Received: 2 October 2020 | Revised: 10 May 2021 | Accepted: 6 July 2021

DOI: 10.1002/navi.440

ORIGINAL ARTICLE

WILEY ION

Air data fault detection and isolation for small UAS using integrity monitoring framework

Kerry Sun | **Demoz Gebre-Egziabher**

Department of Aerospace Engineering and Mechanics, University of Minnesota - Twin Cities, Minneapolis, MN 55455, United States

Correspondence
Kerry Sun, Department of Aerospace Engineering and Mechanics, University of Minnesota - Twin Cities, Minneapolis, MN.
Email: sunx0486@umn.edu

Funding information
Minnesota Invasive Terrestrial Plants and Pests Center (MITPPC)

Abstract

A Fault Detection and Isolation (FDI) algorithm is developed to protect against Water-Blockage (WB) pitot tube failure in the safety-critical Air Data System (ADS) used on small Unmanned Aircraft Systems (UAS). The algorithm utilizes two identical Synthetic Air Data Systems (SADS) as the basis for state estimation. Each SADS works independently with a pitot tube while sharing an IMU and GNSS receiver. The fault detection is designed using the integrity monitoring framework, and the isolation is obtained via independent fault detection channels. The ADS requirements are established, and the WB failure mode is analyzed based on real faulty air data. A new residual-based test statistic is introduced, and the link among the test statistic, observability matrix, and Minimal Detectable Error (MDE) are examined. Finally, a flight data set with a known water-blockage fault signature is used to assess the algorithm's performance in terms of the air data protection levels and alert limits.

KEYWORDS
fault detection and isolation, integrity monitoring, synthetic air data system, UAS

1 | INTRODUCTION

A reliable Air Data System (ADS) plays a vital role in an aircraft's safety and performance. While ADS provides the measurement of various parameters, airspeed V_a, angle-of-attack α, and angle-of-sideslip β are the main parameters that define the flight envelope. As shown in Figure 1, airspeed is the speed of an aircraft relative to the air, and angle-of-attack and angle-of-sideslip are the flow angles relative to the aircraft. Air data is usually measured onboard by accurate air data sensors such as the pitot-static tube and the angle vane. Also, reliability analyses such as Fault Tree Analysis (FTA) and Failure Mode and Effect Analysis (FMEA) are often used to help identify fault modes and certify the redundant hardware systems.

The analytical redundancy approach is particularly useful for small Unmanned Aircraft Systems (UAS) due to the Size, Weight, And Power (SWAP) requirements. However, there is a lack of analytical methods to certify analytical redundancy. As emerging technologies such as Urban Air Mobility (UAM) (Vascik et al., 2018), or UAS operations either in the Line-of-Sight (LOS) or Beyond Visual Line-Of-Sight (BVLOS) (Cour-Harbo, 2017; Fang et al., 2018;

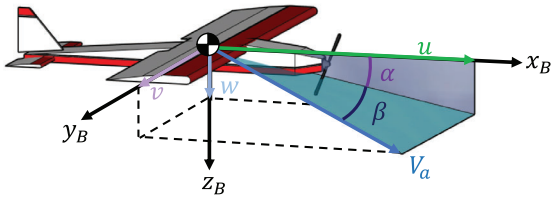


FIGURE 1 Illustration of air data triplet: airspeed V_a, angle-of-attack α, and angle-of-sideslip β [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

Johnson et al., 2017; McCrink & Gregory, 2018; Yapp et al., 2018) mature, the need for rigorous and certifiable analytical redundancy methods will increase. In these applications, ADS is one of the safety-critical subsystems which needs to be certifiable and meet safety requirements.

Most of small UAS usually have one or two pitot tubes as the only sensors in its ADS. For example, the recent Part-135 certification process requires any unmanned air carriers (i.e., package delivery) to have at least one heated pitot tube (Federal Aviation Administration, 2019). However, many small UAS cannot afford to have a heated pitot tube onboard due to its cost and power requirements. Without the heating system, the low-cost pitot tubes on many small Unmanned Aerial Vehicles (UAVs) are prone to Water-Blockage (WB) faults. This is why many small UAV operations, such as agricultural surveying and construction inspection, cannot be carried out reliably during the rainy days. In Figure 2, a typical inexpensive pitot tube [10 US dollars to 20 US dollars (JDrones, 2020; Eagle Tree Systems, 2020)] is shown. It can be seen that the pitot tube is connected to a transducer via plastic tubes. The setup is simple and used by many UAVs but prone to the WB faults since there is no built-in drainage or heating system in the pitot tube. Water can enter the pitot tube on flights during foggy or rainy days, which fully or partially block the stagnation ports and affect the transducer's pressure readings.

To improve the safety and reliability of the ADS and minimize the number of redundant and multiple sensors, one approach being considered is called a Synthetic Air Data System (SADS) (Lie & Gebre-Egziabher, 2013; Sun et al., 2019b). A SADS is an estimator that calculates air data quantities using non-air data sensors such as the GNSS, IMU, magnetometer, and mathematical model of the aircraft.

SADS is a form of analytical redundancy that can help detect and deal with faults in the traditional ADS of small UAS. SADS can also potentially be coupled with one or two air data sensors to resolve the low-reliability issue. In fact, SADS has already been implemented in some commercial aircraft such as the Boeing 787 (Austrailian Transport Safety Bureau, 2015). The use of SADS is also being considered by many other aircraft designs at this time (Federal Democratic Republic of Ethiopia, Ministry of Transport, Aircraft Accident Investigation Bureau, 2020; Komite Nasional Keselamatan Transportasi, Republic of Indonesia, 2018; SeekingAlpha, 2019).

In what follows, we give a brief overview of the prior work on air data Fault Detection and Isolation (FDI) in the literature. We also explain why the existing methods are not adequate to certify ADS on small UAS and why the Integrity Monitoring (IM) framework can potentially solve this problem.

1.1 | Prior work

Air data FDI using advanced control and estimation algorithms has renewed interest over the last decade due to the recent advancements in the safety-critical UAV applications. These air data FDI techniques can be roughly separated into three categories: model-based, model-free, and data-driven algorithms.

The model-based algorithms typically leverage the dynamic model of the aircraft (Freeman et al., 2013; Hansen & Blanke, 2014; Ossmann et al., 2017). For example, Freeman et al. (2013) designed an airspeed fault detection algorithm using the aerodynamic model of the aircraft as well as linear robust H_∞ filters to detect faults, reject disturbance, and provide robustness to the modeling errors.

The model-free algorithms mainly rely on the sensor information and the kinematic models of the vehicles (Eubank et al., 2010; Guo et al., 2018; Lu et al., 2016; Van Eykeren & Chu, 2014). The model-free algorithms also often use Kalman Filter (KF)-based estimation techniques and the innovation χ^2 test to determine faults. An illustrative example is shown in Lu et al. (2016). They use an adaptive three-step unscented KF to detect and isolate air data faults.

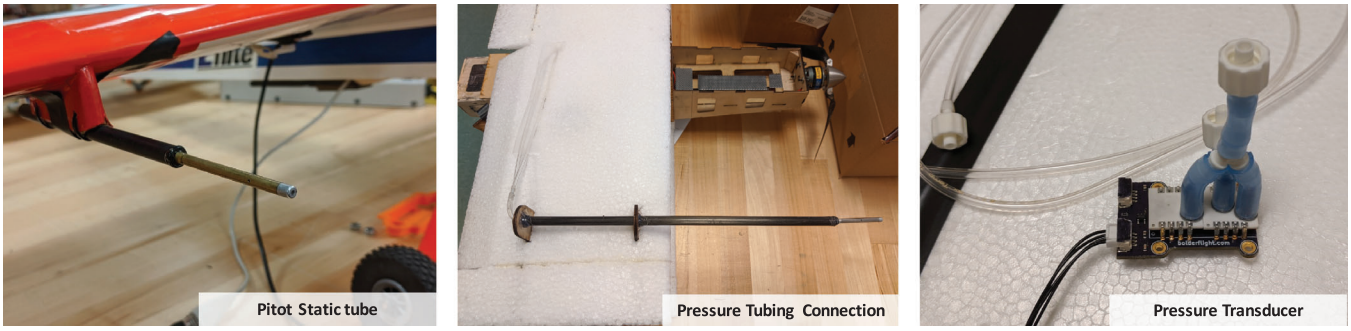


FIGURE 2 Entire ADS using a pitot-static tube, pressure tubes, and a pressure transducer for a small UAS [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

The data-driven algorithms primarily rely on large amounts of data to develop reliable input-output methods (e.g., autoregressive models, neural networks). These models are then used to detect faults through inconsistency check (Borup, 2018; Fravolini et al., 2017, 2018; Rohloff et al., 1999).

These air data FDI algorithms are primarily concerned with how to detect faults accurately, however, they do not provide a framework for ensuring the reliability of the air data fault detection algorithms from a requirement point of view. That is, they do not address the following question: *Can we design an air data FDI algorithm to satisfy a given set of system requirements such as integrity and continuity, and provide statistical protection levels for the air data estimates?*

Recent work (Freeman, 2014; Hu & Seiler, 2015; Kotikalpudi et al., 2020) has made progress towards certification of analytically redundant systems via reliability analysis. In the work here, we borrow tools often used in the field of integrated GNSS navigation to help design air data fault detection algorithms to ensure reliability.

One of the standard techniques for fault detection of the safety-critical aerospace navigation systems is called Receiver Autonomous Integrity Monitoring (RAIM) (Brown & Chin, 1998; Lee, 1986; Parkinson & Axelrad, 1988; Sturza, 1988). RAIM methods are used for safety-critical applications such as GNSS-based precision landing systems for aircraft (Khanafseh et al., 2014; Tanıl et al., 2017a, 2017b; Walter et al., 2008) and have been the subject of a significant amount of work for the last two decades. RAIM methods have also been used recently to provide Integrity Monitoring (IM) for other navigation systems such as the Simultaneous Localization and Mapping (SLAM) problem (Arana et al., 2019a, 2019b; Bhamidipati & Gao, 2019).

The basic idea of RAIM is to leverage redundant measurements at every time step [see snapshot detection scheme (Brown & Chin, 1998; Parkinson & Axelrad, 1988; Sturza, 1988)] or sequentially (Joerger & Pervan, 2013) to come up with probabilistic measures to detect faults and provide statistical bound to protect the state estimate. The advantage of this approach is that it provides the means for rigorous integrity risk computation. It uses redundant measurements to achieve fault detection capability and quantify the impact of undetected faults on state estimation errors. Another advantage is that the calculation of the threshold is based on probability, not selective tuning. Unlike RAIM, in many of the aforementioned works, it is often seen that a particular threshold is handpicked for a given application without rigorous probabilistic calculation.

However, the rigorous IM framework has not always been implemented on emerging non-PNT applications. This is partly due to the sensor measurements' inhomogeneity or the non-linearity of dynamics in many applications such as UAS. Many systems, such as ADS, have limited redundant and heterogeneous measurements at every time step. This limitation sometimes makes the snapshot of residual-based detection function infeasible to determine faults. And while linearization errors are usually small in the GNSS applications, measurement models in other systems are generally highly nonlinear, and the linearization errors' can be large. Therefore, the non-linearity might have a significant effect on the existing IM techniques. Lastly, many systems have observability issues [unobservable states (Kassas & Humphreys, 2014) or conditionally observable states (Sun et al., 2019b)], and this is usually overlooked when dealing with IM in GNSS applications.

Another aspect of fault detection in IM is the selection of an appropriate fault detector or test statistic. The goal of a fault detector is to detect fault quickly without raising too many false alarms. For real-time applications, online fault detectors are preferred. There are many online fault detectors such as the simple residual thresholding, KF innovation χ^2 test, least-squares residual-based χ^2 test [or commonly referred as the snapshot RAIM (Brown & Chin, 1998; Parkinson & Axelrad, 1988; Sturza, 1988) in navigation literature], Sequential Probability Ratio (SPRT), Cumulative Sum (CUSUM) test, and Generalized Likelihood Ratio (GLR) test.

Residual thresholding is the most straightforward test as it only requires a threshold to determine whether the data has exceeded the nominal level. The KF innovation χ^2 test is suitable for any KF-based state estimation, and the least-squares residual-based χ^2 test is a good choice when redundant measurements are available. The three methods mentioned above deal with linear or quadratic functions of the residual.

On the other hand, both SPRT and CUSUM tests are well-known for their nonlinear stopping rules (Gustafsson, 2000). For example, the standard one-sided SPRT test requires three tuning parameters: drift ν, threshold h, and reset level a. The basic idea of a one-sided SPRT test is to test whether the test statistics have drifted away significantly from the threshold. The drift parameter ν is used to subtract from the test statistic to control the drift's level, and the parameter a is used to prevent negative drift.

Similarly, the one-sided CUSUM test is the same as the one-sided SPRT test with the reset level $a = 0$. Several variations of both SPRT and CUSUM tests can be found in the literature. However, both tests require hand-tuning for the desired outcome. The GLR is also a powerful nonlinear test for fault detection, but it usually requires the knowledge of Probability Density Functions (PDF) under different hypotheses. In this paper, the KF-based detectors will be utilized with some additional novel improvement.

1.2 | Contribution

This paper provides four main contributions to the air data FDI literature: First, we design a dual pitot tube air data fault detection and isolation system that can be easily implemented on most UAVs. Second, we expand on sequential IM techniques in the Kalman filter setting to evaluate the integrity risk for the designed fault detection algorithm.

Specifically, we show how to deal with the limited redundant measurement problem and establish an analytical relationship among the residual-based test statistic, the Linear Time-Varying (LTV) observability matrix, and the Minimum Detectable Error (MDE). We also show how monitoring the observability of the system can potentially help rule out false alarms. Furthermore, we generalize the IM performance trade-off design procedure so that we can use it to evaluate other pitot tube failure modes.

Third, we also show how to establish alert limits and protection level bounds for the angle-of-attack and sideslip states. Lastly, we demonstrate our algorithm's capability using a recorded flight data in which a known WB pitot tube failure occurred.

1.3 | Paper organization

The remainder of this paper is organized as follows. Section 2 presents a brief description of the model-free SADS estimator used for the dual pitot tube air data system design developed in this paper. Section 3 presents the air data system requirements needed for the fault algorithm design. Section 4 describes the WB failure mode used in this work. Section 5 presents the fault detection design and analysis, which includes the derivation of the residual-based test statistic and its relation to the observability matrix, the MDE design and analysis, and the IM performance trade-off design procedure. Section 6 derives the alert limits and protection level calculations of angle-of-attack α and sideslip β. Section 7 presents the flight results and its associated detection performance. Concluding remarks and future outlook are given in Section 8.

2 | DUAL PITOT TUBE AIR DATA SYSTEM DESIGN

2.1 | System architecture

For the development that follows, we propose a dual pitot tube air data system for small UAS. The architecture consists of two identical SADS estimates of α and β by fusing airspeed measurements from the pitot tube with information from an IMU and a GNSS receiver (Sun et al., 2018, 2019a, 2019b).

Each SADS utilizes its own pitot tube, but both SADS share a GNSS receiver and IMU. Sharing the GNSS receiver and IMU reduces cost and software complexity. The design can be easily expanded to architecture with dual GNSS receivers and IMU units.

Each SADS can detect faults independently (i.e., identify and isolate the fault source) and is designed to satisfy the given system performance requirements (i.e., integrity and continuity requirements). The two SADS together provide recovery capability for a single faulty pitot tube failure via simple decision logic. This ADS system also provides accurate estimates and protection levels for the synthetic angle-of-attack α and sideslip β. The entire dual ADS design is illustrated graphically in Figure 3.

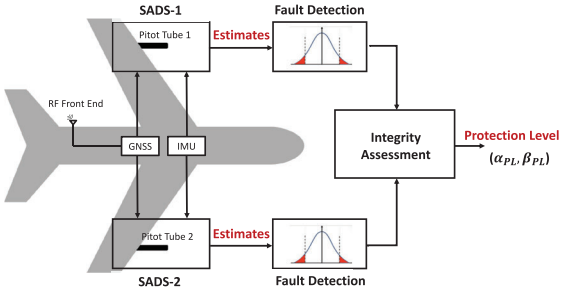


FIGURE 3 Dual air data fault detection design [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

In comparison to the commonly used triple-redundancy ADS design (Yeh, 1996), the dual pitot tube fault-tolerant ADS has a unique advantage: it only includes two small and inexpensive pitot tubes by leveraging a so-called dynamic redundancy approach (Isermann, 2005), which can be easily installed on the UAVs. In the case of a single pitot tube fault, the dual ADS can shut off the faulty pitot tube and continue its nominal operation using the secondary pitot tube (sometimes referred to as a *hot standby*).

Additionally, the two independent SADS filters can be implemented asynchronously on the hardware. It is a more fault-tolerant design choice in comparison to other filter methods [e.g., solution separation method (Joerger et al., 2014)], which in this case would use both pitot tube measurements simultaneously. Note that failure of a single pitot tube would lead to potential loss of control even if multiple filters (e.g., a bank of KF filters for various pitot tube failure modes) were used. Though simultaneous failure of both pitot tubes could occur under the same rainy condition, we considered that simultaneous-failure case beyond the scope of this paper and can be considered in future work. Table 1 summarizes the decision logic for the dual pitot tube ADS fault detection design.

TABLE 1 Dual Pitot tube ADS decision logic for each scenario

	SAD-1	SAD-2	Decision
Scenario 1	Nominal	Nominal	Nominal operation
Scenario 2	Nominal	Faulty	Use SAD-1 and raise alarm
Scenario 3	Faulty	Nominal	Use SAD-2 and raise alarm
Scenario 4	Faulty	Faulty	Terminate mission and land

Since SADS is the one of key components in the proposed algorithm, we will briefly go over some details to facilitate the understanding in the next subsection. For more details, please refer to Sun et al. (2019b).

2.2 | Synthetic Air Data System (SADS)

The synthetic SADS estimator is an extension of the 15-state, loosely-coupled INS/GNSS EKF (Gleason & Gebre-Egizabher, 2009), which blends information from an IMU and GNSS receiver. The INS/GNSS filter's state vector is augmented by three additional states representing the components of the wind velocity vector. Therefore, the SADS filter states, expressed in the error state vector $\delta\mathbf{x} \in \mathbb{R}^{18\times1}$, is given by:

$$\delta\mathbf{x} = \left[\delta\mathbf{p}^T \ \delta\mathbf{v}^{nT} \ \delta\boldsymbol{\psi}_{nb}^{nT} \ \delta\mathbf{b}_a^T \ \delta\mathbf{b}_g^T \ \delta\mathbf{W}^{nT}\right]^T \tag{1}$$

where $\delta\mathbf{p} = [\delta L \ \delta\lambda \ \delta h]^T$ is the position error vector in latitude, longitude, and altitude, $\delta\mathbf{v}^n = [\delta V_N \ \delta V_E \ \delta V_D]^T$ is velocity error vector resolved in the North-East-Down (NED) frame, denoted by the superscript n. The vector $\delta\boldsymbol{\psi}_{nb}^n = [\delta\phi \ \delta\theta \ \delta\psi]^T$ represents the attitude errors which are defined to be the small rotation angles between the actual NED frame and the estimated NED frame. The subscript nb indicates the positive direction is defined as being from the NED frame (n-frame) to the body frame (b-frame). The vectors $\delta\mathbf{b}_a = [\delta b_{ax} \ \delta b_{ay} \ \delta b_{az}]^T$ and $\delta\mathbf{b}_g = [\delta b_{gx} \ \delta b_{gy} \ \delta b_{gz}]^T$ are the accelerometer and rate gyro triad output bias error vectors, respectively. Finally, $\delta\mathbf{W}^n = [\delta W_N \ \delta W_E \ \delta W_D]^T$ is the error in the wind velocity vector resolved in the NED frame.

The synthetic SADS estimator synthesizes an estimate of α and β without using the α and β sensor measurements. The synthetic estimates of α and β are calculated using the EKF state estimates as follows:

$$\alpha = \tan^{-1}\left(\frac{u}{v}\right), \quad \beta = \sin^{-1}\left(\frac{v}{\sqrt{u^2+v^2+w^2}}\right) \tag{2}$$

where:

$$\begin{bmatrix} u & v & w \end{bmatrix}^T = \mathbf{C}_n^b(\boldsymbol{\psi}_{nb}^n)[\mathbf{v}^n - \mathbf{W}^n] \tag{3}$$

The $\mathbf{C}_n^b(\boldsymbol{\psi}_{nb}^n)$ is the coordinate transformation from NED to the body frame. The measurement vector $\mathbf{z} \in \mathbb{R}^{7\times1}$, shown in Equation (4), consists of position $\mathbf{p}$ and velocity $\mathbf{v}^n$ estimates from the GNSS receiver, along with the true airspeed V_a estimate determined using the pressure measurements from the pitot tube.

$$\mathbf{z}_k = \left[\mathbf{p}^T \ \mathbf{v}^{nT} \ V_a\right]_k^T \tag{4}$$

The time and covariance update equations for this filter are, for the most part, identical to those of the filter described in Gleason and Gebre-Egizabher (2009). What is new is the dynamic model for the augmented states (wind) and the measurement model.

Similar to the modeling of the accelerometer and gyroscope biases in the filter, the dynamics of the wind are modeled as a first-order Gauss-Markov model, motivated by Berman and Powell (1998). The details of the Gauss-Markov model for the wind and sensors can be found in (Berman & Powell, 1998) and (Xing, 2010), respectively. However, for the sake of completeness we re-state the process noise matrix $\mathbf{R}_w$ that accounts for accelerometer, gyroscope, and wind, respectively, for a quick reference:

$$\mathbf{R}_w = diag\left(\left[\sigma^2_{\mathbf{w}_a} \ \sigma^2_{\mathbf{w}_g} \ 2\sigma^2_{\mathbf{w}_{ad}}/\tau_{ad} \ 2\sigma^2_{\mathbf{w}_{gd}}/\tau_{gd} \ 2\sigma^2_{\mathbf{w}_{Wd}}/\tau_{Wd}\right]\right) \in \mathbb{R}^{15\times15} \tag{5}$$

where $\sigma^2_{\mathbf{w}_a}$ and $\sigma^2_{\mathbf{w}_g}$ are the accelerometer and gyroscope *white noise* variances, respectively. The parameter $\sigma^2_{\mathbf{w}_{ad}}$, $\sigma^2_{\mathbf{w}_{gd}}$, and $\sigma^2_{\mathbf{w}_{Wd}}$ are the accelerometer, gyroscope, and wind *random walk* variances, respectively. The τ_{ad}, τ_{gd}, and τ_{Wd} are the associated time constants defined in the first-order Markov process.

The linearized measurement model used by the EKF is $\delta\mathbf{z}_k = \mathbf{H}_k\delta\mathbf{x}_k + \boldsymbol{\upsilon}_k$, where $\boldsymbol{\upsilon}_k$, the measurement noise vector, is assumed to follow a normal distribution with zero mean and covariance $\mathbf{R}$, denoted as $N(\mathbf{0}, \mathbf{R})$. The measurement Jacobian $\mathbf{H}_k \in \mathbb{R}^{7\times18}$ is given by:

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_{3\times9} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_{3\times9} & \mathbf{0}_3 \\ \mathbf{0}_1 & H_{\mathbf{v}^n} & \mathbf{0}_{1\times9} & H_{\mathbf{W}^n} \end{bmatrix} \tag{6}$$

where the first two block rows map the EKF states into the GNSS position and velocity measurement errors, and the last block row maps the EKF states into the airspeed measurement error. The matrix $H_{\mathbf{v}^n}$ is derived from linearizing the nonlinear airspeed measurement model

$V_a = ||\mathbf{v}^n - \mathbf{W}^n||_2 + \upsilon_{V_a}$. The matrix $H_{\mathbf{W}^n}$ is similarly derived and happens to be equal to $-H_{\mathbf{v}^n}$. The matrix $H_{\mathbf{v}^n}$ is shown in the following:

$$H_{\mathbf{v}^n} = \frac{1}{V_a}\begin{bmatrix} V_N - W_N & V_E - W_E & V_D - W_D \end{bmatrix} \tag{7}$$

The associated measurement noise covariance $\mathbf{R}$ is shown as follows:

$$\mathbf{R} = diag\left(\begin{bmatrix} \sigma^2_{P_N} & \sigma^2_{P_E} & \sigma^2_{P_D} & \sigma^2_{V_N} & \sigma^2_{V_E} & \sigma^2_{V_D} & \sigma^2_{V_a} \end{bmatrix}\right) \in \mathbb{R}^{7\times 7} \tag{8}$$

where the diagonal of $\mathbf{R}$ contains the position, velocity, and airspeed noise variances.

2.3 | Observability consideration

An advantage of this SADS estimator is that it does not use the aircraft dynamic model, and it provides synthetic α and β estimates as well as their covariances. Specifically, unlike the model-based SADS, which uses the aerodynamic model of the aircraft and six degree-of-freedom dynamic equations, this mode-free SADS estimator only relies on the kinematic equation and sensor measurements. However, this estimator is conditionally observable as analyzed in detail in Sun et al.'s earlier work (2019b). Briefly, ensuring observability of this estimator requires the following two conditions (i.e., conditionally observable):

1. The airplane must be accelerating so that the INS/GNSS heading and gyro bias states become observable (Gleason & Gebre-Egizabher, 2009)
2. The wind vector $\mathbf{W}^n$ must be *quasi-static*. The term *quasi-static* means that the variations in $\mathbf{W}^n$ are assumed to be negligibly small over a small time window whose size is defined in (Sun et al., 2019b).

The second condition is required to ensure that changing airspeed and the wind states $\mathbf{W}^n$ separately observable (i.e., the wind triangle relationship). Furthermore, the quality of estimates, in part, depends on the degree of observability.

The degree of observability is determined quantitatively by analyzing the condition number of observability Gramian in Sun et al. (2019b). Since the synthetic estimate is conditionally observable, the ability to detect air data system faults is also conditional. One of this paper's key contributions is to show how observability is related to the fault test statistic, which is explained and demonstrated in Sections 5 and 7, respectively.

3 | AIR DATA SYSTEM REQUIREMENT

To quantify the air data fault detection performance, we start with given system requirements, such as integrity risk I_{req} and continuity risk C_{req}. The integrity risk is the probability that a hazardous fault goes undetected, and continuity risk is the probability that an alarm is issued about the presence of a fault when in fact there is no fault. Mathematically, they are defined as (Pervan, 1996):

$$I_{req} \triangleq P_{MD} + \sum_{DF} P(MI|DF)P_{DF} \tag{9}$$

$$C_{req} \triangleq P_{FA} + \sum_{DF} P(NI|DF)P_{DF} \tag{10}$$

where P is shorthand for *probability of*, and MD, FA, DF, MI, and NI stand for missed detection, false alarm, detected failure, missed-identified failure, and non-isolable failure, respectively.

Since the primary focus here is to deal with fault detection against pitot tube faults only, the second terms on the right-hand side of Equations (9) and (10) are ignored. These terms are associated with the isolation problem, which deals with all possible fault modes associated with the pitot tube.

In our case, we assume the pitot tube only experiences one particular failure mode, so the fault detection and isolation are essentially achieved at the same time. We also rely on the simple decision logic (Table 1) to recover from the faulty pitot tube experiencing the WB failure. Therefore, we limit our scope to the following performance requirement:

$$\begin{aligned} I_{req} &\approx P_{MD} \\ &= P(\text{Pitot tube fault not sensed} \\ &\quad |\text{Pitot tube has failed due to WB}) \end{aligned} \tag{11}$$

$$\begin{aligned} C_{req} &\approx P_{FA} \\ &= P(\text{issuing an alarm}|\text{no pitot tube failure}) \end{aligned} \tag{12}$$

As stated in Equations (11) and (12), the probability of missed detection P_{MD} represents the probability of not detecting a pitot tube failure given the pitot tube has indeed failed. Similarly, the probability of false alarm P_{FA} is the probability of issuing an alarm when there is no pitot tube failure. Also, if the pitot tube is working properly, this hypothesis is denoted as H_0. If the pitot tube is not working correctly, this hypothesis is denoted as H_1. The details of the particular WB pitot tube fault mode is

analyzed in Section 4. Also, since currently there is no universally accepted the numerical values of I_{req} and C_{req} for the WB pitot tube failure to our knowledge, we treat them as the trade-off variables in the Section 5.

4 | WATER BLOCKAGE FAILURE MODE

As mentioned in Section 1, low-cost pitot tubes such as the one shown in Figure 2 are susceptible to a failure model called Water Blockage (WB) during foggy or rainy days. This is a failure mode where the water particles in the air can enter through the front of the pitot tube and accumulate, leading to a reduction in total pressure either slowly or abruptly, as illustrated in Figure 4.

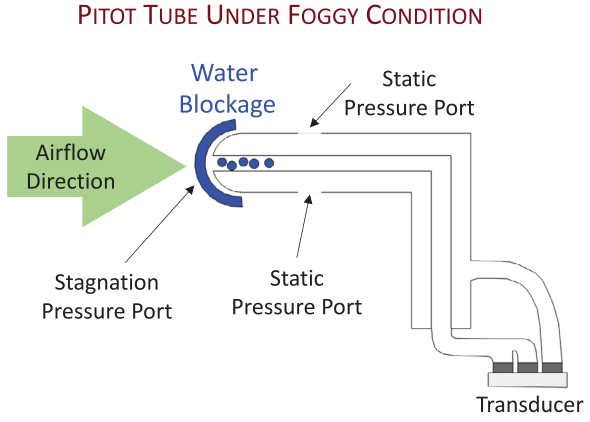


FIGURE 4 Illustration of the pitot static tube experiencing WB fault scenario [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

The airspeed V_a is typically calculated based on Bernoulli's principle as follows[1]:

$$V_a = \sqrt{\frac{2(P_t - P_s)}{\rho}} = \sqrt{\frac{2\Delta P}{\rho}} \tag{13}$$

where P_t is the stagnation or total pressure, P_s is the static pressure, and ρ is the air density. A partially blocked pitot tube would affect ΔP, which often leads to an airspeed drop. The size of the airspeed drop can vary significantly based on how much water is clogging the pitot tube.

Figure 5 shows a time history of two different faulty airspeed data sets from two different UAVs. The first faulty airspeed data (the top figure in Figure 5) comes from an agricultural inspection experiment. The flight data was collected by Sentera LLC.

The UAV took off around 570 secs, but the airspeed quickly decreased at 614 secs due to the WB faulty pitot

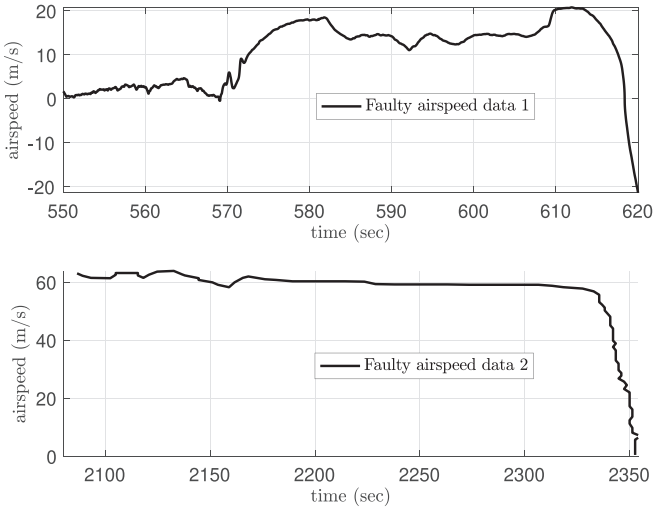


FIGURE 5 Two faulty airspeed data sets due to WB

tube. The other faulty airspeed data shown here[2] is reproduced from Hansen & Blanke (2014). The airspeed experiences a sharp drop at 2335 secs due to the water-clogged pitot tube.

Even though two different UAVs operate at different flight conditions (i.e., the nominal airspeed from the second set is almost three times higher than the first one), both pitot tubes experience a similar pressure drop rate. Those two airspeed drops' slope is estimated to be 2.5 m/s^2 and 3 m/s^2, respectively. Although this profile could be different from different pitot tubes, we will use them as the fault profile from which we need protection for the illustration in this paper.

Ideally, a larger faulty airspeed sample size would be required to represent the WB faulty pitot tube fault characteristics. However, it is difficult to obtain faulty airspeed data due to the WB failure in flight since: 1) the precise occurrence (i.e., timestamp) of the WB fault is usually unknown, and 2) faulty airspeed data is sensitive information and generally not shared in the public domain. In fact, to the best of our knowledge, this is the first paper that utilizes more than one set of faulty airspeed data due to the WB failure mode.

The airspeed measurement model under the faulty condition shown in Equation (14) is used for the SADS estimator. The WB fault mode is modeled as an unknown linear ramp fault, denoted as f_{V_a}, and the nominal airspeed is calculated by taking the euclidean norm (2-norm) of the difference between the inertial velocity $\mathbf{v}^n$ and wind vector $\mathbf{W}^n$ in the navigation frame. The airspeed measurement noise υ_{V_a} is modeled as the white Gaussian noise. The noise variance is typically unknown after the

[1] The temperature and altitude effect are not considered in this study

[2] The second airspeed data shown here is digitally extracted from the paper for illustration

fault occurs. Here we assume the noise variance stays the same before and after the fault:

$$V_a = ||\mathbf{v}^n - \mathbf{W}^n||_2 + f_{V_a} + \upsilon_{V_a} \tag{14}$$

Also, we assume the fault component f_{V_a} affects the nominal airspeed measurement continuously after water clogs in the tube. In other words, the water is assumed to stay in and continue clogging the pitot tube. A timeline for the fault scenario is described in Figure 6.

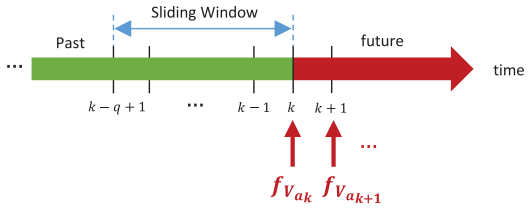


FIGURE 6 Fault scenario: the green portion of the timeline is fault-free section and the red is the faulted case [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

The fault vector $f_{V_{a_k}}$ starts entering at the time k and persists for future times. The test statistics developed in Section 5 uses a sliding window of size q to detect faults. This sliding window moves forward into the red region as time continues. Note that the sliding window would start right away when the measurement update of the KF filter reaches enough measurement for its detector. Hence the fault will most likely fall into the sliding window since the water blockage fault usually happens after the takeoff.

The methodology determining the minimum detectable faulty component f_{V_a} for the fault detection design is presented in Section 5. The minimum detectable airspeed fault depends on various factors, such as the measurement sampling rate T_s, the sliding window size q of the fault test statistic, the integrity risk P_{MD}, and continuity risk requirements P_{FA}. An IM trade-off design procedure is also presented to show how various factors can affect the design choice based on a given set of requirements.

5 | FAULT DETECTION DESIGN AND ANALYSIS

In this section, we first discuss the choice of Kalman filter based test statistics. Specifically, in addition to the conventional innovation-based KF test statistic, we introduce a sequential residual-based test statistic. We show how this test statistic is related to the observability matrix. Second, we discuss how the test statistics are generally designed to meet the integrity requirements.

The Minimum Detector Error (MDE) metric is used to link the integrity requirements and the specific air data system requirements. The MDE metric is determined through the Detector Operating Characteristic (DOC) curves. We also provide a general design procedure to determine the acceptable MDE and detection time τ for different failure modes. Lastly, we examine the quality of the proposed KF residual-based test statistic through the MDE analysis.

5.1 | Kalman filter based test statistics

Since the fault detection design in this paper relies on the Kalman filter based estimation, both innovation and least-squares residual-based χ^2 tests are considered for the fault detector design. Kalman filter based test statistics are widely used in the field of GNSS applications.

5.1.1 | Innovation-based test statistic

The most popular method is called the normalized innovation squared χ^2 test (Bar-Shalom et al., 2002). It uses the innovation vector and its covariance to form a test statistic, which follows a central χ^2 distribution with Df $= mq$ under the fault-free hypothesis H_0 as shown below:

$$D_{\gamma,k}|H_0 = \sum_{j=k-q+1}^{k} \boldsymbol{\gamma}_j^T \mathbf{S}_j^{-1} \boldsymbol{\gamma}_j \sim \chi^2(mq) \tag{15}$$

where $D_{\gamma,k}$ is test statistic at the time step k, $\boldsymbol{\gamma}_k$ is the innovation vector calculated using $\boldsymbol{\gamma}_k = \boldsymbol{z}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_k^-)$, the matrix $\boldsymbol{S}_k$ is the innovation covariance calculated from $\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{R}_k$, m the number of measurements at the time step k, and q is the sliding window size. The vector $\hat{\boldsymbol{x}}_k^-$ is the predicted estimate, and $\boldsymbol{P}_k^-$ is the covariance of the KF prediction.

A suitable threshold for this test can be computed by using the inverse chi-square cdf $F_{\chi^2}^{-1}$ given the desired probability of false alarm and the appropriate Df as follows:

$$T_\gamma = F_{\chi^2}^{-1}(1 - P_{FA}, \text{Df}) = F_{\chi^2}^{-1}(1 - P_{FA}, mq) \tag{16}$$

The innovation-based test statistic's effectiveness depends on the quality of the KF-predicted estimates (linear prediction), KF measurements (sensor quality), and length of q. However, one weakness of the innovation-based test statistic is that it cannot be analyzed easily if there is a fault. This is because all the information embedded in the innovation vector and its covariance, and the contribution from a fault cannot be parsed out analytically. In other words, it would be more beneficial

if we could find a test statistic that is both analyzable and informative.

5.1.2 | Residual-based test statistic

The other common fault detector for KF-based estimation is the least-squares residual-based χ^2 test. However, the residual-based χ^2 test is not applicable to the SADS considered here because there is no redundant airspeed measurement at every time step; past measurements will be required for the fault detection design instead of using the well-established snapshot RAIM method. To overcome this issue, we formulate a KF residual-based test in the following:

$$D_{\mathbf{r},k}|H_0 = \mathbf{r}_{k-q:k}^T \boldsymbol{\Sigma}^{-1} \mathbf{r}_{k-q:k} \sim \chi^2(mq-n) \tag{17}$$

where n is the number of the states in the KF and $\mathbf{r}_{k-q:k}$ is stacked residual vector from the past time step $k-q$ to the current time step k, denoted as $\mathbf{r}_{k-q:k} = \begin{bmatrix} \mathbf{r}_{\mathbf{k-q}}^{\mathbf{T}} & \mathbf{r}_{k-q+1}^T & \cdots & \mathbf{r}_k^T \end{bmatrix}^T$. Each residual $\mathbf{r}_k$ is computed from the difference between the measurements and a posteriori estimate $\boldsymbol{x}_k^+$ using $\mathbf{r}_k = \boldsymbol{z}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_k^+)$. The matrix $\boldsymbol{\Sigma}$ is the covariance matrix of the weighted least residual vector $\mathbf{r}_{k-q:k}$ is shown in Equation (18):

$$\boldsymbol{\Sigma} = \mathbf{I}_{q\times q} \otimes \mathbf{R} + \mathbf{Q}_{w,k-q:k} (\mathbf{I}_{q\times q} \otimes \mathbf{R}_w) \mathbf{Q}_{w,k-q:k}^T \tag{18}$$

where $\otimes$ is the Kronecker tensor product and $\mathbf{R}_w$ is the process noise matrix. The matrix $\mathbf{Q}_{w,k-q:k}^T$ is realized through the batch linear system realization shown in Appendix A. The sliding window residual-based test statistic is used here instead of a one-time step residual test because the number of measurements m is less than the number of states n for the SADS.

In other words, the popular snapshot RAIM method from GNSS does not work here since no redundant measurements are available at each time step. The χ^2 test requires $\mathrm{Df} = mq - n > 0$, therefore the threshold for the residual-based test statistic is calculated as follows:

$$T_{\mathbf{r}} = F_{\chi^2}^{-1}(1-P_{FA}, \mathrm{Df}) = F_{\chi^2}^{-1}(1-P_{FA}, mq-n) \tag{19}$$

The sequential residual-based χ^2 fault detection test statistic has similar properties to the snapshot RAIM method. Also, we make a connection between this residual-based test statistic and the LTV observability matrix. This is done by connecting a window of measurement to a past state vector $\boldsymbol{x}_{k-q}$ using weighted least squares. The state vector $\boldsymbol{x}_{k-q}$ can be estimated by applying weighted linear least squares to the batch linear system shown in Equation (A2):

$$\hat{\boldsymbol{x}}_{k-q} = \boldsymbol{\mathcal{O}}_{k-q:k}^* \mathbf{Z}_{k-q:k} = \hat{\boldsymbol{x}}_{k-q}^+ \tag{20}$$

where $\hat{\boldsymbol{x}}_{k-q}^+$ signifies the posteriori estimate from the past measurement from time step $k-q$ to the current measurement k. The matrix $\boldsymbol{\mathcal{O}}_{k-q:k}^*$ is calculated as:

$$\boldsymbol{\mathcal{O}}_{k-q:k}^* = (\boldsymbol{\mathcal{O}}_{k-q:k}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mathcal{O}}_{k-q:k})^{-1} \boldsymbol{\mathcal{O}}_{k-q:k}^T \boldsymbol{\Sigma}^{-1} \tag{21}$$

where $\mathcal{O}_{k-q:k}$ is the LTV observability matrix shown in Equation (A7). The residual vector $\mathbf{r}_{k-q:k}$ can be subsequently expressed in the following:

$$\mathbf{r}_{k-q:k} = \begin{bmatrix} \mathbf{r}_{\mathbf{k-q}} \\ \mathbf{r}_{k-q+1} \\ \vdots \\ \mathbf{r}_k \end{bmatrix} = (\mathbf{I}_{mq\times mq} - \boldsymbol{\mathcal{O}}_{k-q:k} \boldsymbol{\mathcal{O}}_{k-q:k}^*) \mathbf{Z}_{k-q:k} \tag{22}$$

Under the fault-free H_0 hypothesis, we can also write $\mathbf{r}_{k-q:k}$ as follows:

$$\mathbf{r}_{k-q:k} = (\mathbf{I}_{mq\times mq} - \boldsymbol{\mathcal{O}}_{k-q:k} \boldsymbol{\mathcal{O}}_{k-q:k}^*) \left[\mathbf{Q}_{w,k-q:k} \mathbf{W}_{k-q:k} + \mathbf{V}_{k-q:k} \right] \tag{23}$$

where $\mathbf{r}_{k-q:k}$ follows a normal distribution $N(\mathbf{0}, \boldsymbol{\Sigma})$, and $\mathbf{W}_{k-q:k}$ and $\mathbf{V}_{k-q:k}$ are defined in Equation (A4) and (A5) respectively.

Using Equation (22), we can also write $D_{\mathbf{r},k}$ as follows:

$$\begin{aligned} D_{\mathbf{r},k} &= \mathbf{r}_{k-q:k}^T \boldsymbol{\Sigma}^{-1} \mathbf{r}_{k-q:k} \\ &= \mathbf{Z}^T (\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)^T \boldsymbol{\Sigma}^{-1} (\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*) \mathbf{Z} \\ &= \mathbf{Z}^T \boldsymbol{\Sigma}^{-1} (\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*) \mathbf{Z} \end{aligned} \tag{24}$$

where the subscripts $mq \times mq$ and $k-q:k$ are dropped to shorten the notation. The last equality of Equation (24) is obtained because both $\boldsymbol{\Sigma}^{-1}$ and $\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)$ are symmetric, and the matrix $\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*$ is idempotent. The formal proof of this matrix equality is given in Appendix B.

The mathematical revelation in Equation (24) shows that the KF residual-based test statistic is a function of the observability matrix. Furthermore, the matrix $\boldsymbol{\mathcal{O}}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mathcal{O}}$ inside of $\boldsymbol{\mathcal{O}}^*$ is the discrete weighted observability Gramian or the Fisher information matrix (Bar-Shalom et al., 2002).

This test statistic has a distinct advantage: it gives users a tool to analyze how the system's observability affects the test statistic $D_{\mathbf{r},k}$ given a sliding window of the

measurement from time step k-q to k. By analyzing the observability Gramian, we can tell how well the current system is observable. We can make useful statements between the effect of the test statistic and the motion of the vehicle (indirectly represented by the observability matrix). Furthermore, it can also be used to eliminate false alarms, as demonstrated in Section 7. Hence, we also call this test statistic the observability-based test statistic.

5.1.3 | Limitation of snapshot RAIM test statistic

It is worth noting that the test statistic in Equation (24) is different from the well-known RAIM-like $\sum_{j=k-q+1}^{k} \mathbf{r}_j^T \mathbf{R}_j^{-1} \mathbf{r}_j = \sum_{j=k-q+1}^{k} \boldsymbol{z}_j^T \mathbf{R}_j^{-1} (I - \mathbf{H}_j \mathbf{H}_j^*) \boldsymbol{z}_j$ test statistic, which also follows a central χ^2 with degrees-of-freedom $mq - n$ under H_0 hypothesis.

This test statistic does not account for the process noise from the time update step in the KF prediction step. It loses all the dynamic information between the KF measurement update. Again, though the snapshot $\mathbf{r}_k^T \mathbf{R}_k^{-1} \mathbf{r}_k$ test statistic is often used for GNSS integrity monitoring, it is inapplicable when dealing with the system considered here which does not have redundancy measurements at each time step.

5.2 | Minimum Detectable Error (MDE) design

Before we proceed with determining the Minimum Detector Error (MDE), we will discuss some concepts and define some terms that will be used later. Any fault detection algorithm's goal is to detect credible faults before they lead a hazardous situation (e.g., loss of the aircraft, collision with terrain).

For a given UAV, the stall angle of attack α_{stall} and the minimum airspeed at which the airplane can fly $V_{a,stall}$ (stall speed) are synonymous. We will assume we are dealing with an electrically powered UAV, so its mass does not change during flight. Thus, the flight detection algorithm we design will have to detect faults before the estimated airspeed falls below $V_{a,stall}$.

Since the UAV's operating speed V_a is generally not constant during a given flight, the allowed drop in the estimate of airspeed (before the airplane is outside of the safe-flight envelope) is not constant either. To simplify the design, we use the average operating speed $\bar{V}_a$ as an approximation. We will call the difference between average operating airspeed $\bar{V}_a$ and $V_{a,stall}$ the airspeed Allowable Error, or AE for short.

Given a WB-fault profile, we can determine the time required for the airspeed estimate to drop below the stall speed. We call this τ_{max}, and the fault detection algorithm must detect a WB fault in a time shorter than τ_{max}.

Finally, we call the smallest airspeed estimation error that can be detected consistently (quantified by the missed detection and false alarm rate probabilities) the MDE. The fault detection algorithm ensures that MDE< AE and raises the alarm when the detection time less than τ_{max} after the onset of a pitot tube failure.

5.2.1 | MDE and τ_{max} determination

By examining the slope of the faulty airspeed data in Figure 5, it is determined that the minimum faulty airspeed drop rate is about 2.5 m/s^2. For the particular UAV used in the flight experiment, the average operating airspeed is about 17.5 m/s, and the stall speed is about 10 m/s. The difference 7.5 m/s between $\bar{V}_a$ and $V_{a,stall}$ translates to the maximum detection time $\tau_{max} = 3s$ as follows:

$$\tau_{max} = \frac{\text{AE}}{2.5} = \frac{\bar{V}_a - V_{a,stall}}{2.5} = \frac{17.5 - 10}{2.5} = 3s \tag{25}$$

If the fault detection algorithm fails to detect the fault before the fault exceeds AE, then the algorithm is ineffective against the allowable fault. Mathematically, the AE should satisfy the following:

$$\text{MDE}(P_{FA}, P_{MD}) \leq \text{AE}(\bar{V}_a, V_{a,stall}, \tau_{max}) \tag{26}$$

where the lower bound MDE is a function of P_{FA} and P_{MD}. The AE = 7.5 m/s is reasonable, but a tight bound since the UAV should be able to recover even if the airspeed drops below the stall speed as long as there is a sufficient altitude for recovery. Hence, a larger AE can be found based on the average operating altitude. Nevertheless, we will use 7.5 m/s as the upper bound for AE in the following analysis.

5.2.2 | MDE and sampling rate

We model the airspeed fault as a linear ramp fault using the minimum drop rate of 2.5 m/s^2. The fault detector should still catch any rate that is higher 2.5 m/s^2 since a larger fault would result in a quicker detection.

Since χ^2 based tests are used for the fault detection, one of the requirements for χ^2 test is that the degree-of-freedom (Df) has to be greater than zero. For example, the least-squares residual-based fault detection method (Brown & Chin, 1998) requires the number of the measurement m to be greater than the state n. Because we

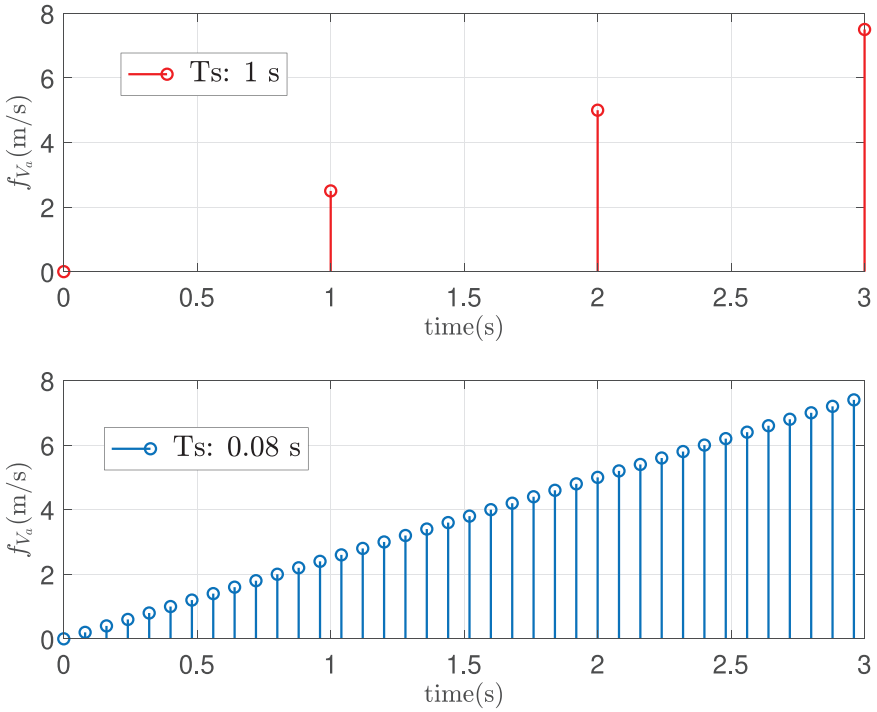


FIGURE 7 Simulated airspeed linear ramp fault profile at two different sampling rates [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

are doing sequential measurement update for the airspeed (i.e., $m = 1$ at each time step k) and we only assume the fault comes from the pitot tube, the minimum number of airspeed measurements needed for the residual-based χ^2 test is 19 since the number of states is 18.

Therefore, we need to collect at least 19 airspeed measurements (i.e., sliding window size $q = 19$) to detect fault within the maximum allowable detection time. Sampling too fast would also degrade the KF's performance because measurements closely spaced in time would cause the innovation vector to be highly correlated with itself, which violates the uncorrelatedness assumption.

Figure 7 shows the simulated airspeed linear ramp fault profiles at 1 s and 0.08 s sampling time over 3 s using the 2.5 m/s airspeed drop rate. The sampling time $T_s = 1$ s might be too slow for detection as the number of faulted measurement is too small for detection test statistic to respond before the time exceeds τ_{max}.

On the other hand, the sampling time $T_s = 0.08$ s is sufficient even if we only use the measurements after the fault occurs to form the test statistic[3]. Therefore, an appropriate sampling time should be chosen for the airspeed measurement based on the AE and the measurement uncorrelatedness assumption.

[3] Since a sliding window of measurement is used, the test statistics always has enough nominal measurements for χ^2 test before the fault occurs. If the faulty measurement enters the sliding window too slowly, the test statistics can be ineffective

For the small UAV we are dealing with here, the sampling time $T_s = 0.08$ s is found to be effective for good estimation and detection performance.

5.2.3 | MDE and detector operating characteristic curves

A Detector Operating Characteristic (DOC) curve (Sturza, 1988) is a graphical plot that illustrates the power of a discrimination threshold given various fault modes. It is used to understand the trade-off between the false alarm and missed detection rate, and the effectiveness of the designed threshold.

The DOC curves are obtained by plotting various P_{MD} and P_{FA} at different fault sizes. The DOC is the similar to the Receiver Operating Characteristic (ROC) curve, except the y-axis of ROC is the probability of detection P_D ($P_D + P_{MD} = 1$). Mathematically, the DOC curve is calculated using Equation (27):

$$P_{MD} = F_{\chi^2(\lambda)}(T, \mathrm{Df}) = F_{\chi^2(\lambda)}\left(F^{-1}_{\chi^2}(1 - P_{FA}, \mathrm{Df}), \mathrm{Df}\right) \tag{27}$$

where T is the designed threshold and determined by P_{FA}. The functions F_{χ^2} and $F_{\chi^2(\lambda)}$ are central and non-central χ^2 Cumulative Distribution Functions (CDF), respectively, with Df degrees-of-freedom. The non-centrality parameter λ represents the sum of the non-zero means. In this case,

it is sum of the biases vectors f_{V_a} over the sliding window q, as shown below:

$$\lambda = \sum_{j=k-q+1}^{k} \frac{f_{V_{a_j}}^2}{\sigma_{V_{a_j}}^2} \tag{28}$$

The expression for λ here is greatly simplified due to the sequential measurement update procedure. In the next subsection, we will generalize λ expression for dealing with various inhomogeneous measurements under the standard KF measurement update setting.

We use measurements over a short period for λ instead of a single measurement. Not only does this satisfy the Df requirement as mentioned earlier, but also a single faulty measurement may be ineffective. For example, if we wait until the fault grows to 7.5 m/s at 3 seconds, and use this measurement to form the test statistic (e.g., the innovation-based χ^2 test statistic only requires one measurement at minimum), then it would be too late in issuing the alarm. Hence we accumulate measurements over a short period and use them to form the detection test statistic.

For a fixed size sliding window q, λ can be calculated using a sliding window of the normalized $f_{V_{a_j}}^2/\sigma_{V_{a_j}}^2$ from the past time step $k-q+1$ to the current time step k. We define two parameters $\overline{\text{MDE}}$ and MDE in Equations (29) and (30):

$$\overline{\text{MDE}} \triangleq \sqrt{\lambda} = \sqrt{\sum_{j=k-q+1}^{k} \frac{f_{V_{a_j}}^2}{\sigma_{V_{a_j}}^2}} \tag{29}$$

$$\begin{aligned} \text{MDE} &\triangleq max\left(f_{V_{a_{k-q+1}}}, f_{V_{a_{k-q+2}}}, \ldots f_{V_{a_{k-1}}}, f_{V_{a_k}}\right) = f_{V_{a_k}} \\ &= f_{V_{a_{max}}} \end{aligned} \tag{30}$$

where $\overline{\text{MDE}}$ is the square root of the sum of normalized fault vectors and MDE the represents the magnitude of the largest fault in the sliding window. The definition of MDE is compatible with the one mentioned in Equation (26) because it represents the maximum tolerable fault size in the sliding window.

Figure 8 shows $\overline{\text{MDE}}$ using a window size $q = 19$ and a constant $\sigma_{V_a} = 1.75$ m/s. Each stem represents the sum of the past 19 normalized faulty measurements at sampled time k, where each faulty measurement is simulated based on the linear ramp fault profile shown in the bottom subfigure of Figure 7. The $\overline{\text{MDE}}$ represents the sum of the past normalized faulty measurements.

Figure 9 shows the DOC contour plot for a single degree-of-freedom: $\text{Df} = mq - n = 1 \times 19 - 18 = 1$. The plot is created by sweeping different λ over a wide range of P_{FA} and P_{MD}. The contour color is represented by the largest

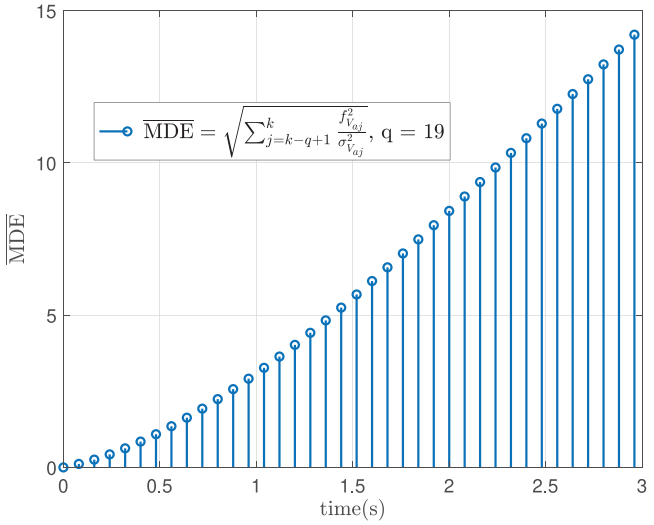


FIGURE 8 $\overline{\text{MDE}}$ over 3 secs period using the linear ramp fault profile in Figure 7 [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

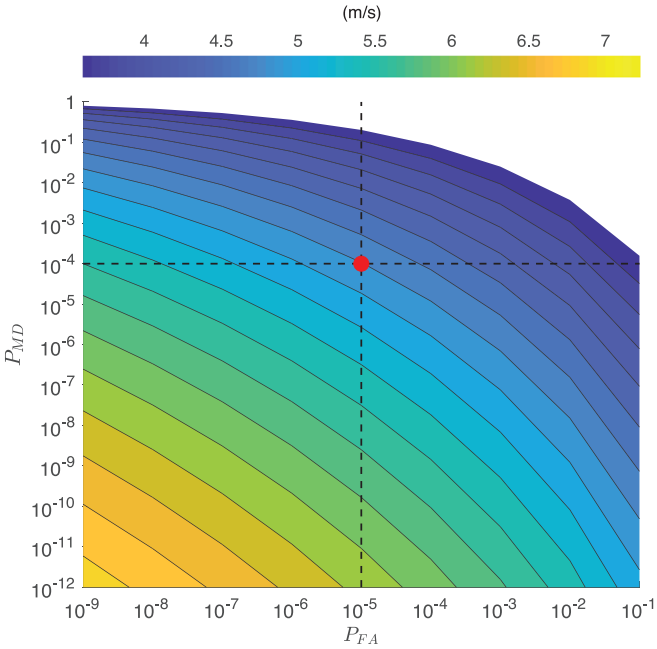


FIGURE 9 DOC curves where the color represents the associated MDE [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

minimum detectable airspeed error $f_{V_{a_{max}}}$ in the sliding window.

If the red dot represents a design choice $(P_{FA}, P_{MD}) = (10^{-5}, 10^{-4})$, the minimum detectable error requirement MDE is 5.0 m/s, which satisfies the inequality in Equation (26). In other words, the fault detection algorithm can detect the airspeed fault f_{V_a} after it reaches 5.0 m/s, which corresponds to a detection time $\tau = 2$ s starting from the beginning of the fault profile shown in the

TABLE 2 The sliding window $\overline{\text{MDE}}$ (m/s) and its associated MDE (m/s) in the square bracket for Df = 1

P_{FA}/P_{MD}	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}
10^{-1}	2.93 [3.6]	3.97 [3.6]	4.74 [3.6]	5.36 [3.8]	5.91 [4.0]	6.40 [4.2]	6.84 [4.4]	7.26 [4.6]	7.64 [4.8]
10^{-2}	3.86 [3.6]	4.90 [3.6]	5.67 [3.8]	6.29 [4.2]	6.84 [4.4]	7.33 [4.6]	7.78 [4.8]	8.19 [5.0]	8.57 [5.2]
10^{-3}	4.57 [3.6]	5.61 [3.8]	6.38 [4.2]	7.01 [4.4]	7.56 [4.8]	8.04 [5.0]	8.49 [5.2]	8.90 [5.4]	9.29 [5.4]
10^{-4}	5.17 [3.6]	6.22 [4.2]	6.98 [4.4]	7.61 [4.8]	8.16 [5.0]	8.64 [5.2]	9.10 [5.4]	9.51 [5.6]	9.89 [5.8]
10^{-5}	5.70 [4.0]	6.74 [4.4]	7.51 [4.8]	8.14 [5.0]	8.68 [5.2]	9.17 [5.4]	9.62 [5.6]	10.03 [5.8]	10.41 [6.0]
10^{-6}	6.17 [4.2]	7.22 [4.6]	7.98 [5.0]	8.61 [5.2]	9.16 [5.4]	9.65 [5.6]	10.10 [5.8]	10.51 [6.0]	10.89 [6.2]
10^{-7}	6.61 [4.4]	7.75 [4.8]	8.42 [5.0]	9.04 [5.4]	9.59 [5.6]	10.08 [5.8]	10.53 [6.0]	10.94 [6.2]	11.32 [6.4]
10^{-8}	7.01 [4.4]	8.06 [5.0]	8.82 [5.2]	9.45 [5.6]	9.99 [5.8]	10.48 [6.0]	10.93 [6.2]	11.34 [6.4]	11.73 [6.4]
10^{-9}	7.39 [4.6]	8.44 [5.2]	9.20 [5.4]	9.83 [5.6]	10.37 [6.0]	10.86 [6.2]	11.31 [6.4]	11.72 [6.4]	12.11 [6.6]

bottom sub-figure in Figure 7. Both of those numbers satisfy the given constraint as follows:

$$\begin{aligned} \text{MDE} = 5m/s \leq 7.5m/s = \text{AE} \\ \tau = 2s \leq 3s = \tau_{max} \end{aligned} \tag{31}$$

Of course, this is an ideal situation given the χ^2 test statistic is assumed to come from a perfect zero mean, unit variance white sequences. Nevertheless, the analysis provides a systematic way of assessing the fault detection capability for a realistic ramp airspeed fault profile.

Table 2 shows both $\overline{\text{MDE}}$ and MDE values over range of P_{FA} and P_{MD}. The numbers in the square bracket are the corresponding MDE values. It is seen that the MDE decreases as P_{FA} and P_{MD} increase and vice versa. This trend is correct and intuitive because a more stringent integrity and continuity requirement would enforce the detection function to catch a fault reliably at a larger MDE since a larger airspeed fault would trigger the detection function to cross the threshold more easily.

The final sliding window size was chosen to be 19 for both detectors. The sensitivity analysis is done for different sliding window sizes, and it is observed that increasing q from the minimum window size (i.e., 19) does not significantly change the DOC curves. Also, too big of a window size would make the test statistic function sluggish because it tends not to respond slower than the latest change in the measurement.

5.2.4 | Trade-off analysis procedure

The above subsections complete the determination of the MDE design based on the integrity requirements (I_{req}, C_{req}) and physical air data system requirement (AE, τ_{max}). Our MDE design can also be used for different pitot tube failure modes (e.g,. stuck or oscillatory fault). Note that analyzing a complete set of pitot tube failure modes would make the MDE calculation statistically more reliable, which is the subject of future studies. In what follows, we summarize the necessary IM performance trade-off design procedures:

1. Determine a suitable sampling rate for the measurement
2. Determine a reasonable AE based on the realistic fault mode profile (e.g., constant, ramp, or oscillatory)
3. Determine a feasible set of requirements I_{req} and C_{req} that satisfies the constraint in Equation (26) using DOC curves
4. If I_{req} and C_{req} are satisfied, then record the MDE and τ for assessing the fault detection performance
5. If I_{req} and C_{req} are not satisfied, return step 3. If the pair (I_{req}, C_{req}) is given as a hard requirement, then AE needs to be relaxed (return step 1)

5.2.5 | Important trade-off factors

The design procedure is also presented in a flowchart shown in Figure 10. It can be seen that when MDE and τ requirements are not satisfied, either performance requirements I_{req} and C_{req} are needed to be redefined, or AE and τ_{max} are needed to be re-adjusted. We also list the essential trade-off factors and their relationships that need to be considered for the IM performance trade-off design:

1. Sampling time versus auto-correlation
2. Effectiveness of test statistics versus sliding window size
3. System performance requirements (I_{req}, C_{req}) versus maximum allowable operating conditions (AE, τ_{max})
4. Sliding window size versus maximum allowable operating conditions (AE, τ_{max})

These variables are closely interconnected, and ultimately we need to find a set of (AE, τ_{max}) that is suitable for the given requirements (I_{req}, C_{req}). The choice of

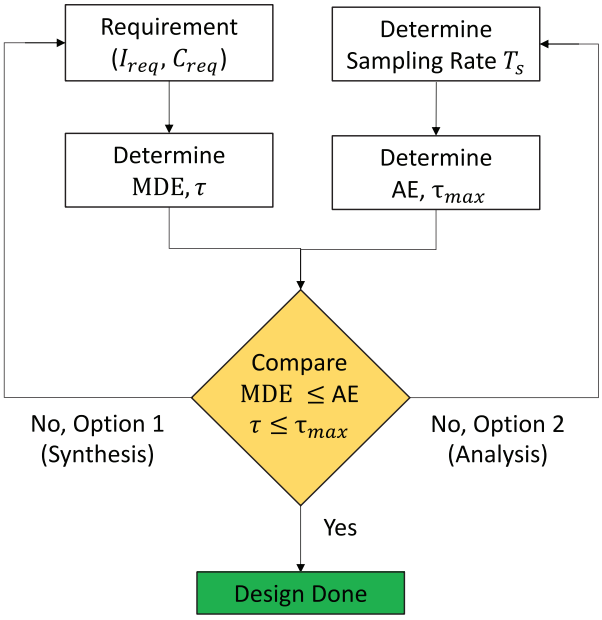


FIGURE 10 Trade-off design procedure [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

(AE, τ_{max}) would largely depend on the size of the sliding window, effectiveness of the chosen detectors (or test statistics), and sampling rate.

The fundamental limitation of change detection is that the design is a compromise between detecting true changes and avoiding false alarms. A poor design gives either a slow filter (no alarm from the test statistic) or a fast filter (many false alarms), and that is the worst can happen.

5.3 | MDE analysis for KF residual-based test statistic

The quality of a test statistic is often determined by its MDE (Grosch et al., 2017; Sturza, 1988). As demonstrated in Section 5.2.3, given a set of (P_{FA}, P_{MD}), the MDE (e.g., additive bias) can be found. The MDE informs us of the limits of the fault detection capability. In this section, we show the general expression of MDE for the residual-based test statistic. First, we can express $\mathbf{r}_{k-q:k}$ under the faulty H_1 hypothesis:

$$\mathbf{r}_{k-q:k} = (\mathbf{I} - \mathcal{O}\mathcal{O}^*)[\mathbf{Q}_w\mathbf{W} + \mathbf{V} + \mathbf{F}] \tag{32}$$

where $\mathbf{F}$ contains at most q steps since the start of the fault as defined in Equation (A6). $\mathbf{F}$ is the fault vector used to represent faults for different types of measurements. If we use both GNSS and airspeed measurement to update KF at the same time, then $\mathbf{F}$ can contain at most mq non-zero values. In this case, since the only faulty measurement is assumed to come from the pitot tube, we can further extract the faulty vector and maintain the correct matrix size by applying a column vector $\mathbf{E}_7$:

$$\mathbf{r}_{k-q:k} = (\mathbf{I} - \mathcal{O}\mathcal{O}^*)\left[\mathbf{Q}_w\mathbf{W} + \mathbf{V} + (\mathbf{E}_7^T\mathbf{F})\mathbf{E}_7\right] \tag{33}$$

where $\mathbf{E}_7 = [\boldsymbol{e}_7^T \quad \boldsymbol{e}_7^T \quad \dots \quad \boldsymbol{e}_7^T]^T \in \mathbb{R}^{7q\times 1}$. The unit vector $\mathbf{e}_7$ is used because the airspeed measurement is on the seventh row shown in Equation (4). The size of $\mathbf{I}$ is now $7q \times 7q$. Then, the expectation of the residual $\mathbf{r}_{k-q:k}$ is calculated as follows since $\mathbb{E}[\mathbf{W}]$ and $\mathbb{E}[\mathbf{V}]$ are zero respectively:

$$\mathbb{E}\left[\mathbf{r}_{k-q:k}\right] = (\mathbf{I} - \mathcal{O}\mathcal{O}^*)(\mathbf{E}_7^T\mathbf{F})\mathbf{E}_7 \tag{34}$$

Under the faulted hypothesis H_1, the residual-based test statistic follows a non-central χ^2 distribution with Df equal to $mq - n$:

$$D_{\mathbf{r},k}|H_1 \sim \chi^2_{(\lambda)}(mq - n) \tag{35}$$

The non-centrality parameter λ can be computed as:

$$\begin{aligned}\lambda &= \mathbb{E}\left[\mathbf{r}_{k-q:k}^T\boldsymbol{\Sigma}^{-1}\mathbf{r}_{k-q:k}\right] \\ &= \mathbf{E}_7\mathbf{F}^T\mathbf{E}_7^T(\mathbf{I} - \mathcal{O}\mathcal{O}^*)^T\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \mathcal{O}\mathcal{O}^*)\mathbf{E}_7^T\mathbf{F}\mathbf{E}_7 \\ &= \mathbf{E}_7\mathbf{F}^T\mathbf{E}_7^T\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \mathcal{O}\mathcal{O}^*)\mathbf{E}_7^T\mathbf{F}\mathbf{E}_7\end{aligned} \tag{36}$$

In general, fault can come from any measurement or a combination of different measurements, and the fault characteristic depends on the particular type of measurement used. For an inhomogeneous fault vector $\mathbf{F}$, λ can be expressed as $\mathbf{F}^T\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \mathcal{O}\mathcal{O}^*)\mathbf{F}$ without any loss of generality. This might create a challenge if isolation is required. Hence, we continue with the expression shown in Equation (36). The parameter λ can be determined by solving Equation (27) for given a set of P_{FA}, P_{MD}, and Df.

Assuming we have an exact q-step pitot tube faults in the vector $\mathbf{F}$, we can project λ to a q-step detectable error for the pitot tube by reformulating Equation (36):

$$\sum_{j=k-q+1}^{k} f_{V_{a_j}}^2 = \frac{\lambda}{\mathbf{E}_7^T\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \mathcal{O}\mathcal{O}^*)\mathbf{E}_7} \tag{37}$$

This formulation assumes the fault happens in every time step in the sliding window. In reality, the fault can occur at any time, but it can be only accounted up to q

steps. Therefore, the following definition of MDE of f_{V_a}, denoted as $\text{MDE}_{f_{V_a}}$, is conservative if q is large, but less conservative when compared to the definition of MDE in Equation (30). The relationship between $\text{MDE}_{f_{V_a}}$ and MDE is shown below:

$$\begin{aligned}\text{MDE}_{f_{V_a}} &\triangleq \frac{1}{q}\sqrt{\sum_{j=k-q+1}^{k} f_{V_{a_j}}^2} \\ &= \frac{1}{q}\frac{\sqrt{\lambda}}{\sqrt{\mathbf{E}_7^T\boldsymbol{\Sigma}^{-1}(\mathbf{I}-\boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)\mathbf{E}_7}} \\ &\leq \frac{\sqrt{\sum_{j=k-q+1}^{k} f_{V_{a_{max}}}^2}}{q} = f_{V_{a_{max}}} = \text{MDE}\end{aligned} \tag{38}$$

The $\text{MDE}_{f_{V_a}}$ can be interpreted as an average fault over the sliding window and is smaller than the MDE defined in Section 5.2.3. The difference between $\text{MDE}_{f_{V_a}}$ and MDE depends on the sliding window size and the fault profile. If a small sliding window size and a slow ramp fault profile are used, the MDE in the previous subsection is not a bad choice for the fault detection design.

6 | PROTECTION LEVEL CALCULATION

In the previous section, we introduced the test statistics and MDE design and analysis for the air data fault detection system. In this section, we derive a new protection level for α and β. We first introduce the definition of the alert limit and protection level in the context of synthetic air data, then we give the formal definition of the protection levels.

6.1 | Protection level and alert limit

We define the alert limit of angle-of-attack and sideslip angles needed for the protection level calculation. Alert Limit, denoted as AL, is usually defined as the maximum error in a state estimate that can be tolerated before a system is considered hazardous. Protection Level, denoted as PL, is defined as the guaranteed upper bound of the estimation error uncertainty σ. In theory, we want the probability of the state estimate error ϵ being greater than AL to be extremely low to assure integrity. Practically, we can also formulate the integrity requirement by using PL as follows[4]:

$$P(|\epsilon| > PL|H_i)P(H_i) \leq P_{MD} \text{ for } i = \{0,1\} \tag{39}$$

As a consequence, if we have $PL < AL$, the integrity requirement will be met. The error uncertainty σ is usually inflated by a factor K. This inflated error uncertainty $K\sigma$ is the protection level PL.

For a given integrity risk requirement P_{MD}, the PL are calculated. Protection Level PL is a function of the sensing system, and AL is a function of the operation. In other words, PL and AL are independent from each other.

In the case of pitot tube failures being considered here, we are interested in detecting WB faults before they result in the air data estimate (i.e., α and β) being outside of the safe-flight envelope, thereby leading to a control system (or a pilot in the case of a manned aircraft) to execute unnecessary but potentially hazardous maneuvers.

For the purpose of simplicity, we will assume the safe-flight envelope is a rectangle where the upper edge of the safe-flight envelope is defined by the UAVs maximum angle of attack α_{max} (which is the stall angle of attack α_{stall}). The lower edge is defined by a minimum angle of attack α_{min}, which is due to some aircraft structural considerations. The left and right edges are defined by β_{min} and β_{max}, which are derived either from aerodynamic control or structural strength limits. Therefore, we define the alert limit α_{AL}^{abs} and β_{AL}^{abs} as the *absolute* nominal safe operating region here since the true reference α and β are typically not available on UAVs. For the UAV considered here, the lower and upper bound of α_{AL}^{abs} and β_{AL}^{abs} is given in the following:

$$\begin{aligned}\alpha_{AL}^{\text{abs}} &= [\alpha_{min}, \alpha_{max}] = [-20°, 15°] \\ \beta_{AL}^{\text{abs}} &= [\beta_{min}, \beta_{max}] = [-30°, 30°]\end{aligned} \tag{40}$$

These boundaries of the safe-flight envelope form the absolute alert limits for the fault detection algorithm and are generally not the same for different UAVs.

Note that the protection level bound α_{PL} represents a deviation from the true state and the alert limit α_{AL} represents the *relative* error tolerance in the GNSS applications. However, the *absolute* alert limit α_{AL}^{abs} in this case is still valid. To see why this is the case, consider the following

[4] This expression is simplified from the following: $P(|\epsilon| > PL, D < T|H_i)P(H_i) = P(|\epsilon| > PL|H_i)P(D < T|H_i)P(H_i) \leq P(|\epsilon| > PL|H_i)$ $P(H_i) \leq P_{MD}$, where D is a detector function or test statistic, and T is a designed threshold

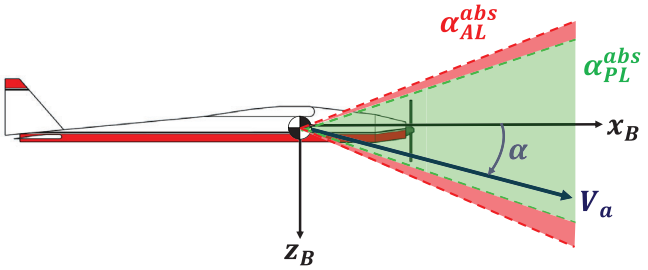


FIGURE 11 Depiction of protection level and alert limit of α [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

mathematical equivalence:

$$\alpha_{PL} < \alpha_{AL} \Rightarrow \alpha_{PL} + \hat{\alpha} < \alpha_{AL} + \hat{\alpha} \Rightarrow \alpha_{PL}^{\text{abs}} < \alpha_{AL}^{\text{abs}} \tag{41}$$

where α_{AL} represents the alert limit in the relative sense, and the absolute alert limit is defined as $\alpha_{AL}^{\text{abs}} \triangleq \alpha_{AL} + \hat{\alpha}$. Similarly, the absolute protection level is defined as $\alpha_{PL}^{\text{abs}} \triangleq \alpha_{PL} + \hat{\alpha}$. Hence, the new definition of α_{AL}^{abs} does not conflict with the typical definition of α_{AL}. In general, I_{req} is a set of discrete probability values representing various fault modes, and the probability of missed detection P_{MD_α} for α, as an example, would be part of the integrity budget I_{req}. Nevertheless, we will use the same value for the protection level calculation since we are only dealing with one fault mode, that is, $P_{MD_\alpha} = P_{MD_\beta} = P_{MD} = I_{req}$.

Figure 11 graphically depicts both protection level α_{PL}^{abs} and alert limit α_{AL}^{abs} in relation to α. Ideally, α_{PL}^{abs} should rarely go over α_{AL}^{abs} due to the small integrity risk requirement. When α_{PL}^{abs} does exceed α_{AL}^{abs}, we can safely conclude it is highly likely to have been the result of a faulty pitot tube and the integrity of α is lost.

6.2 | Protection level for synthetic air data

The fault detection algorithm needs to guarantee (at a certain level of confidence) that estimation error in α and β do not exceed their alert limits. Various slope-based PL calculations have been used to protect horizontal and vertical state errors against single (Walter & Enge, 1995; Brown & Chin, 1998; Milner & Ochieng, 2011) or multiple (Pervan et al., 1998; Angus, 2006; Blanch et al., 2009; Jiang & Wang, 2014) GNSS faults. However, these methods are developed for solving the redundant measurement problem.

Furthermore, the PL is usually calculated to protect states in an EKF. We develop a PL method that can protect the states derived from the EKF states. In particular, we calculate the PL for the synthetic angle-of-attack α and sideslip β estimates.

Under the fault-free hypothesis H_0, the PL is calculated by inflating state errors by a factor of K. In particular, the PL of α and β are calculated as follows:

$$\begin{aligned}\alpha_{PL,H_0} &= K_{\alpha,0}\sqrt{\mathbf{e}_1^T\mathbf{A}_{\alpha\beta}\mathbb{E}[\delta\boldsymbol{x}_k\delta\boldsymbol{x}_k^T]\mathbf{A}_{\alpha\beta}^T\mathbf{e}_1} \\ &= K_{\alpha,0}\sqrt{\mathbf{e}_1^T\mathbf{A}_{\alpha\beta}\boldsymbol{P}_k\mathbf{A}_{\alpha\beta}^T\mathbf{e}_1} = K_{\alpha,0}\sigma_\alpha\end{aligned} \tag{42}$$

$$\begin{aligned}\beta_{PL,H_0} &= K_{\beta,0}\sqrt{\mathbf{e}_2^T\mathbf{A}_{\alpha\beta}\mathbb{E}[\delta\boldsymbol{x}_k\delta\boldsymbol{x}_k^T]\mathbf{A}_{\alpha\beta}^T\mathbf{e}_2} \\ &= K_{\beta,0}\sqrt{\mathbf{e}_2^T\mathbf{A}_{\alpha\beta}\boldsymbol{P}_k\mathbf{A}_{\alpha\beta}^T\mathbf{e}_2} = K_{\beta,0}\sigma_\beta\end{aligned} \tag{43}$$

where $K_{\alpha,0}$ and $K_{\beta,0}$ under H_0 are calculated as follows:

$$K_{(\cdot),0} = Q^{-1}(P_{MD}/2) \tag{44}$$

where Q is the tail distribution function of the standard normal cdf. The matrix $\boldsymbol{P}_k$ is the state covariance from the EKF and $\mathbf{A}_{\alpha\beta}$ is the flow angle propagation transformation matrix specified in Sun et al. (2019b). The unit vectors $\mathbf{e}_1 = [1,0]^T$ and $\mathbf{e}_2 = [0,1]^T$ are used to extract the diagonal terms of $\mathbf{A}_{\alpha\beta}\boldsymbol{P}_k\mathbf{A}_{\alpha\beta}^T$.

In the presence of fault under the hypothesis H_1, PL needs to be increased to account for the faulty airspeed component f_{V_a}. The formulation is done as follows according to (Brown & Chin, 1998; Angus, 2006):

$$\alpha_{PL,H_1} = \text{slope}_\alpha\sqrt{\lambda_U} + K_{\alpha,1}\sigma_\alpha = \frac{\sigma_\alpha}{\sqrt{D_{\mathbf{r},k}}}\sqrt{\lambda_U} + K_{\alpha,1}\sigma_\alpha \tag{45}$$

$$\beta_{PL,H_1} = \text{slope}_\beta\sqrt{\lambda_U} + K_{\beta,1}\sigma_\beta = \frac{\sigma_\beta}{\sqrt{D_{\mathbf{r},k}}}\sqrt{\lambda_U} + K_{\beta,1}\sigma_\beta \tag{46}$$

where slope_α and slope_β represent the ratio between the α or β state error and the standard deviation of the test statistic $D_{\mathbf{r},k}$. The notion of the maximum slope (Brown & Chin, 1998) is not applicable here since there is no redundant airspeed at each time step to calculate a set of slopes. Specifically, since the accumulated sequential residual or innovation vectors come from the single pitot tube, so the source of the fault is always known and singular. In other words, the multiple hypothesis tests of determining which GNSS measurement is faulty by calculating the maximum slope does not apply here. The slope value in this study depends on the test statistic window size and severity of the fault profile.

The inflation factors $K_{\alpha,1}$ and $K_{\beta,1}$ under H_1 can be calculated as follows:

$$K_{(\cdot),1} = Q^{-1}\left(\frac{P_{MD}}{2P_{H_1}}\right) \tag{47}$$

where P_{H_1} represents the failure rate of the pitot tube due to water blockage. Generally, the failure rate of the pitot tube due to a particular fault is calculated based on rigorous sensor characterization testing. Since the failure rate testing is not done for this study, we assume $K_{(\cdot),1} = K_{(\cdot),0} = Q^{-1}(P_{MD}/2)$ for both α and β. This assumption is valid but conservative as $K_{(\cdot),1}$ is generally greater than $K_{(\cdot),0}$ in general. The parameter λ_U is the upper confidence bound for the maximum q-step airspeed faults, and it is computed as follows:

$$\begin{aligned}\lambda &= \mathbf{E}_7\mathbf{F}^T\mathbf{E}_7^T\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)\mathbf{E}_7^T\mathbf{F}\mathbf{E}_7 \\ &\leq \mathbf{E}_7\mathbf{F}^T\mathbf{E}_7^T\boldsymbol{\Sigma}^{-1}\mathbf{E}_7^T\mathbf{F}\mathbf{E}_7 \\ &\leq \mathbf{E}_7\mathbf{F}^T\mathbf{E}_7^T\boldsymbol{R}^{-1}\mathbf{E}_7^T\mathbf{F}\mathbf{E}_7 \\ &\leq \sum_{j=k-q+1}^{k} \frac{f_{V_{a_j}}^2}{\sigma_{V_{a_j}}^2} = \sqrt{\overline{\text{MDE}}} = \lambda_U\end{aligned} \tag{48}$$

This upper bound λ_U can make the protection levels overly conservative if the size of the sliding window q is large. Hence, the tightness of the protection level is another factor for choosing an appropriate q.

FIGURE 12 PHX by Sentera LLC [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

It is worth noting that many small UAVs do not have angle vane sensors to provide a set of α and β. The protection levels of α and β are particularly useful when the angle vane sensor is not available. We can monitor α and β based on the synthetic air data estimates and protect the vehicle from exceeding α and β flight envelopes due to the pitot tube failure.

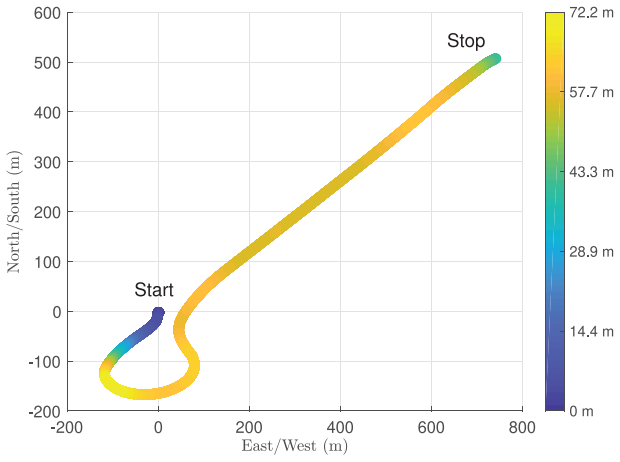
7 | FLIGHT DATA TESTING

The fault detection algorithm is tested using a flight data set recorded by a UAV. The UAV is the Sentera Phoenix (PHX) shown in Figure 12. It has an Eagle Tree pitot tube (Eagle Tree Systems, 2020) attached to the right wing to measure airspeed. PHX utilizes a similar version of Pixracer autopilot (Pixhawk FMU-V4) for control and navigation.

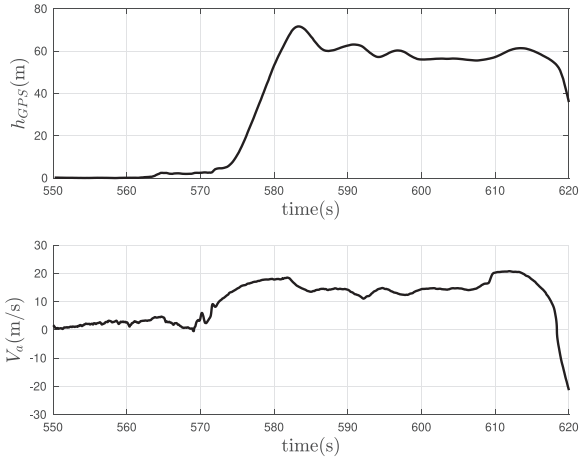
For this particular flight operation, the UAV crashed 45 seconds into the flight due to a water-plugged faulty pitot tube. This flight data is suitable for the fault detection algorithm analysis since it contains a known WB pitot tube fault signature.

Also, since the UAV has only one pitot tube, only one SADS is employed to show the flight results. Ideally if this UAV carried two pitot tubes and two independent SADS, then it would follow the decision logic in Table 1 to raise alarm and switch to SADS-2 when SADS-1 detected and isolated the fault airspeed measurement.

Figure 13(a) and 13(b) show the trajectory, altitude, and airspeed over the short flight period before the crash. The UAV takes off around 570 s and circles up to about 70 m, then flies straight northwest direction to the edge of the crop field. It is seen in Figure 13b that the airspeed of UAV experiences a sharp drop right after 618 s. The airspeed



(a) Recorded Trajectory using GPS data.



(b) Recorded altitude and airspeed over time.

FIGURE 13 Trajectory and airspeed information over the time of flight [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

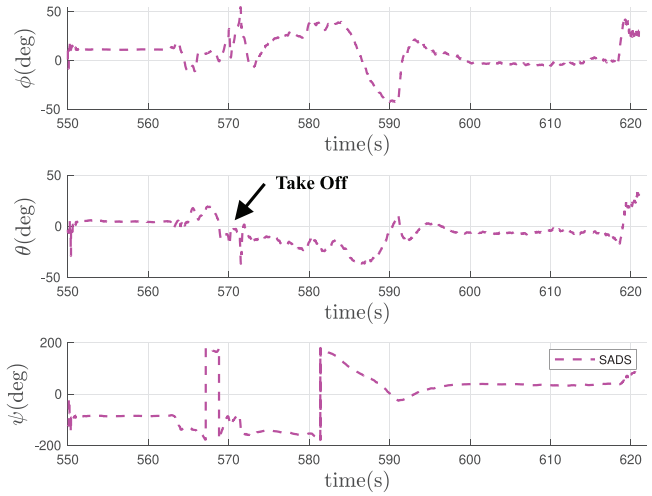


FIGURE 14 SADS attitude estimates over time [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

measurement eventually drops to a negative value. The last recorded altitude is around 35 m by the onboard autopilot.

Figure 14 shows the attitude estimates from the SADS filter. Notice that the Euler angle estimates start to change abruptly at the end of the flight due to the faulty pitot tube. For example, the pitch angle θ and roll angle ϕ increased dramatically around 618 s. The high value θ indicates the UAV might have been pitching up and stalling.

Figure 15 shows three different test statistics and their corresponding thresholds from the actual takeoff timestamp of 570 s to the end of the flight.

The top sub-figure shows the sliding window innovation-based test statistic D_γ over time where the window size q is 19. The bottom sub-figure shows a residual-based test statistic $D_{\mathbf{r}}$ and a Geometric Moving Average (GMA) residual-based test statistic $D_{\mathbf{r}}^{\text{GMA}}$, where the window size q is also 19 for both. The $D_{\mathbf{r}}^{\text{GMA}}$ is defined as the same as $D_{\mathbf{r}}$ except the residual vector $\mathbf{r}_{k-q:k}^{\text{GMA}}$ is written as follows:

$$\mathbf{r}_{k-q:k}^{\text{GMA}} = \left[\lambda^q \mathbf{r}_{k-q}^T \ \lambda^{q-1}\mathbf{r}_{k-q+1}^T \ \ldots \ \lambda \mathbf{r}_{k-1}^T \ \mathbf{r}_k^T\right]^T \tag{49}$$

In Equation (49), it is seen that the $D_{\mathbf{r}}^{\text{GMA}}$ uses an exponential forgetting factor μ^i to weigh on the past measurements less than the recent. The innovation-based threshold T_γ and residual-based $T_{\mathbf{r}}$ are different from each other due to the difference in Df, even though the same size of the sliding window is used.

It is seen that the innovation-based test statistic exceeds its threshold three times. It appears that the first occurrence at 593 s is most likely a false alarm given the fact we know the pitot tube WB fault occurs at the end of the flight. Furthermore, by examining the attitude in Figure 14, the large ϕ and sudden change in θ suggests a momentary accelerated stall.

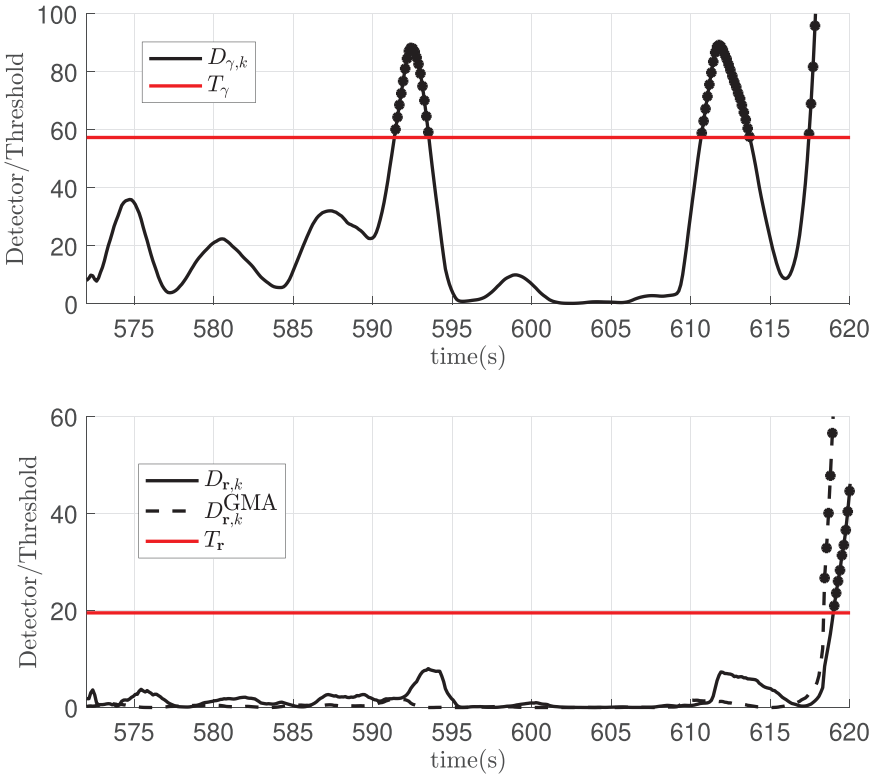


FIGURE 15 Detection variables over time [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

It is well known that the innovation-based test statistic is sensitive to the highly dynamic motion. The UAV experiences a large roll change at 593 s, which may cause D_γ to think there might be a fault in the system. The second occurrence might come from a sudden increase in airspeed at 610 s. Though we do not know if the fault has occurred yet by visually examining the airspeed plot, D_γ indicates there might be a fault.

Notice D_γ goes below the threshold at 614 s before rising to cross the threshold again at 618 s for the third time. This unstable behavior is not a good characteristic for a detector because it will be issuing many false alarms and failing to provide a crisp decision of a fault's occurrence.

It is also well known that the innovation-based test statistic is sensitive to sensor noise scaling (Gustafsson, 2000), and the innovation covariance $\mathbf{S}$ inside of D_γ is sensitive to noise scaling. The relationship is illustrated by Equation (50): $\mathbf{S}$ is affected by noise covariance matrices $\mathbf{R}_w$ and $\mathbf{R}$, and the initial state covariance $\mathbf{P}_0$. Small η can make D_γ sensitive to noise even though the state estimates are not affected.

$$\left.\begin{aligned} \bar{\mathbf{R}} &= \eta\mathbf{R} \\ \bar{\mathbf{R}}_w &= \eta\mathbf{R}_w \\ \bar{\mathbf{P}}_0 &= \eta\mathbf{P}_0 \end{aligned}\right\} \Rightarrow \begin{aligned} \bar{\mathbf{P}} &= \eta\mathbf{P} \\ \bar{\mathbf{S}} &= \eta\mathbf{S} \end{aligned} \tag{50}$$

Though not shown here, by scaling the process noise $\sigma_{\mathbf{w}_{Wd}}$ of the wind vector without adversely changing the performance of the estimator, the first two crossing occurrences of D_γ can be suppressed. However, hand-tuning is not recommended because it might adversely affect other estimates.

On the other hand, both residual-based test statistics $D_\mathbf{r}$ and $D_\mathbf{r}^{\text{GMA}}$ exceed their corresponding thresholds only once at the end of the flight. The threshold value for $D_\mathbf{r}$ and $D_\mathbf{r}^{\text{GMA}}$ are the same, hence only one $T_\mathbf{r}$ is plotted. Though small increases at 593 s and 612 s can be seen in $D_\mathbf{r}$, $D_\mathbf{r}$ did not cross its threshold.

The residual-based test statistic is less sensitive to both highly dynamic motion and noise scaling due to the larger weighting factor Σ. Furthermore, the GMA test statistic $D_\mathbf{r}^{\text{GMA}}$ appears to be even less sensitive. $D_\mathbf{r}^{\text{GMA}}$ is discounting many early measurements in the sliding window. Also, it rises faster than $D_\mathbf{r}$ when crossing the threshold, indicating a shorter fault detection time. When comparing detection time alone, the innovation does appear to have the fastest detection at the third crossing.

The GMA technique used here illustrates that sometimes different techniques can be used to improve the baseline detector function. For example, we may combine the CUSUM and residual-based test statistics to improve detectability. Ultimately, it is a trade-off between the detection time and false alarm when choosing a detection test statistic. In this case, it appears the GMA residual-based test statistic $D_\mathbf{r}^{\text{GMA}}$ is the best for the faulty pitot tube detection since it crosses the threshold at the correct incident and provides a good detection time.

In order to see how observability can actually play an important role in fault detection, we also monitor the observability Gramian of the system over time. Different metrics can measure the degree of discrete observability Gramian, such as the determinant, trace, or the condition number (Summers et al., 2016; Avant & Morgansen, 2019). In work here, we utilize the condition number κ to quantify the degree of observability. That is:

$$\kappa\left[\mathcal{G}_{d,k-q:k}\right] \triangleq \kappa\left[\boldsymbol{\mathcal{O}}_{k-q:k}^T\boldsymbol{\mathcal{O}}_{k-q:k}\right] \tag{51}$$

where $\mathcal{G}_{d,k-q:k}$ the discrete observability Gramian using the information from the time step k-q to the current k. In general, if the condition number of the observability Gramian is large, it means the states are not well observed. The observability Gramian here only requires a finite horizon instead of the infinite horizon. This calculation is done to monitor the recent motion of the dynamics of the vehicle. It also reduces the computational burden for the computer processor.

Figure 16 shows the normalized condition number of observability Gramian over time. The poor condition number is expected before takeoff since the observability

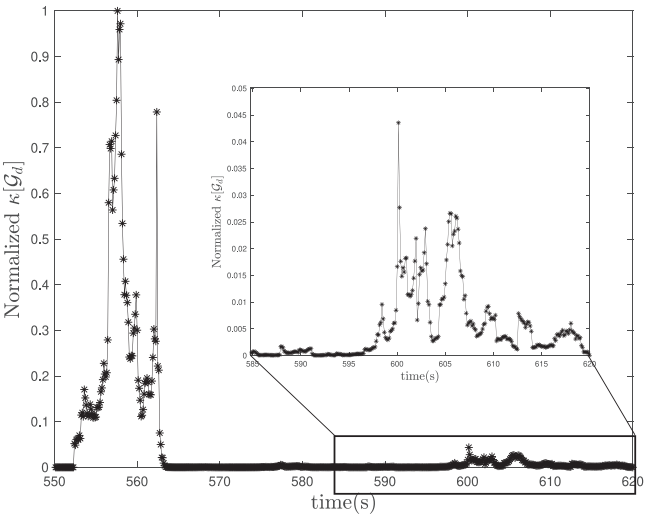


FIGURE 16 Normalized condition number of observability Gramian

Gramian rank is deficient. The condition number quickly reaches near zero after 570 s (takeoff). However, it is seen that the condition number got a bit worse from 600 s to 607 s. This change is perhaps due to the straight level flight (no heading change), or some water started to enter the pitot tube before it became evident.

The second claim is deduced based on examining all the EKF state estimates at 600 s. Since there is no abnormal phenomenon from the position, velocity, and attitude estimates around 600 s, the IMU and GNSS measurements are assumed to be nominal. Hence, the culprit is either the poor observability due to the trajectory or the pitot tube.

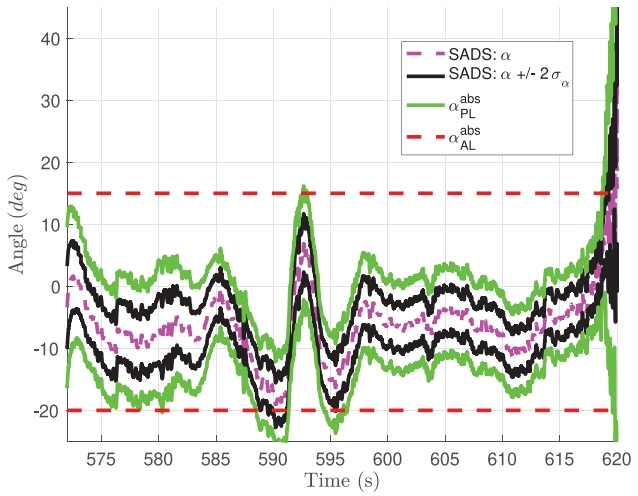
A poor condition number can potentially trigger the fault detection algorithm to raise the alarm even though there is no real fault. If there is a fault that is about to happen, then monitoring the observability Gramian can potentially warn the system before the fault occurs. If a poor condition number is a result of the straight-level flight, then weak observability can mask the fault detectability if a fault occurs during this time. Therefore, a close examination should be carried out when using observability to detect the actual fault.

Figures 17(a) and 17(b) show the α and β estimates, 2σ uncertainty bounds, protection levels α_{PL} and β_{PL}, and alert limits α_{AL} and β_{AL}. The protection level bound is for the single SAD. The protection level α_{PL} goes outside the alert limit momentarily at 590 s, 592 s, and 595 s, and exceeds both lower and upper alert limit at the end of the flight. The s-turn causes the first three crossings before heading to northwest direction. The last one is caused by the faulty pitot tube. It can be seen that the α increases drastically at the end of the flight, which leads to the stall and eventual crashing. Sideslip β also experiences a similar change during the s-turn. The sideslip estimate β changes from positive to negative, then positive again around 590 s, which is intuitively correct based on the s-turn and the northeast tail wind direction.

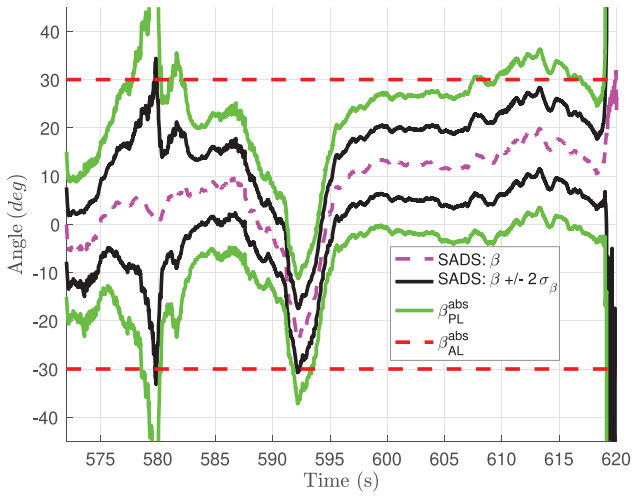
The protection level β_{PL} exceeds the alert limit a couple of times and eventually goes outside the alert limit at the end of the flight as expected. The protection level of α and β can be used to check the validity of ADS integrity, which is much more useful than just looking at the confidence uncertainty bound represented by 2σ. For example, we can see that the α and its $2\sigma_\alpha$ seem to be acceptable from 593 s to 596 s, but the integrity of the α estimate is actually lost during this time based on the protection level. This loss of integrity can be used as another flag for the validity of the α estimate.

8 | CONCLUSION AND FUTURE OUTLOOK

A dual pitot tube ADS is designed with fault detection and isolation capability for small UAS. The purpose of this algorithm is to provide a reliable ADS and recovery strategy for safe drone operations in case of the pitot tube water-blockage fault under rainy and foggy conditions. The fault detection algorithm is designed to detect faults based on the given integrity requirements and known water-blockage fault profile.



(a) Angle-of-attack α estimate, 2 σ uncertainy bound, protection level and alert limit.



(b) Sideslip β estimate, 2 σ uncertainy bound, protection level and alert limit.

FIGURE 17 Flow angle α and sideslip β estimate, uncertainty bound, protection level, and alert limit [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

Systematic procedures and various factors are laid out to show how to design the fault detection algorithm. Though the stringent performance requirement, such as 10^{-7}, may not be realizable to certify ADS in small UAS due to the low-cost sensors onboard, this IM approach allows engineers to assess the performance of the FDI algorithm from the requirement point of view. The high-performance requirements can potentially be achieved for the ADS in small UAS when highly accurate sensors (e.g., good GPS and IMU), good control design (reject external disturbances), and sensible path planning (observability-aware trajectory design) are employed.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the Minnesota Invasive Terrestrial Plants and Pests Center (MITPPC) through the Minnesota Environment and Natural Resources Trust Fund for financial support to conduct research associated with increasing the reliability of small UAV technology used for surveying applications. The authors also gratefully acknowledge Todd Colten and Sentera LLC for donating the flight data for the air data fault detection research.

ORCID

Kerry Sun https://orcid.org/0000-0002-5310-2815
Demoz Gebre-Egziabher https://orcid.org/0000-0001-6623-9602

REFERENCES

Angus, J. E. (2006). Raim with multiple faults. *NAVIGATION*, *53*(4), 249–257. https://doi.org/10.1002/j.2161-4296.2006.tb00387.x

Arana, G. D., Hafez, O. A., Joerger, M. & Spenko, M. (2019a). Recursive integrity monitoring for mobile robot localization safety. *International Conference on Robotics and Automation (ICRA)*, (pp. 305–311). https://doi.org/10.1109/ICRA.2019.8794115

Arana, G. D., Joerger, M. & Spenko, M. (2019b). Efficient integrity monitoring for kf-based localization. *International Conference on Robotics and Automation (ICRA)*, (pp. 6374–6380). https://doi.org/10.1109/ICRA.2019.8794362

Austrailian Transport Safety Bureau. (2015). *Erratic airspeed indications Boeing 787-8, VH-VKE (Technical Report)*. Austrailian Transport Safety Bureau. https://www.atsb.gov.au/media/5773029/ao-2015-149-final.pdf

Avant, T., & Morgansen, K. A. (2019). *Observability properties of object pose estimation*. American Control Conference (ACC), (pp. 5134–5140). https://doi.org/10.23919/ACC.2019.8814791

Bar-Shalom, Y., Li, X., & Kirubarajan, T. (2002). *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*. John Wiley & Sons, Ltd. https://doi.org/10.1002/0471221279.ch5

Berman, Z. & Powell, J. D. (1998, April). The role of dead reckoning and inertial sensors in future general aviation navigation. In *IEEE Position Location and Navigation Symposium*, (pp. 510–517). https://doi.org/10.1109/PLANS.1998.670206

Bhamidipati, S. & Gao, G. X. (2019, September). SLAM-based integrity monitoring using GPS and fish-eye camera. *Proc. of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019)*, (pp. 4116–4129). https://doi.org/10.33012/2019.17117

Blanch, J., Walter, T., & Enge, P. (2009). RAIM with optimal integrity and continuity allocations under multiple failures. *IEEE Transactions on Aerospace and Electronic Systems*, *46*(3), 1235–1247. https://doi.org/10.1109/TAES.2010.5545186

Borup, K. T. (2018). Air Data Estimation for Small Unmanned Aircraft. (Doctoral dissertation, Norwegian University of Science and Technology). http://hdl.handle.net/11250/2565573

Brown, R. G., & Chin, G. (1998). *GPS RAIM: Calculation of thresholds and protection radius using chi-square methods ; a geometric approach*. The Institute of Navigation Monograph Series.

Harbo, A. L. Cour- (2017). Quantifying risk of ground impact fatalities of power line inspection bvlos flight with small unmanned aircraft. *International Conference on Unmanned Aircraft Systems (ICUAS)*, (pp. 1352–1360). https://link.springer.com/article/10.1007/s10846-018-0853-1

Eagle Tree Systems (2020). Prandtl Pitot Tube Kit.

Eubank, R., Atkins, E., & Ogura, S. (2010). *Fault detection and fail-safe operation with a multiple-redundancy air-data system*. AIAA Guidance, Navigation, and Control Conference. https://doi.org/10.2514/6.2010-7855

Fang, S., O'Young, S., & Rolland, L. (2018). Development of small UAS Beyond-Visual-Line-Of-Sight (BVLOS) flight operations: System requirements and procedures. *Drones*, *2*, 13. https://doi.org/10.3390/drones2020013

Federal Aviation Administration. (2019). 14 AC Part 135 Air Carrier and Operator Certification. https://www.ecfr.gov/cgi-bin/text-idx?rgn=div5&node=14%3A3.0.1.1.11&se14.3.135_1144#se14.3.135_1149

Federal Democratic Republic of Ethiopia, Ministry of Transport, Aircraft Accident Investigation Bureau. (2020, March). Aircraft Accident Investigation Bureau Interim Report. http://www.aib.gov.et/wp-content/uploads/2020/documents/accident/ET-302InterimInvestigationReportMarch92020.pdf

Fravolini, M. L., del Core, G., Papa, U., Valigi, P., & Napolitano, M. R. (2017). Data-driven schemes for the robust fault detection of aircraft air data system sensors. *IEEE Transactions on Control Systems Technology*, *27*(1). https://doi.org/10.1109/TCST.2017.2758345

Fravolini, M. L., Napolitano, M. R., Core, G. D., & Papa, U. (2018, September). Experimental interval models for the robust fault detection of aircraft air data sensors. *Control Engineering Practice*, *78*, 196–212. http://www.sciencedirect.com/science/article/pii/S0967066118302326

Freeman, P. (2014). Reliability assessment for low-cost unmanned aerial vehicles. (Ph.D. thesis, University of Minnesota). https://hdl.handle.net/11299/170136

Freeman, P., Seiler, P., & Balas, G. J. (2013). Air data system fault modeling and detection. *Control Engineering Practice*, *21*(10), 1290–1301. https://doi.org/10.1016/j.conengprac.2013.05.007

Gleason, S., & Gebre-Egizabher, D. (2009). *GNSS Applications and Methods, Chapter 6,7*. Norwood, MA: Artech Houser.

Grosch, A., Crespillo, O. G., Martini, I. & Günther, C. (2017). Snapshot residual and Kalman filter based fault detection and exclusion schemes for robust railway navigation. *European Navigation Conference (ENC)*, (pp. 36–47). https://doi.org/10.1109/EURONAV.2017.7954171

Guo, D., Zhong, M., & Zhou, D. (2018). Multisensor data-fusion-based approach to airspeed measurement fault detection for unmanned aerial vehicles. *IEEE Transactions on Instrumentation and Measurement*, *67*(2), 317–327. https://doi.org/10.1109/TIM.2017.2735663

Gustafsson, F. (2000). *Adaptive Filtering and Change Detection*. John Wiley & Sons, Inc.

Hansen, S., & Blanke, M. (2014). Diagnosis of airspeed measurement faults for unmanned aerial vehicles. *IEEE Transactions on Aerospace and Electronic Systems*, *50*(1), 224–239. https://doi.org/10.1109/TAES.2013.120420

Hu, B., & Seiler, P. (2015). A probabilistic method for certification of analytically redundant systems. *International Journal of Applied Mathematics and Computer Science*, *25*(1), 103–116. http://eudml.org/doc/270194

Isermann, R. (2005). *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Springer Berlin Heidelberg. https://books.google.com/books?id=6yUfoZhGMY0C

JDrones. (2020). Pitot Tube and Silicon Tube. Last accessed: May 29, 2020. http://store.jdrones.com/AirSpeed_Pitot_tube_for_UAV_p/senpito01.htm

Jiang, Y., & Wang, J. (2014). A new approach to calculate the vertical protection level in A-RAIM. *Journal of Navigation*, *67*(4), 711–725. https://doi.org/10.1017/S0373463314000204

Joerger, M., Chan, F., & Pervan, B. (2014). Solution separation versus residual-based raim. *NAVIGATION*, *61*(4), 273–291. https://doi.org/10.1002/navi.71

Joerger, M., & Pervan, B. (2013). Kalman filter-based integrity monitoring against sensor faults. *Journal of Guidance, Control, and Dynamics*, *36*(2), 349–361. https://doi.org/10.2514/1.59480

Johnson, M., Jung, J., Rios, J., Mercer, J., Homola, J., Prevot, T., Mulfinger, D., & Kopardekar, P. (2017). *Flight test evaluation of an unmanned aircraft system traffic management (utm) concept for multiple beyond-visual-line-of-sight operations*. Twelfth USA/Europe Air Traffic Management Research and Development Seminar.

Kassas, Z. M., & Humphreys, T. E. (2014). Observability analysis of collaborative opportunistic navigation with pseudorange measurements. *IEEE Transactions on Intelligent Transportation Systems*, *15*(1), 260–273. https://doi.org/10.1109/TITS.2013.2278293

Khanafseh, S., Roshan, N., Langel, S., Chan, F., Joerger, M., & Pervan, B. (2014). *GPS spoofing detection using RAIM with INS coupling, (pp. 1232–1239)*. IEEE/ION Position, Location and Navigation Symposium. https://doi.org/10.1109/PLANS.2014.6851498

Komite Nasional Keselamatan Transportasi, Republic of Indonesia. (2018). *Aircraft accident investigation report - pt. lion mentari airlines boeing 737-8 (max) pk-lqp tanjung karawang, west java republic of indonesia (Technical Report)*. Komite Nasional Keselamatan Transportasi. https://reports.aviation-safety.net/2018/20181029-0_B38M_PK-LQP_PRELIMINARY.pdf

Kotikalpudi, A., Danowsky, B. P., & Seiler, P. J. (2020). *Reliability analysis for small unmanned air vehicle with algorithmic redundancy*. AIAA Scitech Forum. https://doi.org/10.2514/6.2020-0739

Lee, Y. C. (1986). Analysis of range and position comparison methods as a means to provide GPS integrity in the user receiver. *Proc. of the 42nd Annual Meeting of The Institute of Navigation*. https://www.ion.org/publications/abstract.cfm?articleID=12197

Lie, F. A. P., & Gebre-Egziabher, D. (2013). Synthetic air data system. *Journal of Aircraft*, *50*(4), 1234–1249. https://doi.org/10.2514/1.C032177

Lu, P., Van Eykeren, L., Van Kampen, E., De Visser, C. C., & Chu, Q. P. (2016). Adaptive three-step kalman filter for air data sensor fault detection and diagnosis. *Journal of Guidance, Control, and Dynamics*, *39*(3), 590–604. https://doi.org/10.2514/1.G001313

McCrink, M., & Gregory, J. W. (2018). Design and development of a high-speed uas for beyond line-of-sight operation. *AIAA Information Systems-AIAA Infotech at Aerospace*. https://doi.org/10.2514/6.2018-0750

Milner, C. D., & Ochieng, W. Y. (2011). Weighted raim for apv: The ideal protection level. *Journal of Navigation*, *64*(1), 61–73. https://doi.org/10.1017/S0373463310000342

Ossmann, D., Joos, H., & Goupil, P. (2017). Enhanced sensor monitoring to maintain optimal aircraft handling in case of faults. *Journal of Guidance, Control, and Dynamics*, *40*(12), 3127–3137. https://doi.org/10.2514/1.G002341

Parkinson, B., & Axelrad, P. (1988, August). Autonomous GPS integrity monitoring using the pseudorange residual. *NAVIGATION*, *35*(2), 255–274. https://doi.org/10.1002/j.2161-4296.1988.tb00955.x

Pervan, B. (1996). *Navigation integrity for aircraft precision landing using the global positioning system*. (Ph.D. thesis, Stanford University). https://web.stanford.edu/group/scpnt/gpslab/pubs/theses/BorisPervanThesis96.pdf

Pervan, B., Pullen, S., & Christie, J. (1998). A multiple hypothesis approach to satellite navigation integrity. *NAVIGATION*, *45*(1), 61–84. https://doi.org/10.1002/j.2161-4296.1998.tb02372.x

Rohloff, T. J., Whitmore, S. A., & Catton, I. (1999). Fault-tolerant neural network algorithm for flush air data sensing. *Journal of Aircraft*, *36*(3), 541–549. https://doi.org/10.2514/2.2489

SeekingAlpha. (2019). Here Are The Disturbing Internal Emails Boeing Just Released About The 737 MAX. https://seekingalpha.com/article/4316117-are-disturbing-internal-emails-boeing-just-released-737-max. Last accessed: June 06, 2020. https://seekingalpha.com/article/4316117-are-disturbing-internal-emails-boeing-just-released-737-max

Sturza, M. A. (1988). Navigation system integrity monitoring using redundant measurements. *NAVIGATION*, *35*(4), 483–501. https://doi.org/10.1002/j.2161-4296.1988.tb00975.x

Summers, T. H., Cortesi, F. L., & Lygeros, J. (2016). On submodularity and controllability in complex dynamical networks. *IEEE Transactions on Control of Network Systems*, *3*(1), 91–101. https://doi.org/10.1109/TCNS.2015.2453711

Sun, K., Regan, C. D. & Egziabher, D. G. (2018). GNSS/INS based estimation of air data and wind vector using flight maneuvers. In *2018 IEEE/ION Position, Location, and Navigation Symposium (PLANS)*, (pp. 838–849). https://doi.org/10.1109/PLANS.2018.8373461

Sun, K., Regan, C. D., & Gebre-Egziabher, D. (2019a). *A GNSS/IMU-based 5-hole pitot tube calibration algorithm*. AIAA Scitech Forum. https://doi.org/10.2514/6.2019-0360

Sun, K., Regan, C. D., & Gebre-Egziabher, D. (2019b). Observability and performance analysis of a model-free synthetic air data estimator. *Journal of Aircraft*, *56*(4), 1471–1486. https://doi.org/10.2514/1.C035290

Tanıl, C., Khanafseh, S., Joerger, M., & Pervan, B. (2017a). An INS monitor to detect GNSS spoofers capable of tracking vehicle position. *IEEE Transactions on Aerospace and Electronic Systems*, *54*(1), 131–143. https://doi.org/10.1109/TAES.2017.2739924

Tanıl, C., Khanafseh, S., & Pervan, B. (2017b). Detecting global navigation satellite system spoofing using inertial sensing of aircraft disturbance. *Journal of Guidance, Control, and Dynamics*, *40*(8), 2006–2016. https://doi.org/10.2514/1.G002547

Van Eykeren, L., & Chu, Q. P. (2014). Sensor fault detection and isolation for aircraft control systems by kinematic relations. *Control Engineering Practice*, *31*, 200–210. http://www.sciencedirect.com/science/article/pii/S0967066114000884

Vascik, P. D., Hansman, R. J., & Dunn, N. S. (2018). Analysis of urban air mobility operational constraints. *Journal of Air Transportation*, *26*(4), 133–146. https://doi.org/10.2514/1.D0120

Walter, T. & Enge, P. (1995). Weighted RAIM for precision approach. *Proc. of the 8th International Technical Meeting of the Satellite Division of The Institute of Navigation*. pp. 1995-2004. https://www.ion.org/publications/abstract.cfm?articleID=2524

Walter, T., Enge, P., Blanch, J., & Pervan, B. (2008). Worldwide vertical guidance of aircraft based on modernized GPS and new integrity augmentations. *Proceedings of the IEEE*, *96*(12), 1918–1935. https://doi.org/10.1109/JPROC.2008.2006099

Xing, Z. (2010). *Over-Bounding Integrated INS/GNSS Output Errors*. (Ph.D. thesis, University of Minnesota Twin Cities). https://hdl.handle.net/11299/101401

Yapp, J., Seker, R. & Babiceanu, R. (2018). Providing accountability and liability protection for UAV operations beyond visual line of sight. IEEE Aerospace Conference, (pp. 1–8). https://doi.org/10.1109/AERO.2018.8396532

Yeh, Y. C. (1996). Triple-triple redundant 777 primary flight computer. *IEEE Aerospace Applications Conference. Proceedings*, *1*, 293–307. https://doi.org/10.1109/AERO.1996.495891

How to cite this article: Sun, K., & Gebre-Egziabher, D. (2021). Air data fault detection and isolation for small UAS using integrity monitoring framework. *NAVIGATION*, *68,* 577–600. https://doi.org/10.1002/navi.440

APPENDIX A: BATCH LINEAR SYSTEM REALIZATION

In this appendix, we present one form of the batch linear system realization to faciliate the connection between the LTV observability matrix and the residual-based test statistic shown in Section 5. We start with a model of the discrete linear dynamic system with an unknown additive fault:

$$\begin{aligned}\mathbf{z}_k &= \mathbf{H}_k\boldsymbol{x}_k + \boldsymbol{v}_k + \mathbf{f}_k\\ \mathbf{x}_{k+1} &= \boldsymbol{\Phi}_k\boldsymbol{x}_k + \boldsymbol{\Gamma}_k\mathbf{w}_k\end{aligned} \tag{A1}$$

where $\mathbf{f}_k$ is the deterministic additive fault vector, and $\mathbf{x}_k$, $\mathbf{z}_k$, $\mathbf{w}_k$ and $\boldsymbol{v}_k$ are defined in Section 2.2. The vector $\mathbf{w}_k$ is the process noise vector and is assumed to follow $N(\mathbf{0}, \mathbf{R}_w)$. The matrices $\boldsymbol{\Phi}_k$ and $\boldsymbol{\Gamma}_k$ are the state transition matrix and discrete noise coefficient matrix, respectively.

Only additive faults are chosen for this study to analyze the feasibility of air data fault detection capability. Nonlinear fault with more sophisticated models should be considered in the future.

Since there are not enough redundant airspeed measurements at each time step k, we obtain the following batch realization by stacking all the measurement vectors from the past time $k-q$ to the current time step k in Equation (A2):

$$\begin{aligned}\mathbf{Z}_{k-q:k} &= \boldsymbol{\mathcal{O}}_{k-q:k}\boldsymbol{x}_{k-q} + \mathbf{Q}_{w,k-q:k}\mathbf{W}_{k-q:k}\\ &\quad + \mathbf{V}_{k-q:k} + \mathbf{F}_{k-q:k}\end{aligned} \tag{A2}$$

The matrices $\mathbf{Z}_{k-q:k}$, $\mathbf{W}_{k-q:k}$, $\mathbf{V}_{k-q:k}$, $\mathbf{F}_{k-q:k}$, $\boldsymbol{\mathcal{O}}_{k-q:k}$ and $\mathbf{Q}_{w,k-q:k}$ are defined in the following:

$$\mathbf{Z}_{k-q:k} = \begin{bmatrix}\mathbf{z}_{\mathbf{k-q}}^{\mathbf{T}} & \mathbf{z}_{k-q+1}^{T} & \dots & \mathbf{z}_k^{T}\end{bmatrix}^T \tag{A3}$$

$$\mathbf{W}_{k-q:k} = \begin{bmatrix}\mathbf{w}_{\mathbf{k-q}}^{\mathbf{T}} & \mathbf{w}_{k-q+1}^{T} & \dots & \mathbf{w}_k^{T}\end{bmatrix}^T \tag{A4}$$

$$\mathbf{V}_{k-q:k} = \begin{bmatrix}\mathbf{v}_{\mathbf{k-q}}^{\mathbf{T}} & \mathbf{v}_{k-q+1}^{T} & \dots & \mathbf{v}_k^{T}\end{bmatrix}^T \tag{A5}$$

$$\mathbf{F}_{k-q:k} = \begin{bmatrix}\mathbf{f}_{\mathbf{k-q}}^{\mathbf{T}} & \mathbf{f}_{k-q+1}^{T} & \dots & \mathbf{f}_k^{T}\end{bmatrix}^T \tag{A6}$$

$$\boldsymbol{\mathcal{O}}_{k-q:k} = \begin{bmatrix}\mathbf{H}_{\mathbf{k-q}}\\ \mathbf{H}_{k-q+1}\boldsymbol{\Phi}_{k-q}\\ \vdots\\ \mathbf{H}_k\boldsymbol{\Phi}_{k-1}\dots\boldsymbol{\Phi}_{k-q}\end{bmatrix} \tag{A7}$$

$$\mathbf{Q}_{w,k-q:k} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{H}_{k-q+1}\boldsymbol{\Gamma}_{k-q} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{H}_{k-q+2}\boldsymbol{\Phi}_{k-q+1}\boldsymbol{\Gamma}_{k-q} & \mathbf{H}_{k-q+2}\boldsymbol{\Gamma}_{k-q+1} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & & \ddots & \vdots \\ \mathbf{H}_k\boldsymbol{\Phi}_{k-1}\dots\boldsymbol{\Phi}_{k-q+1}\boldsymbol{\Gamma}_{k-q} & \mathbf{H}_k\boldsymbol{\Phi}_{k-2}\dots\boldsymbol{\Phi}_{k-q+1}\boldsymbol{\Gamma}_{k-q+1} & \dots & \mathbf{H}_k\boldsymbol{\Gamma}_{k-1} & \mathbf{0} \end{bmatrix} \tag{A8}$$

The matrix $\boldsymbol{\mathcal{O}}_{k-q:k}$ is the discrete LTV observability matrix over a sliding window. This batch realization [extension of the linear time-invariant system in (Isermann, 2005)] formulation connects the stacked measurements $\mathbf{Z}_{k-q:k}$ to the past state vector $\boldsymbol{x}_{k-q}$ and its associated observability matrix $\boldsymbol{\mathcal{O}}_{k-q:k}$ nicely.

Batch realization is not unique; Joerger and Pervan (2013) define a different formulation where the past measurements are a function of all the past states. However, the measurement model in Joerger and Pervan (2013) is not explicitly expressed as a function of the observability matrix.

APPENDIX B: PROOF OF MATRIX EQUALITY

In this appendix, we prove the following equation is true:

$$(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)^T\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*) = \boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*) \tag{B1}$$

where $\boldsymbol{\mathcal{O}}$ is the observability matrix shown in Equation (A7), $\boldsymbol{\mathcal{O}}^*$ is left pseudoinverse of $\boldsymbol{\mathcal{O}}$ shown in Equation (21), and $\boldsymbol{\Sigma}$ is the weighting matrix shown in Equation (18). Note that $\boldsymbol{\Sigma}$ is symmetric by construction, that is, $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}^T$

Proof. First, we show $\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)$ is symmetric:

$$\begin{aligned} \left(\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)\right)^T &= (\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)^T\boldsymbol{\Sigma}^{-1} \\ &= (\mathbf{I} - (\boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)^T)\boldsymbol{\Sigma}^{-1} \\ &= (\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)\boldsymbol{\Sigma}^{-1} \\ &= \left(\mathbf{I} - \left(\left(\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}}\right)^{-1}\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\right)^T\boldsymbol{\mathcal{O}}^T\right)\boldsymbol{\Sigma}^{-1} \\ &= \left(\mathbf{I} - \boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}}(\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}})^{-T}\boldsymbol{\mathcal{O}}^T\right)\boldsymbol{\Sigma}^{-1} \\ &= \boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}}(\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}})^{-T}\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1} \\ &= \boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}}(\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}})^{-1}\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1} \\ &= \boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^* \\ &= \boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*) \end{aligned} \tag{B2}$$

where the 7th equality is achieved because $\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}}$ is also symmetric.

Now we have the following fact:

$$(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)\boldsymbol{\Sigma}^{-1} = \boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*) \tag{B3}$$

With the equality above, we can express Equation (B1) as follows:

$$\begin{aligned} (\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)^T\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*) &= (\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)^T(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)\boldsymbol{\Sigma}^{-1} \\ &= (\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)\boldsymbol{\Sigma}^{-1} \end{aligned} \tag{B4}$$

Now we proceed to prove the following: $(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T) = \mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T$.

This can be shown to be true by moving the left-hand side of equation to the right as follows:

$$\begin{aligned} (\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T) - (\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T) &= 0 \\ (\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)(\mathbf{I} - \mathbf{I} + \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T) &= 0 \\ (\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)(\boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T) &= 0 \end{aligned} \tag{B5}$$

We also realize the following equation is true:

$$\begin{aligned} \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T &= \left((\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}})^{-1}\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\right)^T\boldsymbol{\mathcal{O}}^T \\ &= \boldsymbol{\Sigma}^{-T}\boldsymbol{\mathcal{O}}(\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}})^{-T}\boldsymbol{\mathcal{O}}^T \\ &= \boldsymbol{\Sigma}^{-T}\boldsymbol{\mathcal{O}}(\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}})^{-1}\boldsymbol{\mathcal{O}}^T \\ &= \boldsymbol{\Sigma}^{-T}\boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^{-1}\boldsymbol{\Sigma}\boldsymbol{\mathcal{O}}^{-T}\boldsymbol{\mathcal{O}}^T \\ &= \mathbf{I} \end{aligned} \tag{B6}$$

Note that $(\boldsymbol{\mathcal{O}}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{O}})^{-1} = \boldsymbol{\mathcal{O}}^{-1}\boldsymbol{\Sigma}\boldsymbol{\mathcal{O}}^{-T}$ is justified because $\boldsymbol{\mathcal{O}}^T$ is right-invertible. However, the right inverse (not Moore-Penrose) of $\boldsymbol{\mathcal{O}}^T$ is not unique.

By substituting Equation (B6) into Equation (B5), we show that $(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T) = \mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T$ is indeed true. Finally, we arrive the matrix property that we want to prove:

$$\begin{aligned} (\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)^T\boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*) &= (\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*)^T(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)\boldsymbol{\Sigma}^{-1} \\ &= (\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)(\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)\boldsymbol{\Sigma}^{-1} \\ &= (\mathbf{I} - \boldsymbol{\mathcal{O}}^{*T}\boldsymbol{\mathcal{O}}^T)\boldsymbol{\Sigma}^{-1} \\ &= \boldsymbol{\Sigma}^{-1}(\mathbf{I} - \boldsymbol{\mathcal{O}}\boldsymbol{\mathcal{O}}^*) \end{aligned} \tag{B7}$$