085	25.6019
$p\ (deg/s)$	-1.03095	1.851849	1.679188	1.29319	0.000197	6.998946
$q\ (deg/s)$	-1.28827	1.872867	1.858952	1.308192	0.000633	7.484872
$V_g\ \ (m/s)$	0.351005	0.817978	0.638282	0.620368	0.000086	4.469241

These results show that like Benchmark 1 in wind, there is as substantial increase in the magnitude of the attitude error when wind is added to the simulation. Also like in benchmark 1 in wind, there is no noticeable increase in error in any of the other states, except for the altitude and ground speed errors which are corrected at every GPS update as they have in all the other previous simulations. However, it can be noticed that there is no longer a noticeable drift in the pitch or bank angle estimates; there are only wild swings in the estimate errors. Do determine why that is, the table below contains the errors introduced from the IMU propagation from the IMU propagator's different sources of information.

Table 6.16: IMU Attitude Propagation Error in Benchmark 2 and Dryden Wind

	$\boldsymbol{\Delta\phi}$ (degrees)	$\boldsymbol{\Delta\theta}$ (degrees)	$\boldsymbol{\Delta\psi}$ (degrees)	$\boldsymbol{\Delta\phi + \Delta\phi + \Delta\phi}$ (degrees)
Mean Error with True Attitude and Angular Rates $(\boldsymbol{\Sigma\overline{\Delta\psi_k}/n})$	0.016291	0.022511	0.137849	0.176651
Mean Error with True Attitude $\left(\boldsymbol{\Sigma\,\overbrace{\Delta\psi_k}\,/n}\right)$	0.023908	0.025817	0.145326	0.195051
Mean Estimate Error $(\boldsymbol{\Sigma\widehat{\Delta\psi_k}/n})$	0.02658	0.026475	0.146673	0.199729

This table shows that the vast majority, 88%, of the error introduced from the attitude propagation is incurred from the integration itself. When true attitude and true angular rate information is fed through the integrator, 88% of the error still remains. The table also shows that the error produced from the actual estimate and the error produced from the estimate with just the true attitude angles are essentially the same; only 2.3% of the error can be attributed to the attitude estimates. This means that only 10% of the IMU propagation error can be attributed to bad estimates for the angular rates. This is a decrease in the attitude error's dependence on the gyro bias estimates when compared to benchmark 1 in wind, where 30% of the error was attributed to the angular rate error. A similar improvement can also be seen with respect to benchmark 2 in zero wind, where according to the table below, about 30% of the error is due to the angular rate estimates.

Table 6.17: IMU Attitude Propagation Error in Benchmark 2 and Zero Wind

	$\boldsymbol{\Delta\phi}$ (degrees)	$\boldsymbol{\Delta\theta}$ (degrees)	$\boldsymbol{\Delta\psi}$ (degrees)	$\boldsymbol{\Delta\phi + \Delta\phi + \Delta\phi}$ (degrees)
Mean Error with True Attitude and Angular Rates $(\boldsymbol{\Sigma\overline{\Delta\psi_k}/n})$	0.000035	0.00372	0.120072	0.123827
Mean Error with True Attitude $\left(\boldsymbol{\Sigma\,\overbrace{\Delta\psi_k}\,/n}\right)$	0.019659	0.014726	0.13992	0.174306
Mean Estimate Error $(\boldsymbol{\Sigma\widehat{\Delta\psi_k}/n})$	0.014844	0.0146	0.140437	0.169881

This decreased dependence of the attitude error on the angular rate error over both benchmark 2 in zero wind and benchmark 1 in wind can be attributed to an improved estimate for the gyro rate bias, as shown below. The only other component of the angular rate error is the noise of the gyro sensors, which does not change from simulation to simulation.

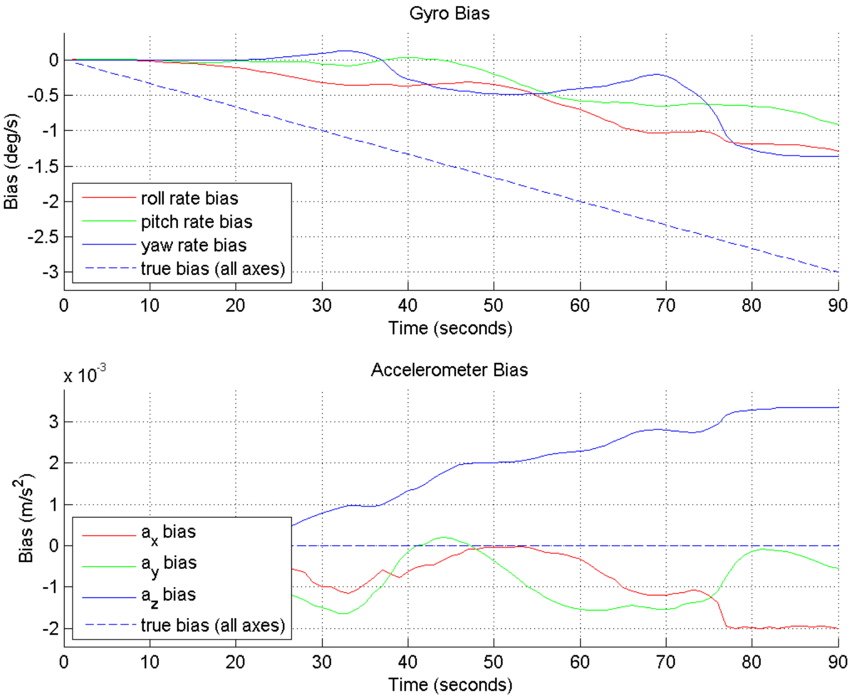


Figure 6.29: Gyro Bias Estimates for Benchmark 2 in Dryden Wind

This plot shows improved gyro bias tracking over benchmark 2 in zero wind. Further, the magnitude of the error of the gyro bias estimates is on par with the extra maneuvers simulation performed in section 6.2.2.3 with benchmark 2 in zero wind. This makes sense, as the poor observability of the roll and yaw rate biases seen in benchmark 2 in zero wind was attributed to the lack of a variable jerk vector. It was shown in section 6.2.2.3 that by adding an aircraft acceleration that varies in magnitude and direction, the unobservable attitude error mode and thus the unobservable gyro bias mode become shared amongst all the axes. The addition of wind also performs this action by creating forces and moments on the aircraft that the autopilot then corrects. This is perhaps the most important realization about method 2's attitude estimation performance. By performing significant maneuvers or from being under the influence of Dryden wind, the observability and thus the long-term accuracy of the attitude errors and gyro bias estimates is maximized. There are still only 2 observable attitude error and gyro bias modes at any given point in time, but by performing maneuvers, errors in the different axes can be found. In the context of a coordinated turn, the pitch and bank angle errors, even with significant gyro bias drift, are corrected to zero mean error. Unfortunately, the IMU propagation method of method 2, at least in the form exists here, causes large variations in the attitude estimates when Dryden wind is added. Nonetheless, the improved observability of the attitude estimates from the wind is able to continually correct out the pitch and roll attitude error to keep the attitude solution bounded and have zero mean error.

Last, below are the extra states that are estimated with method 2 for this simulation.

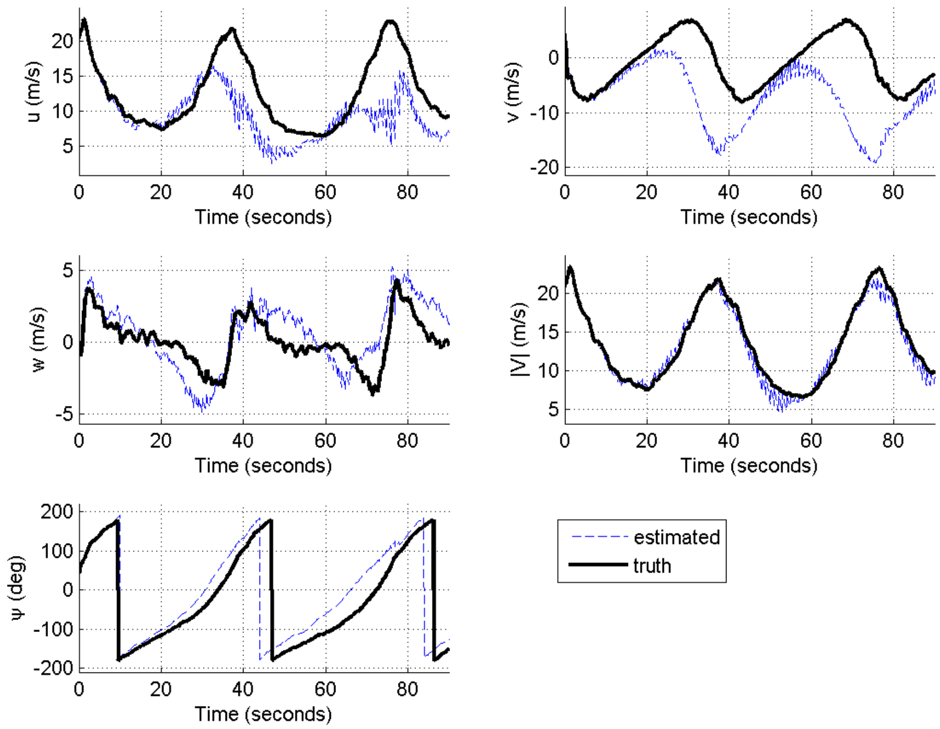


Figure 6.30: Extra State Estimation Results for Method 2 in Benchmark 2 and Dryden Wind

These results show that for benchmark 2 in zero wind, unlike the roll and pitch estimates, there is a slight long-term bias in the yaw angle estimate. This is still an improvement over benchmark 2 in zero wind, but it makes sense that the yaw angle error is less observable than the other attitude errors, given that most of the acceleration experienced by the aircraft, even in wind, is due to gravity. These results also show significant errors in the body frame components of the inertial velocity vector, but as shown in Figure 6.28, the errors incurred into the ground speed and altitude estimates from these measures are still corrected out at each GPS measurement update.

In conclusion, benchmark 2 in Dryden wind suffers from additional attitude error when compared to benchmark 2 without wind. The vast majority of this error actually comes from the integration method of the IMU estimates and not the actual estimates for the angular rates or attitude. Further, unlike benchmark 2 in zero wind, the attitude estimates have near zero mean error. This centering of the attitude estimates on the true values is due to the increased attitude error and gyro bias observability encountered when variable accelerations are encountered from the Dryden wind.

6.2.3.4 Benchmark 3

The final method 2 simulation performed is for a wings level ascent in Dryden wind. Below are the results of this simulation.

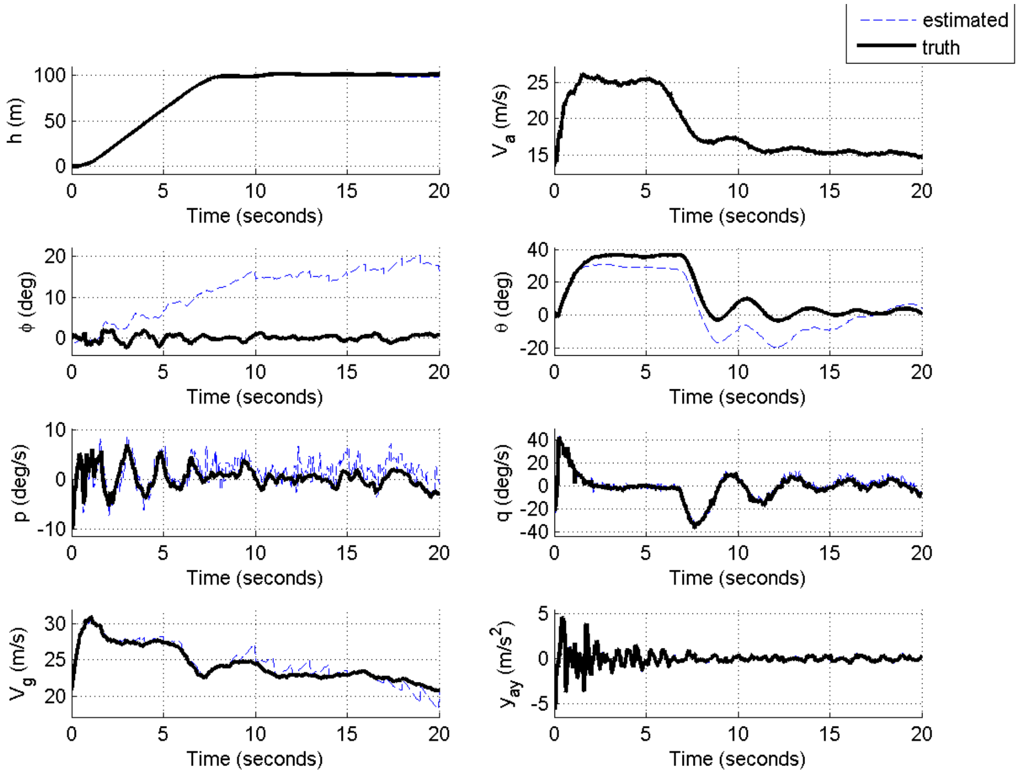


Figure 6.31: State Estimation Results for Method 2 in Benchmark 3 and Dryden Wind

Table 6.18: State Estimation Error Statistics for Method 2 in Benchmark 3 and Dryden Wind

	μ	σ	μ_{abs}	σ_{abs}	min_{abs}	max_{abs}
$h\ (m)$	0.828567	0.749619	0.830541	0.74743	0.000125	3.003041
$V_a\ (m/s)$	0.000444	0.071805	0.0376	0.061169	0.000006	0.809414
$\phi\ (deg)$	-11.7071	5.917904	11.77308	5.785396	0.002469	19.89022

$\theta\ (deg)$	7.48828	6.067708	8.137141	5.164619	0.007361	17.57743
$p\ (deg/s)$	-1.26613	1.940041	1.886599	1.344104	0.002697	6.213617
$q\ (deg/s)$	-1.67052	2.062024	2.210278	1.468405	0.001065	7.728789
$V_g\ (m/s)$	-0.19031	0.806317	0.583808	0.587693	0.000074	2.723817

These results show that like all other method 2 simulations, the only states of concern are the estimates for the pitch and roll angles. Errors in the ground speed that arise from attitude errors are corrected at each GPS measurement update, and fluctuations in the estimated altitude are visually lost within the large change in altitude for this benchmark. However, there is a visible drift in the angular rate estimates.

So, to address the attitude errors seen here, it is important to note that as the angular rate bias drifts over time, the estimates for the pitch and bank angles do not grow unboundedly. The maximum pitch error seen is about 20 degrees and the maximum bank angle error seen is about 18 degrees. That said, the plot below shows how the yaw angle error does in fact grow unboundedly over the course of the simulation.

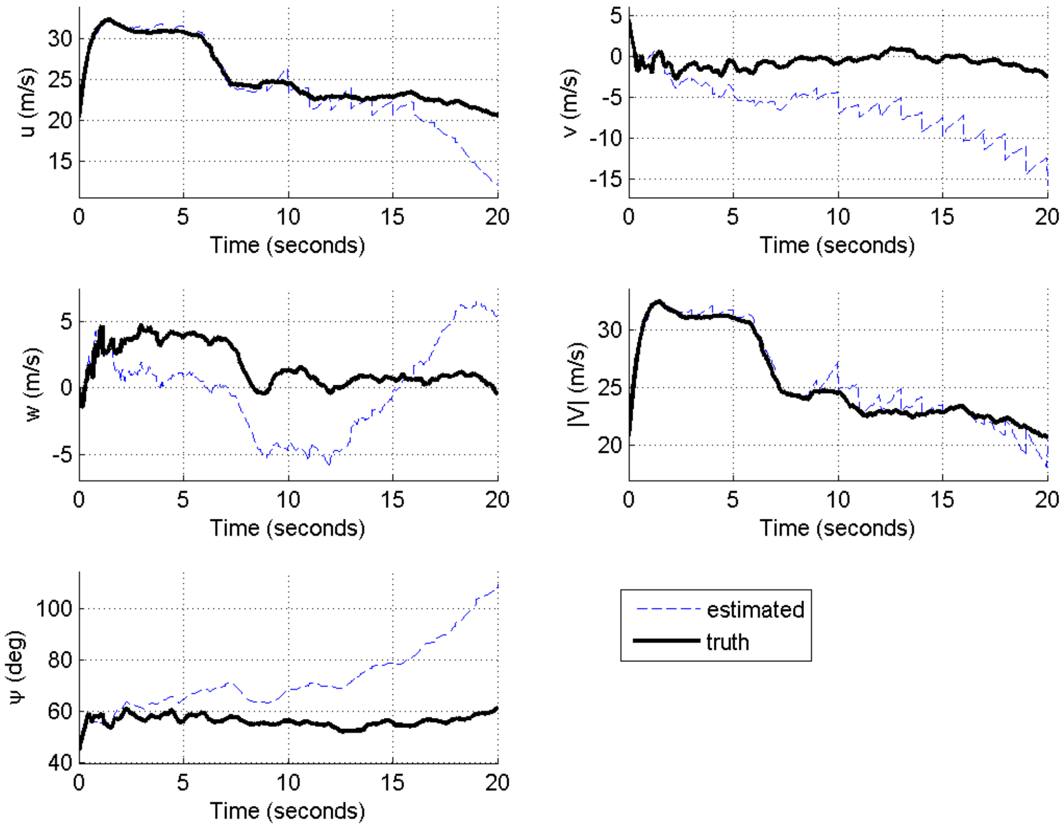


Figure 6.32: Extra State Estimation Results for Method 2 in Benchmark 3 and Dryden Wind

In order to assess the attitude angle observability, the estimates for the gyro bias over time needs to be presented. These estimates are below.

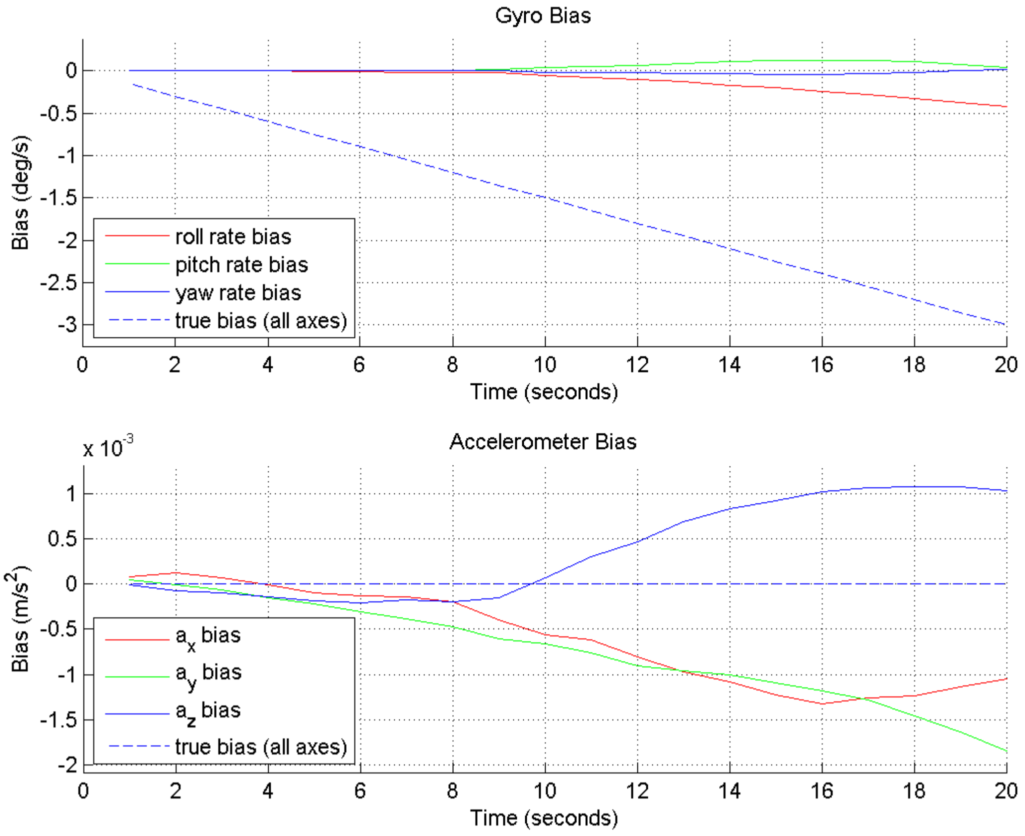


Figure 6.33: Bias Estimation Results for Method 2 in Benchmark 3 and Dryden Wind

These results show that the estimated gyro bias stays close to zero over the simulation. This makes sense because like in benchmark 3 in zero wind, even though all attitude and gyro bias states are able to be related to the velocity errors, there are still at most 2 observable modes. In benchmark 3 in zero wind, this mode was shown to be about the yaw angle. This appears to also be the case here, but it should be noticed that and increased variation in the gyro bias drift is seen here, when compared to benchmark 3 in zero wind. Like benchmark 2 in wind, benchmark 3 in wind experiences a jerk vector that varies in magnitude and direction. As such, the gyro bias states can be all be seen to begin to change towards the simulation. Unfortunately, like benchmark 3 in zero wind demonstrates, the gyro bias drift rate is still expected to be zero by the system model, and thus the estimates for the gyro bias require large errors in attitude before they begins to change their values. Due to this, the results do not really indicate which gyro rate biases are more observable than the others. In benchmark 3 in zero wind, it was shown with decent certainty that the roll and pitch rate biases were more observable, but in wind, since the unobservable modes drifts some, a confident determination of which axes are more observable cannot really be made in such a short simulation.

Now, even though the gyro bias estimates are close to zero and cannot be well-characterized, the observability of the attitude solutions can be. The solutions for the pitch and bank angles do not grow unboundedly. This indicates that the pitch and bank angles are being constantly corrected and that the pitch and bank angle states are observable. In benchmark 3 in zero wind, where even though the attitude errors were shown to be dependent on all axes of acceleration, the yaw angle proved to be the most unobservable angle because with a climb angle of 30 degrees, most of the weight of all of the measured accelerations were placed on updating the roll and pitch estimates. The same notion applies here for benchmark 3 in wind, but one might expect a small increase of observability to the yaw angle error relative to the other angles due to the increase in variation of the jerk vector. Unfortunately, however, such an increase isn't really seen, as the yaw angle estimate errors both with and without wind grow to between 30 and 35 degrees over the 20 second simulation.

Last, now that the relative observability of the different attitude angle errors and gyro bias axes has been addressed, it can be noticed that there really isn't an increase in error in the attitude solution for benchmark 3 when wind was added. Significant increases in attitude error magnitude were seen when wind was added to both benchmarks 1 and 2. However, the magnitude of the error in bank angle is nearly identical for benchmark 3 with and without wind, and the error in the pitch angle for benchmark 3 is a little bit better with wind than without wind. To quantify that error, below are tables that represent the errors introduced from the IMU propagation for benchmark 3 both with wind and without wind.

	$\boldsymbol{\Delta\phi}$ (degrees)	$\boldsymbol{\Delta\theta}$ (degrees)	$\boldsymbol{\Delta\psi}$ (degrees)	$\boldsymbol{\Delta\phi+\Delta\phi+\Delta\phi}$ (degrees)
Mean Error with True Attitude and Angular Rates $(\boldsymbol{\Sigma\overline{\Delta\psi_k}/n})$	0.033001	0.022855	0.03074	0.086596
Mean Error with True Attitude $\left(\boldsymbol{\Sigma\,\widetilde{\Delta\psi_k}\,/n}\right)$	0.040258	0.026719	0.041417	0.108395
Mean Estimate Error $(\boldsymbol{\Sigma\widehat{\Delta\psi_k}/n})$	0.038787	0.026053	0.045793	0.110634

	$\boldsymbol{\Delta\phi}$ (degrees)	$\boldsymbol{\Delta\theta}$ (degrees)	$\boldsymbol{\Delta\psi}$ (degrees)	$\boldsymbol{\Delta\phi+\Delta\phi+\Delta\phi}$ (degrees)
Mean Error with True Attitude and Angular Rates $(\boldsymbol{\Sigma\overline{\Delta\psi_k}/n})$	0.000077	0.000626	0.000144	0.000847
Mean Error with True Attitude $\left(\boldsymbol{\Sigma\,\widetilde{\Delta\psi_k}\,/n}\right)$	0.021064	0.018364	0.020979	0.060407
Mean Estimate Error $(\boldsymbol{\Sigma\widehat{\Delta\psi_k}/n})$	0.02352	0.017948	0.023682	0.06515

These table show that the magnitude of the error introduced by actual method 2 IMU propagation equation is 60% larger in roll and 45% larger in pitch. The only way that this error would be accounted for to result in a roughly equivalent magnitude of errors between the zero wind and wind scenarios is an increase in the observability of the pitch and roll angle error states. This makes sense, given that the yaw angle was just determined to be the unobservable mode and increased observability of the attitude errors can be expected with winds or maneuvers. That said, it follows that such an increased correction of the attitude angles was not seen between the Dryden and zero wind simulations of method 1 because in the method 1 simulation, the gyro rate biases were tracked very similarly between with and without wind. In the benchmark 3 simulations, there is an increased rate of bias drift, which results in larger errors in bias estimation, which in turn leads to a larger overall attitude error. This larger actual attitude error results in a larger attitude error EFK3 state estimate due to increased errors δu and δv, which are shown by the corrections to u and v in the plots above. In turn, this provides more attitude corrections, even with the same amount of covariance between velocity and attitude error that results from similar wind-based accelerations. Essentially, larger attitude errors did not provide an increased covariance and thus Kalman gain, but instead a larger velocity error state that was then applied to the attitude error estimates.

In conclusion, the benchmark 3 results between the Dryden wind and zero wind scenarios are very similar. In benchmark 3 with wind, additional observability of the pitch and yaw angle estimates is provided by wind-based accelerations. Further, in both benchmark 3 simulations, additional gyro bias drift was used. This extra bias drift caused larger attitude errors and a slightly faster change in the gyro bias estimates. An additional increase in the amount of attitude angle corrections was identified to be due to this increase in error.

6.2.4 Method 2 Analysis Summary

In conclusion, method 2 estimates the non-attitude states h, V_a, V_g, and y_{a_y}very well in all benchmarks. Airspeed is directly measured from the airspeed sensor and y_{a_y} is directly measured from the y-axis accelerometer. Both of these states are estimated identically well in method 1 because the same method is used there. The altitude and ground speed states, however, will suffer from errors incurred from integrating upon poor body frame inertial velocity component estimates, attitude estimates, and angular rate estimates. Fortunately though, altitude and ground speed are directly measurable from GPS information, and EKF3 places large weights these relationships, so the altitude and ground speed estimates are corrected back to very close to their true values at each GPS measurement update. The estimates for roll rate p and pitch rate q, which are just the respective gyro inputs added to EFK3's gyro bias states, track their true states well, but do deviate from their true values when EKF3 produces poor estimates for the gyro bias states.

The estimates for the attitude states ϕ and θ were by far the most interesting topic of discussion. Unlike method 1, which directly solves for these attitude angles, method 2 relies upon integration of a-priori attitude information. The method of this integration was found to be the cause of the vast majority of the increase in attitude error when Dryden wind was added to the simulations. Poor estimates of the gyro rate biases were seen to cause steady state errors in the attitude estimates, but their effects were less in the wind scenarios than they were in the zero wind scenarios. This reduction in the effect of poor gyro rate bias estimation on attitude estimation was shown to be due to increased amount of attitude corrections performed by EKF3 due to increased observability of the attitude angles in wind. It was shown that when the aircraft experiences varying accelerations on varying axes, the observability of both the attitude estimates and the gyro biases improved significantly. When that happened, the overall attitude error decreased significantly.

Last, even though wind increased attitude error and gyro bias observability, and subsequently the amount of corrections performed by EFK3, the magnitude of the attitude errors still grew substantially in wind. As mentioned, this was due to the increased integration error from equations (6.3) and (6.4). A future study could look into ways to improve this integration method to reduce the additional error incurred in dynamic windy conditions.

7 Method Comparison and Future Work

Table 7.1, below, is a numerical summary of the performance of the two methods of state estimation in the three different benchmarks under Dryden wind. The table contains the mean μ, standard deviation σ, and the maximum of the absolute value max_{abs} of the errors in state estimates. The results in the table below were all from the simulations that included gyro bias drift, and is simply a consolidation of the results presented separately in earlier sections.

Table 7.1: Method 1 and Method 2 in Dryden Wind State Estimate Error Statistics Comparison

Benchmark		**Method 1**			**Method 2**		
		μ	σ	max_{abs}	μ	σ	max_{abs}
1 (straight and level flight)	$h\ (m)$	0.009912	0.263184	1.017485	-2.29423	1.207648	5.014218
	$V_a\ (m/s)$	-0.00013	0.031237	0.280067	0.000185	0.032004	0.283372
	$\phi\ (deg)$	-2.48169	5.422106	26.76654	-8.46745	5.769247	21.90765
	$\theta\ (deg)$	-0.08141	6.245348	29.17006	-5.77646	8.164708	25.50788
	$p\ (deg/s)$	-1.54382	2.015224	7.948609	-0.99568	1.816456	6.935281
	$q\ (deg/s)$	-1.56719	1.967791	7.773666	-0.51144	1.796622	7.005333
	$V_g\ \ (m/s)$	0.422333	0.676247	3.267293	0.971821	0.891811	4.969819
	$y_{a_y}\ (m/s^2)$	-0.00368	0.107917	1.100337	-0.00237	0.121143	4.727883
2 (coordinated turn)	$h\ (m)$	0.008504	0.27609	1.04847	0.030457	2.323744	6.508412
	$V_a\ (m/s)$	0.000026	0.027756	0.261154	0.000179	0.026681	0.285108
	$\phi\ (deg)$	-3.41256	5.397888	25.33505	-1.79709	9.909461	22.06134
	$\theta\ (deg)$	-0.08103	6.011813	39.69093	3.551419	8.174121	25.6019
	$p\ (deg/s)$	-1.5443	1.881663	10.47874	-1.03095	1.851849	6.998946
	$q\ (deg/s)$	-1.56605	1.88061	10.2309	-1.28827	1.872867	7.484872
	$V_g\ \ (m/s)$	0.583431	0.783121	2.861224	0.351005	0.817978	4.469241
	$y_{a_y}\ (m/s^2)$	-0.00369	0.106615	1.447511	-0.00322	0.116953	4.23508
3 (wings level ascent)	$h\ (m)$	0.014131	0.273638	1.170085	0.828567	0.749619	1.607249
	$V_a\ (m/s)$	-2.70E-05	0.028042	0.263253	0.000444	0.071805	0.266219
	$\phi\ (deg)$	-3.71141	5.792228	26.56745	-11.7071	5.917904	22.51239
	$\theta\ (deg)$	-0.06719	6.169774	37.16324	7.48828	6.067708	14.74268
	$p\ (deg/s)$	-1.54406	2.017114	8.126008	-1.26613	1.940041	6.221454
	$q\ (deg/s)$	-1.56714	1.964096	7.632921	-1.67052	2.062024	7.200226
	$V_g\ \ (m/s)$	0.492612	0.750946	3.324215	-0.19031	0.806317	3.797198
	$y_{a_y}\ (m/s^2)$	-0.00369	0.106183	1.447511	-0.00794	0.24693	0.266386

This table supports the findings on the state estimation methods performance that were discussed separately in detail earlier in the simulation results sections.

First, both methods obtain airspeed from a low pass filter on an airspeed sensor and side specific force directly from the Y axis accelerometer, and no concerning issues were observed with the estimates with or without wind. The mean errors for these states are very close to zero. Since both methods use the same scheme for estimating these states, the performance in estimating these states was also identical.

Second, the errors in the estimates for the angular rates p and q are smaller for benchmarks 1 and 2 with method 2 than method 1. This is due to the tracking of the gyro biases that method 2 is able to do, whereas method 1 has no mechanism to track the gyro biases. Unfortunately though, the benchmark 3 estimate errors for the pitch and roll rates are nearly identical between methods 1 and 2. This is because in the benchmark 3 simulation was short in duration, and as discussed previously, the gyro bias estimates for method 2 only began to track the true biases at the very end of the simulation. This behavior of the method 2 gyro bias estimates could perhaps be improved in future work by adding gyro bias drift rate states to EKF3.

Third, both methods 2 and methods 1 perform similarly in terms of their estimates for the ground speed V_g and the altitude h. Both methods correct their estimates to close to the values provided by the GPS measurements at each GPS measurement update due to the low expected noise on the GPS measurements. However, both methods' estimates for V_g and h quickly degrade between GPS measurement updates whenever the attitude angles ϕ, θ, and ψ are estimated poorly. Method 1's estimates degrade less quickly than method 2's, which is seen in the error statistics in the above table, and this is due to method 1's assumption of level flight, whereas method 2 makes no assumption on the motion of the aircraft. Nonetheless, both methods' estimates degrade in wind due to poor estimates of the attitude angles between GPS measurement updates, and are corrected at each GPS measurement update.

Last, the estimates for the roll and pitch attitude angles ϕ and θ are poor for both method 1 and method 2 in wind. The table above shows that both methods produce fairly similar error statistics. However, it should be noted that method 1 produces large errors of short duration that are a direct results of acceleration experienced by the aircraft, whereas method 2 produces errors from integration of the angular rates that aren't quite as large in magnitude as method 1's errors, but persist for long periods of time as they are slowly corrected out by EKF3. In wind, method 2 has been shown to provide up to around 25 degrees of pitch angle error and 20 degrees of roll angle error. Method 1 was shown to provide up to 40 degrees of pitch angle error and 25 degrees of roll angle error.

Unfortunately, these results indicate that particular use of method 1 or method 2 for state estimation for autopilot control of small UAVs in wind cannot be recommended as they exist as described in this document. Both methods estimate the non-attitude states fairly well in a Dryden wind field, but the attitude estimates incur large errors. This poor attitude estimation performance of both estimation methods, and the effects of this poor attitude estimation on the states V_g and h, illustrate the need to augment or improve upon these standard methods to obtain accurate attitude estimates in wind. There is a multitude of ways this could be done, some of which are identified below.

7.1.1 Direct Attitude Sensing

The states that are necessary for autopilot control that are consistently estimated the poorest in wind throughout this work are the estimates for the attitude angles ϕ and θ. The attitude estimates from state estimation method 1 are poor due to the assumption that the time derivatives of the body frame components of the inertial velocity vector are zero, which was required in obtain an estimate. State estimation 2 estimates the attitude angles the poorest because of the integration errors associated with integrating on a priori data with current angular rates and lack of observability of the attitude angle errors.

Measuring at least some of the attitude angles directly would be one way to reduce these errors. Unfortunately though, direct attitude sensing can be difficult in implementation, depending on what mechanism to sense attitude is used. For example, magnetometers are subject to magnetic interference from the surrounding aircraft materials and horizon sensors are subject to visibility of the ground. Nonetheless, if attitude is successfully measured, it can be added to either method presented here. Most notably though, it can be easily added to method 2 by adding n attitude angle measurements as a third set of measurements to EKF3 as is shown below.

$$\boldsymbol{y}_{EKF3_{augmented}} = \boldsymbol{\epsilon}_{IMU-GPS,k} = \begin{bmatrix} {}^{N}\left[\boldsymbol{p}_{B/N,k}\right]_{IMU} \\ {}^{B}\left[\boldsymbol{V}_{B/N,k}\right]_{IMU} \\ \boldsymbol{\psi}_{IMU} \end{bmatrix} - \begin{bmatrix} {}^{N}\left[\boldsymbol{p}_{B/N,k}\right]_{GPS} \\ \boldsymbol{R}_{BN}\left(\psi_{k,IMU}\right)^{N}\left[\boldsymbol{V}_{B/N,k}\right]_{GPS} \\ \boldsymbol{\psi}_{Attitude\ Sensor} \end{bmatrix} \tag{7.1}$$

$$\frac{\partial \boldsymbol{H}_{EKF3_{augmented}}}{\partial \boldsymbol{x}_{EKF3_{augmented}}} = \left[\boldsymbol{I}_{6+nX6+n}\ \boldsymbol{0}_{6+nX9-n}\right] \tag{7.2}$$

However, it would also have to be recalled that the expected measurement noise $\boldsymbol{Q}_{EKF3_{augmented}}$ would have to have the expeted sensor noise of the attitude sensors added to it, like below.

$$\boldsymbol{Q}_{EKF3_{augmented}} = \boldsymbol{Q}_{EKF3} + \begin{bmatrix} 0_{6x6} & 0_{6x1} & 0_{6x1} & 0_{6x1} & 0_{6x6} \\ 0_{1x6} & \sigma_{\phi}^{2} & 0 & 0 & 0_{1x6} \\ 0_{1x6} & 0 & \sigma_{\theta}^{2} & 0 & 0_{1x6} \\ 0_{1x6} & 0 & 0 & \sigma_{\psi}^{2} & 0_{1x6} \\ 0_{6x6} & 0_{6x1} & 0_{6x1} & 0_{6x1} & 0_{6x6} \end{bmatrix} \tag{7.3}$$

Off-diagonal terms of this noise matrix may also be needed if the noise is coupled across the different attitude angle measurements.

7.1.2 Replace Integration Method in State Estimation Method 2

The method 2 results presented in this work show that the majority of the error that increase in attitude error seen in wind is due to the method of numerical integration. In zero wind, using true angular rates and attitude states with the same integration method resulted in nearly zero error, while doing so in wind showed that using true states reduces the amount of attitude error by at most 25%. In wind, the aircraft experiences higher frequency dynamics that this state estimation method was unable to account for, thus resulting in increased attitude estimation error. It is possible that a higher order integrator, such as a 4th order Runge-Kutta integrator, could introduce a much smaller attitude error across IMU updates and thus a smaller attitude estimate error. Other techniques, such as interpolating gyro and accelerometer sensor data between updates, may also yield a reduction in attitude estimate error.

7.1.3 Parallel Estimators

Another way to improve the attitude estimates could be to add assumptions on the flight of the aircraft when providing corrections to the IMU estimates of method 2. This could be done by essentially running a method 1 state estimator alongside of a method 2 state estimator. By running both, the best features of both methods could be taken advantage of. For example, because method 2 makes no assumptions on the flight dynamics of the aircraft, the error incurred in the attitude estimates only grows over long periods of time. Method 1 provides good attitude estimates when the time derivatives of the body frame components of inertial velocity vector are close to zero. Thus, when the IMU estimates those derivatives to be close to zero, per equation (6.4), or when the specific force is measured by the accelerometers to be close to the expected gravity vector, the method 1 estimates could be used as an attitude measurement in EFK3. Nonetheless, this is simply a suggestion and care would have to be taken to ensure that the IMU estimates are not corrected to poor estimates.

This list of possible improvements is by no means intended to be a complete list. Rather, it is a list of improvements that were immediately apparent from performing the work documented here, and is intended to be a starting point for readers interested in improving upon these state estimation methods.

8 Conclusion

This research has analyzed the performance of two commonly used methods of state estimation for autopilot control of small UAVs in windy conditions. It was first shown that the set of six states $h\ p\ q\ \phi\ \theta\ \ V_a\ V_g\ y_{a_y}$ is sufficient for tracking a commanded course rate, altitude, and airspeed in windy conditions. Both of the state estimation methods analyzed here estimate a superset of these states, but only the estimator's performance in estimating these states was heavily scrutinized.

The results of the simulations of these two methods of state estimation in wind show that use of either of these state estimation methods in wind cannot be recommended.   Instead, these results show that future work is necessary to enhance these estimation methods to achieve increased state estimate accuracy, especially for the roll and pitch angles ϕ and θ. Nonetheless, this work has fulfilled its purpose by contributing knowledge of exactly why the observed behavior of both methods in wind occurs. These findings are summarized below.

Both methods obtain airspeed from a low pass filter on an airspeed sensor and side specific force directly from the Y axis accelerometer, and no concerning issues were observed with the estimates with or without wind. Method 1, which solves directly for attitude using gyro and accelerometer inputs, suffered steady state bank angle errors when gyro bias drift was added to the simulation and during coordinated turns, and also suffered steady state pitch angle errors during climbs. These errors that were seen without wind were shown to be due to the assumptions made in the development of the method's attitude determination process. The ground speed estimate was also shown to degrade when gyro bias is present due to EKF2's inability to account for gyro bias drift. When wind was added to the simulation, the only difference was that the variations in the pitch and bank angle estimates became much larger in all benchmarks. This was shown to be due to the assumption that the body frame components of the inertial velocity vector are constant, which is certainly not the case in wind.

Method 2 integrates upon a-priori attitude and translation data to determine the current attitude and translation states, and then attempts to correct both the translation and the attitude estimates with GPS measurements. It was shown that while the ground speed and altitude errors are corrected well by EKF3 with GPS data in all benchmarks, attitude estimates were only corrected well when gyro bias or wind were not present. Both gyro bias and wind increased the error introduced into the attitude estimates by the IMU integration, but EFK3 was still able to keep the attitude solutions bounded in all cases. It was shown that the increase in error seen in wind was due to the integration of the angular rates performed from the IMU measurements. Future work could explore improving upon the 1st order integration method that was used to here to achieve smaller attitude error. However, without direct attitude measurements, method 2 still exhibits poor observability of both the attitude angles and sensor biases in benchmarks that don't involve wind or significant maneuvering. It was shown that there are always at most 2 observable attitude and gyro bias modes, and because in standard aircraft flight X and Y accelerations are minimized, the unobservable modes are about the Z axis. However, it was also show that by performing significant maneuvers or by being perturbed by a Dryden wind field, the unobservable mode can be dynamically changed and thus share observability amongst all axes and thus attitude angle errors.

Overall, this research has provided an in-depth understanding of the changes in performance of the two most common methods of state estimation for small UAVs when the aircraft is subjected to a Dryden wind field. This understanding will assist future research to improve upon small UAV state estimation methods so that the states necessary for autopilot control can be well estimated in wind.

9 References

[1] Honeywell Inc., "Avionics Zone," Spring 2003. [Online]. Available: http://www.honeywell.com/sites/docs/DZHZL77WSTHRHWKLQS9EKSY4C4A68QR9.pdf. [Accessed 22 July 2014].

[2] W. J. Pisano and D. Lawrence, "Control Limitations of Small Unmanned Aerial Vehicles in Turbulent Environments," American Institude of Aeronautics and Astronautics, Boulder, CO.

[3] J. D. Barton, "Fundamentals of Small Unmanned Aircraft Flight," *Johns Hopkins APL Technical Digest, Volume 31, Number 2,* pp. 132-149, 2012.

[4] R. W. Beard and T. W. McLain, Small Unmanned Aircraft: Theory and Practice, Princeton, New Jersey: Princeton Universtiy Press, 2012.

[5] A. M. Eldredge, "Improved State Estimation for Miniature Air Vehicles," Department of Mechanical Engineering, Brigham Young University, Provo, Utah, 2006.

[6] S. Gleason and G.-E. Demoz, GNSS Applications and Methods, Norwood, MA: Artech house, 2009.

[7] P. D. Groves, GNSS, Inertial and Multisensor Integrated Navigation Systems, Norwood, MA: Artech House, 2008.

[8] P. Axelrad and D. Lawrence, "ASEN 3200 Attitude Representations," University of Colorado at Boulder, Boulder, CO, 2006.

[9] R. Stengel, "Aircraft Equations of Motion - 2," Princeton University, Princeton, New Jersey, 2012.

[10] F. A. P. Lie and D. Gebre-Egziabher, "Synthetic Air Data System," *Journal of Aircraft,* 2012.

[11] J. W. Langelaan, N. Alley and J. Neidhoefer, "Wind Field Estimation for Small Unmanned Aerial Vehicles," in *AIAA guidance, Navigation, and Control Conference*, Toronto, Canada, 2010.

[12] W. Premerlani, "IMU Wind Estimation (Theory)," 2009. [Online]. Available: http://gentlenav.googlecode.com/files/WindEstimation.pdf. [Accessed 6 November 2013].

[13] E. Frew, "Alternative Autopilot Design Approaches," University of Colorado at Boulder, Boulder, CO, 2013.

[14] N. O. a. A. Administration, "Beaufort Wind Scale," [Online]. Available: http://www.spc.noaa.gov/faq/tornado/beaufort.html. [Accessed 2014].

[15] D. P. Poorman, "Exploring Flying Qualities through Simulation of a Small Unmanned Aircraft Autopilot on a F-16," University of Colorado at Boulder, Boulder, CO, 2012.

[16] A. D. Wu, E. N. Johnson, M. Kaess, F. Dellaert and G. Chowdhary, "Autonomous Flight in GPS-Denied Environments Using Monocular Vision and Inertial Sensors," American Institute of Aeronautics and Astronautics.

[17] A. Bry, A. Bachrach and N. Roy, "State Estimation for Aggresive Flight in GPS-Denied Environments Using Onboard Sensing," in *IIEEE International Conference on Robotics and Automation*, 2012.

[18] H. Schaub and J. L. Junkins, "Analytical Mechanics of Space Systems, Second Edition," American Institude of Aeronautics and Astronautics, Reston, Virginia, 2009.

[19] J. Elston, "Overview and Comparison of Wind Field Estimation in the Atmospheric Boundary Layer Using Small Unmanned Aircraft," University of Colorado at Boulder, Boulder, CO, 2013.

[20] R. Rogers, Applied Mathematics in Integrated Navigation Systems, Third Edition, Books24x7. Web. Oct. 30, 2013, 2007.

[21] B. Douglas, "Robotic Car - How to read Gyro Datasheets," Control System Laboratories, Snohomish, WA, 2013.

[22] I. Rhee, M. F. Abdel-Hafez and J. L. Speyer, "Observability of an Integrated GPS/INS During MAneuvers," *IEEE Transactions on Aerospace and Electronic Systems,* vol. 40, no. 2, pp. 526-534, 2004.

[23] W. J. Pisano and D. A. Lawrence, "Autonomous Gust Insensitive Airaft," American Institute of Aeronautics and Astronautics, 2008.