Spring 1-1-2017

sUAS Wind Sensing with Computational Fluid Dynamics and a Distributed Flush Airdata System

Roger Jean Laurence III
University of Colorado at Boulder, rlaur001@gmail.com

Recommended Citation

Laurence III, Roger Jean, "sUAS Wind Sensing with Computational Fluid Dynamics and a Distributed Flush Airdata System" (2017). *Aerospace Engineering Sciences Graduate Theses & Dissertations*. 208.
https://scholar.colorado.edu/asen_gradetds/208

sUAS Wind Sensing with Computational Fluid Dynamics and a Distributed Flush Airdata System

by

Roger J. Laurence III

B.S., Mechanical Engineering, Florida International University, 2011

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Aerospace Engineering Sciences

2017

This thesis entitled:
sUAS Wind Sensing with Computational Fluid Dynamics and a Distributed Flush Airdata System
written by Roger J. Laurence III
has been approved for the Department of Aerospace Engineering Sciences

Professor Brian Argrow

Dr. de Boer

Professor Ahmed

Professor Doostan

Professor Farnsworth

Professor Frew

Professor Hamlington

Date ____________________

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

Laurence III, Roger J. (Ph.D., Aerospace Engineering Sciences)

sUAS Wind Sensing with Computational Fluid Dynamics and a Distributed Flush Airdata System

Thesis directed by Professor Brian Argrow

Small unmanned aircraft systems (sUAS) have proven their effectiveness for measuring both the inertial and aircraft-relative wind. Multiple methods for wind measurement from sUAS exist, but one of the more common instruments is the multi-hole probe (MHP). While the MHP is accurate and simple to use, there are two main drawbacks: 1) the MHP airdata system can cost several times that of the sUAS, and 2) the probe itself is often exposed to damage during routine operations. Flush airdata systems (FADS) are an alternative method of wind sensing, and work with pressure ports mounted flush with the aircraft surface. This removes any external components, thereby mitigating the risk of damage to the airdata system.

The work presented details the implementation of a FADS for sUAS. Computational fluid dynamics simulations were used to determine the port locations of the FADS. Airframe locations were sorted based on the total sensitivity over a range of angles of attack and sideslip. Upon completion of hardware installation, the FADS was calibrated in flight using an onboard MHP. A portion of the flight testing was reserved for validation of the FADS.

Multi-layer feedforward neural networks are employed to produce estimates of the angle of attack and sideslip, while static and stagnation ports on the fuselage measure airspeed. In the validation portion of flight tests, the FADS exhibited an overall mean error of 0.12 m/s in airspeed, but errors in angle of attack and sideslip were unbiased. Root-mean-square errors were 0.42 m/s, 0.65°, and 0.87°, respectively. Additionally, 97.7% of the errors in airspeed were within 1 m/s of the MHP, while 93.8% and 87.3% of the angle of attack and sideslip errors were within 1°. Flight tests show that a FADS can be calibrated in flight, and is an effective method for measuring the aircraft-relative wind from a small UAS.

Dedication

I'd like to dedicate this to my family, friends, all of my teachers throughout school and to everyone who has gladly shared their knowledge with me.

Acknowledgements

There are a large number of people that have helped me on the path to my Ph.D., be it with school and research, or helping me to really enjoy my time in Colorado and forget about the stress of graduate school. The biggest thank you needs to go to my advisor, Professor Brian Argrow, for all the help he has given me in graduate school. Many thanks to my family for pushing me and putting me in a position to succeed. All of my labmates (past and present) deserve acknowledgement, but especially Tevis Nichols, Will Silva, and Nick Campbell. Without the help from the rest of the RECUV/IRISS team, including James Mack, Will Finamore, Abhilash Manjunath, Thomas Lillis, Dan Hesselius, Steve Borenstein, Chris Choate and Chris Brown, I never would have been able to complete wind tunnel or flight tests. I'm thankful for the help from my committee and Professor Lawrence. I'm also grateful for the help provided by Jack Elston of Black Swift Technologies and Steven Semmer of NCAR. I'd also like to thank my aerospace friends for keeping me sane through graduate school, notably Tyler Mixa, Jordan Holquist, Jake Mashburn, and Greg Lucas; all the friends I met through the Colorado Mountain Club (notably Kahle and the Meowtaineers); and definitely my Colorado family of Chris Roath, Trevor DiMartino, Alex Watson, Scott Wieland, and Lindsey Moloznik, plus all the friends of the "Orange Light District" over the years. Finally, thanks to all my friends from back home, especially my climbing ones, for convincing me how great moving to Colorado would be.

I'd also like to acknowledge my various funding sources: US Air Force Office of Scientific Research (AFOSR; award FA9550-12-1-0412), the National Science Foundation (NSF; award AGS-1231096), the National Robotics Initiative (NRI; award IIS-1527919), and Black Swift Technologies.

Contents

Chapter

Appendix

Tables

Table

Figures

Figure

Chapter 1

Introduction

1.1 Motivation

Many different disciplines exhibit a need for accurate wind measurements. Wind measurements are used with weather prediction models for forecasting, to validate those models, and also improve understanding of weather phenomena. Knowing the typical wind field can inform the design of buildings or wind turbine farms. Aircraft also require at least the magnitude of the aircraft-relative wind for proper control.

When discussing measuring the inertial wind (what an Earth-relative stationary object would experience), aircraft offer many advantages, chief among which is the ability to make targeted *in situ* observations. When using aircraft, there are some unique differences between manned and unmanned aircraft for wind sensing. It is not currently feasible to integrate some of the instruments manned aircraft use, such as radar and lidar, into small unmanned aircraft systems (sUAS). However, sUAS are able to fly in locations where it may be unwise, impractical, or impossible for manned aircraft to fly. One such scenario is trying to sample severe weather at low altitudes.

Different methods for measuring the wind are available to sUAS, and they have proven very effective for wind sensing missions. Typical missions include sampling the atmospheric boundary layer [24] or measuring the wind in the wake of wind turbines [32, 82]. Additionally, UAS are gaining popularity for use in larger scale, multi-team deployments [12, 13]. The Tempest UAS is one platform that has proven its efficacy for meteorological missions, and is shown in Fig. 1.1.

Clearly, the need for sUAS to perform wind sensing exists (especially since it is possible

Figure 1.1: Kevin Rauhauser launching the Tempest UAS. Photo credit: Sean Waugh, NSSL.

to gather other *in situ* measurements at the same time). However, many of the wind sensing instruments are several times the cost of the sUAS, which negates the low-cost advantage of sUAS. This dissertation focuses on wind sensing from small UAS with distributed pressure sensors, in the form of a flush airdata system (FADS). Section 1.2 will give a background on some wind measurement methods for small UAS (including flush airdata systems), and Section 1.3 discusses potential errors wind sensing systems are susceptible to.

1.2 Wind Sensing

1.2.1 UAS Methods

Measurements from the Global Positioning System (GPS), inertial measurement unit (IMU), the aircraft dynamics model, and pitot-static probe can be fused together within extended Kalman filters to estimate the inertial wind [39]. It is also possible to estimate the wind using information from just the inertial navigation system (INS) and wind forecast [40]. Combining the inertial measurements with aircraft-relative wind instruments (such as angle of attack and sideslip vanes) can lead to improved accuracy [59].

Relative wind measurement techniques include multi-hole probes and flush airdata systems. A good summary of wind sensing techniques tailored to small UAS is found in [11]. Two FADS-like approaches that reduce the installation complexity include using thin hot-film flow sensors [15] or strips of pressure sensors on the wings [4]. The greatest advantage of both methods is that modifications to the airframe are unnecessary, thereby potentially reducing integration cost and time.

Currently, one of the simplest ways (in terms of installation) to measure the relative wind from small UAS is with the use of a multi-hole probe (MHP). Standard pitot-static probes only measure airspeed, while the angle of attack (α) and sideslip (β) are required for the full relative wind vector (though as mentioned previously, inertial measurements can be combined with pitot-static measurements to estimate the wind). Multi-hole probes are an adaptation of the classic pitot-static probe with additional ports surrounding the stagnation port [70]. These probes come in a variety of shapes, with different numbers of ports, and as small as 0.9 mm in diameter [75]. These probes can utilize a least squares approach with look up tables [77] or neural networks [56] for measuring the airspeed and flow angles (α and β). Compressible calibration allows for a wide usable speed range for these probes [29].

Multi-hole probe use with sUAS for meteorological observations can be found throughout the literature [58, 23], and careful system setup by the end user can lead to improvements in the quality of the measurements [83]. The accuracy, simplicity, and small size of the probes has led to their widespread use with sUAS. However, multiple disadvantages exist. Flow effects from the airframe can bias the measurements, and commercially available probes can be expensive (possibly several times the cost of a flight-ready UAS). Without landing gear or a smooth runway, the probes are susceptible to damage during takeoff and landing. A multi-hole probe mounted on the X-8 Skywalker is shown in Fig. 1.2, and it is easy to imagine the exposed probe being damaged during a rough landing. This is the primary motivation for moving away from multi-hole probes, and why

Figure 1.2: Photo of the Skywalker UAS coming in to land. Note the exposed probe above the nose. Photo credit: Joanie Wiesman, CU Boulder.

flush airdata systems are investigated in this dissertation.

1.2.2 Flush Airdata Systems

Aircraft commonly use pitot-static probes for determining the airspeed. This can be problematic for aircraft flying at supersonic or hypersonic speeds, partly due to increased aerodynamic and thermal loads. To address this, one of the first flush airdata systems was developed for the X-15 research aircraft [5]. This system used a hydraulically actuated sphere with four pressure ports: two in the angle of attack plane, and two in the sideslip plane. The sphere would rotate to null the differential pressure from the angle of attack and sideslip ports. The sphere's offset from center indicated the angle of attack and sideslip. Additional ports that measured stagnation and static pressure were used to determine Mach number and pressure altitude.

The National Aeronautics and Space Administration (NASA) was a pioneer in the field of FADS research. Early work was done with a variety of wind tunnel models, and was oftentimes concerned with subsonic or transonic speeds [34]. In-flight testing of FADS was performed on a variety of aircraft shapes, sizes, and speeds, and was implemented on the Space Shuttle [67], an F-14 [33], F-18 [78], the X-33 [81], and a KC-135A [35]. Multiple calibration approaches have been developed [7, 19]. Some of the calibration instruments, such as ground radar or large wind tunnels, are likely inaccessible to a large portion of teams implementing a FADS on sUAS. Calibrating against a reference boom, on the other hand, can be adapted for use with sUAS due to the availability of small multi-hole probes.

The hardware costs of a flush airdata system can be several times less than that of a multi-hole probe, especially with the low cost of mass produced pressure sensors. Additionally, there are no exposed parts with a flush airdata system, thereby reducing the risk of damage. The fact that sUAS typically fly slow enough to be in the incompressible range also greatly simplifies the calibration. An in-flight approach to calibration for small UAS is detailed in Chapter 4. This approach used a multi-hole probe during calibration flights, but not during normal operations. Therefore, only a single probe (or other reference instrument) is required to calibrate an entire fleet of aircraft.

While the relative wind from a flush airdata system provides useful information for the control of the aircraft, the inertial wind is required by meteorologists or when mapping the wind field (such as in the wake of wind turbines). Inertial sensors on the aircraft can be used to transform the relative wind (measured by a flush airdata system, multi-hole probe, etc) into the inertial wind. This transformation, along with some general wind sensing errors, are discussed in the next section.

1.3 Potential Wind Sensing Errors

1.3.1 Errors in Transformation to Inertial

Measuring the relative wind is only the first step towards computing the inertial winds. An aircraft state estimate[1] is also required to transform from the relative to the inertial wind. Because the state estimate accuracy requirements can be higher for inertial wind transformations than for the autopilot navigation, it can be a good idea to install a higher quality aftermarket INS. However,

---

[1] The aircraft state includes the instantaneous aircraft velocity, orientation, and rotation rates

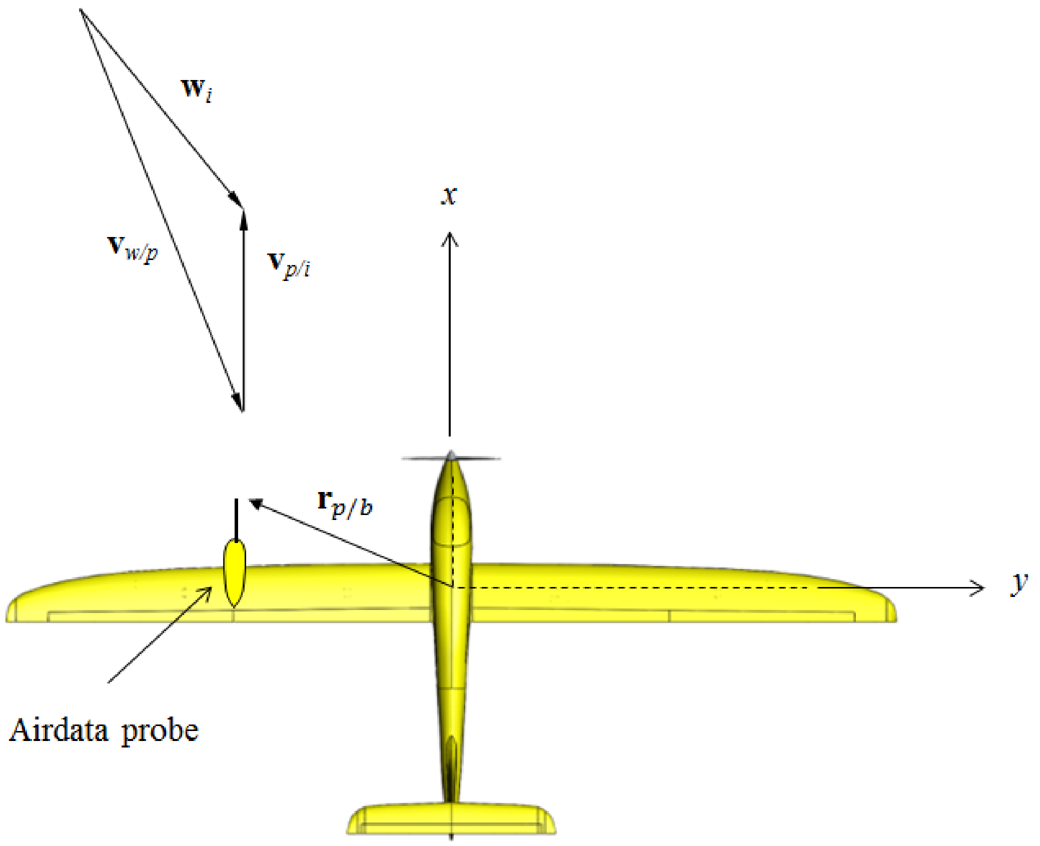


Figure 1.3: Diagram showing a typical setup with the Tempest UAS for relative wind sensing with a multi-hole probe.

the same issue of cost versus accuracy appears again when choosing the INS. High end units can cost orders of magnitude greater than the sUAS, thereby minimizing their use.

Errors in the state estimate will appear in the final inertial transformation, with a more thorough explanation given by [49] (another good error reference is [10]). This means the accuracy requirements for the relative wind sensor depends on both the user requirements and the state estimator used. Some error will be present in the inertial wind transformation that is dependent upon the state estimate; this leads to a point of diminishing returns on the accuracy of the relative wind sensor. The general setup for using a multi-hole probe for relative wind sensing can be seen in Fig. 1.3. The inertial wind is broken down into three components:[2]

---

[2] Adapted from Frew [17]

$$\mathbf{w}_i = \mathbf{v}_{b/i} + \mathbf{v}_{p/b} + \mathbf{v}_{w/p} \tag{1.1}$$

where $\mathbf{w}_i$ is the wind relative to the inertial frame, $\mathbf{v}_{b/i}$ is the velocity of the body relative to the inertial frame, $\mathbf{v}_{p/b}$ is the velocity of the probe relative to the body and $\mathbf{v}_{w/p}$ is the velocity of the wind relative to the probe. The relative wind measured by the probe in the body frame can be found through the following transformation:

$$\mathbf{v}^b_{w/p} = \begin{bmatrix} \cos\,\alpha & 0 & -\sin\,\alpha \\ 0 & 1 & 0 \\ \sin\,\alpha & 0 & \cos\,\alpha \end{bmatrix} \begin{bmatrix} \cos\,\beta & \sin\,\beta & 0 \\ -\sin\,\beta & \cos\,\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -\mathrm{V_a} \\ 0 \\ 0 \end{bmatrix} \tag{1.2}$$

where α is the angle of attack, β is the side slip and $\mathrm{V_a}$ is the airspeed. The coordinate system defining α, β, and $\mathrm{V_a}$ is shown in Fig. 1.4 (with $\mathrm{V_a}$ equivalent to the magnitude of $\overline{\mathrm{V}}$ in the image). Assuming rigid body rotation, the following expression is obtained for the velocity of the probe relative to the body:

$$\mathbf{v}^b_{p/b} = \mathbf{\Omega}^b_{b/i} \times \mathbf{r}_{p/b} \tag{1.3}$$

with $\mathbf{\Omega}^b_{b/i}$ representing the angular rates about the center of mass and $\mathbf{r}_{p/b}$ the position offset of the probe from the center of mass. The rotation into the inertial frame is:

$$R_{i/b} = \begin{bmatrix} \cos\,\psi & -\sin\,\psi & 0 \\ \sin\,\psi & \cos\,\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\,\theta & 0 & \sin\,\theta \\ 0 & 1 & 0 \\ -\sin\,\theta & 0 & \cos\,\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\,\phi & -\sin\,\phi \\ 0 & \sin\,\phi & \cos\,\phi \end{bmatrix} \tag{1.4}$$

with ψ, θ and ϕ representing yaw, pitch, and roll. Roll, pitch, and yaw follow the standard convention, and the definition can be found in [2] (or other flight mechanics books). Rearranging Eq. 1.1 to solve for the inertial wind yields:

$$\mathbf{w}^i_i = \mathbf{v}^i_{b/i} + R_{i/b}\left(\mathbf{\Omega}^b_{b/i} \times \mathbf{r}_{p/b} + \mathbf{v}^b_{w/p}\right) \tag{1.5}$$

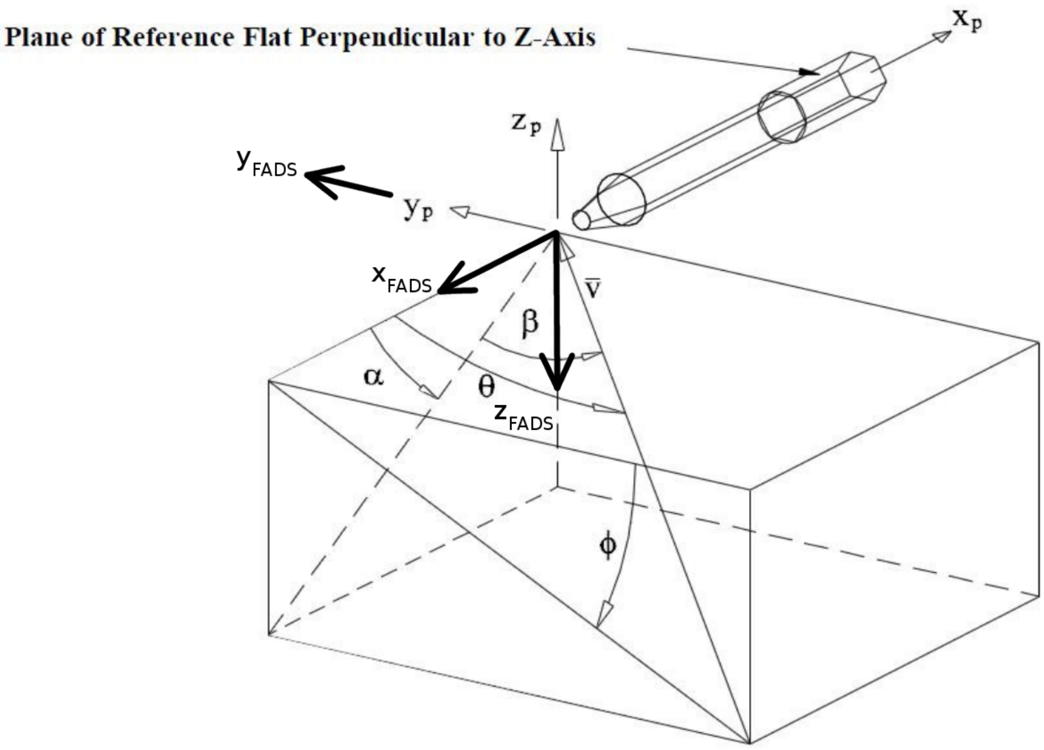


Figure 1.4: FADS coordinate system superimposed on the Aeroprobe coordinate system [1].

where $\mathbf{v}_{b/i}^{i}$ is the inertial velocity of the aircraft relative to the inertial frame. Other factors that affect the probe velocity (such as aircraft flexing) are excluded for simplicity. It is possible to use Eq. 1.5 in a simple sensitivity analysis to see how the different accuracies of the state estimate and relatives wind will affect the inertial winds. The inertial wind equation is actually three equations (x, y and z) and has numerous variables in it. By taking the Jacobian of $\mathbf{w}_{i}^{i}$, it is possible to see how sensitive the errors in the inertial winds are to errors in the state estimate and relative winds. The sensitivity to a few parameters is shown in Fig. 1.5. There is minimal effect from the roll angle, but pitch and yaw do play vital roles in influencing the inertial wind calculation. The same is true for α and β.

Now that the sensitivities are known for each parameter of interest, it is possible to take a first-order look at how the errors in the magnitude of the inertial winds are affected by errors in the relative wind for a chosen inertial navigation system. Several of the Jacobian terms are at least an order of magnitude smaller than the largest terms, and are therefore neglected. The inertial

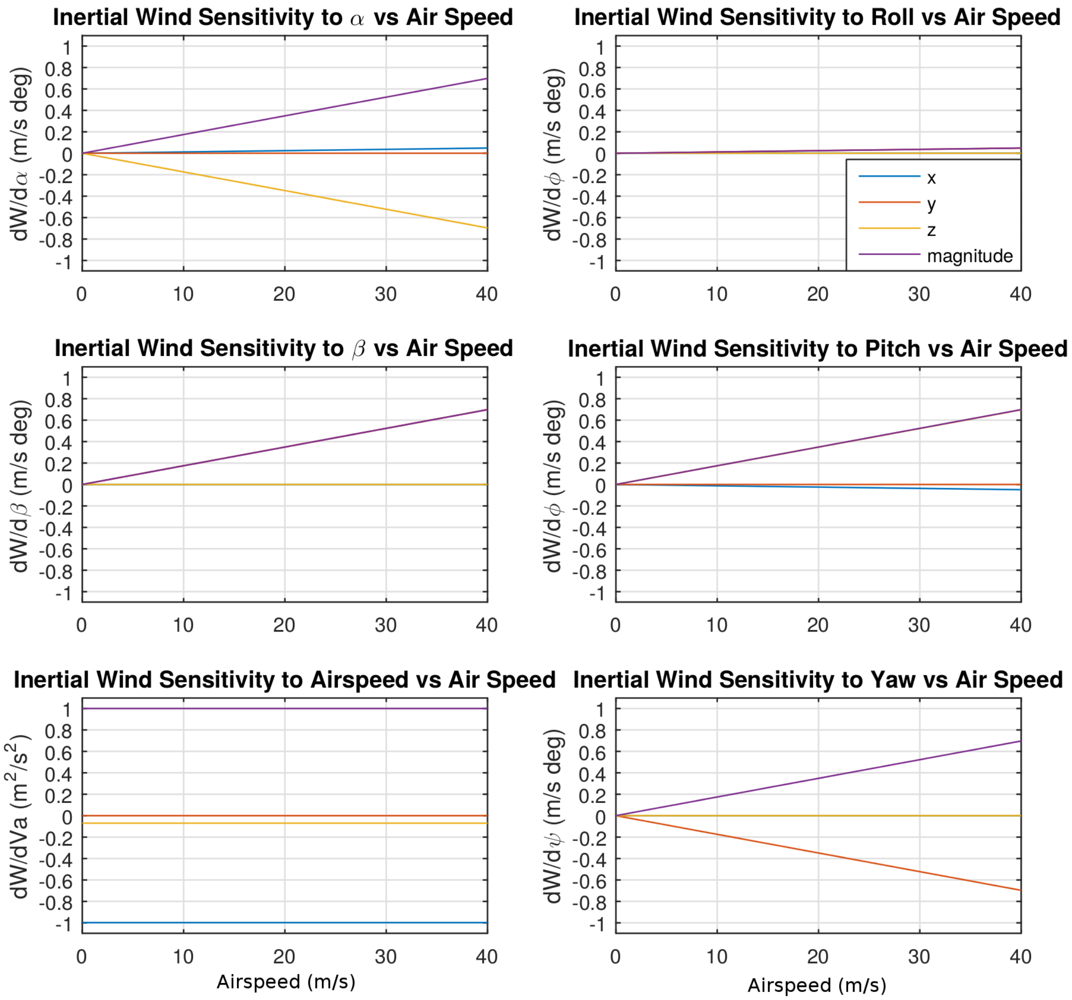


Figure 1.5: Sensitivity of the inertial winds to several of the parameters that are used in the inertial wind transformation. Roll, pitch, and yaw are set to 0°, with the position offset at [30,0,5] cm. The aircraft has an inertial velocity of $[10, 10, 0]$ m/s.

wind equation was formulated using a multi-hole probe, so when analyzing the sensitivity for a distributed pressure sensor system, the rigid body rotation term is ignored (for simplicity). It is also assumed that the errors in the inputs are zero-mean Gaussian. Using the method of combining normal distributions to group the individual component errors into magnitude errors yields the following equation for a first-order investigation into the errors in the inertial wind magnitude:

$$\sigma_w \approx \sqrt{3\sigma_v^2 + \left(\frac{\partial W}{\partial \psi}\sigma_\psi\right)^2 + \left(\frac{\partial W}{\partial \theta}\sigma_\theta\right)^2 + \left(\frac{\partial W}{\partial V_a}\sigma_{V_a}\right)^2 + \left(\frac{\partial W}{\partial \alpha}\sigma_\alpha\right)^2 + \left(\frac{\partial W}{\partial \beta}\sigma_\beta\right)^2} \tag{1.6}$$

where σ_w is the standard deviation of the errors in the inertial wind magnitude, σ_ψ is the standard deviation of the error in yaw, σ_θ is the standard deviation of the error in pitch, σ_{V_a} is the standard deviation of the error in the airspeed and σ_α and σ_β are the standard deviation of the error in α and β (for simplicity it is assumed they are the same).

The relation between σ_w and a range of values for σ_α and σ_β is seen in Fig. 1.6. The errors in the state estimate are set to the stated errors of the VectorNav VN-200 [73, 74], an inertial navigation system that is commonly flown by the Research and Engineering Center for Unmanned Vehicles (RECUV). Using σ_{V_a} of 1 m/s (based on a search of air-data probes suitable for sUAS use), there is no real advantage in being able to get $\sigma_{\alpha\beta}$ below about 0.5° (where $\sigma_{\alpha\beta}$ is the standard deviation of the error in α and β). This is the point of diminished returns. For a less accurate INS, this point of diminished returns would be shifted towards higher values of $\sigma_{\alpha\beta}$. That is, the errors in the state estimate create a minimum error bound, and there is a point at which further increases in accuracy of the relative wind sensing system will no longer decrease σ_w. Fig. 1.6 will change based on the INS system chosen and on the airspeed accuracy. It is assumed, though, that the values used are representative of systems that will be employed during flight tests of the distributed pressure sensor project. Because of these factors, and that commercial MHP systems offer comparable accuracy, the goal of this dissertation is to achieve an accuracy of 1 m/s in airspeed and 1° in α and β.

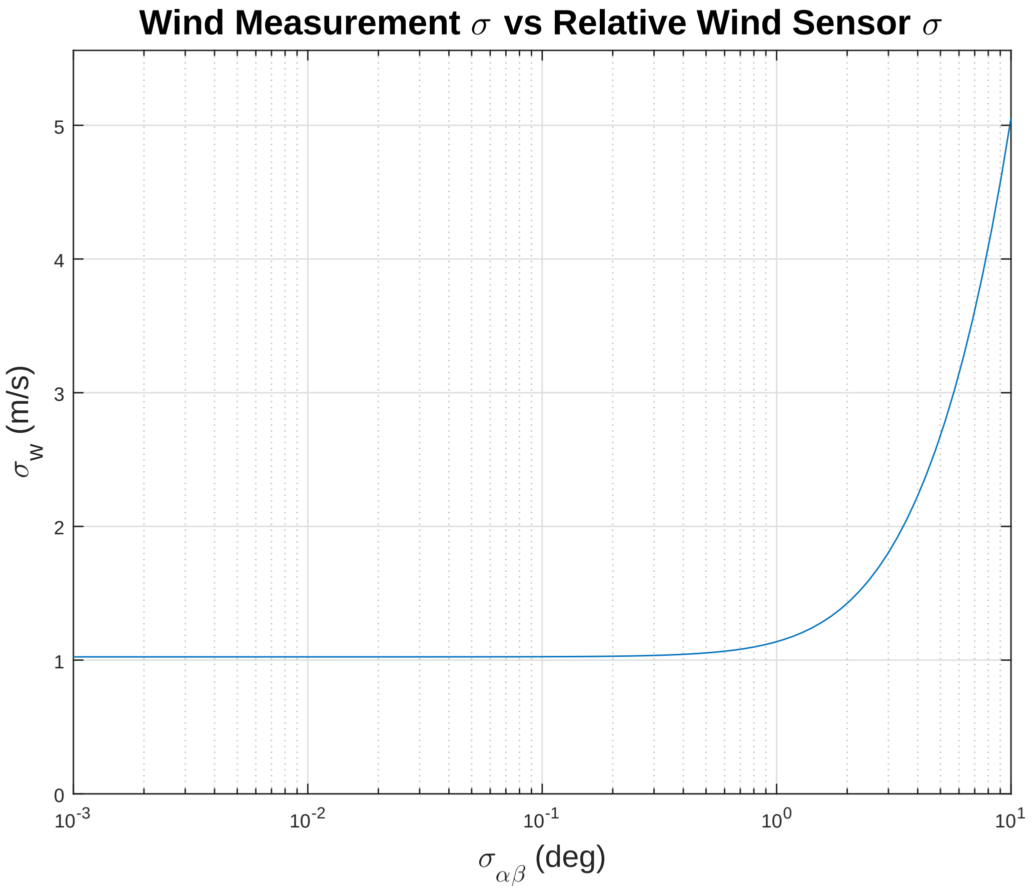


Figure 1.6: Plot showing the relation between the standard deviation of errors in the magnitude of the inertial winds and the standard deviation of errors in α and β.

1.3.2 Pressure and Temperature Errors

The true airspeed (TAS) is related to the indicated airspeed (IAS) through the following equation:

$$\mathrm{TAS} = \mathrm{IAS}\sqrt{\frac{\rho_0}{\rho}} \tag{1.7}$$

with ρ_0 representing the density at sea level, and ρ representing the actual density. The indicated airspeed is the airspeed from the airdata system (pitot-static probe, MHP, FADS, etc) using sea level density. Define the percent error in true airspeed ($\% \ \varepsilon_{\mathrm{TAS}}$) as:

$$\% \ \varepsilon_{\mathrm{TAS}} = \frac{\mathrm{TAS_T} - \mathrm{TAS}}{\mathrm{TAS}} \tag{1.8}$$

where the subscript T specifies the true value, and no subscript represents the measured value. Substituting Eq. 1.7 into Eq. 1.8 yields the following equation:

$$\% \ \varepsilon_{\mathrm{TAS}} = \frac{\mathrm{IAS_T}\sqrt{\frac{\rho_0}{\rho_\mathrm{T}}} - \mathrm{IAS}\sqrt{\frac{\rho_0}{\rho}}}{\mathrm{IAS}\sqrt{\frac{\rho_0}{\rho}}} \tag{1.9}$$

Concerning ourselves only with the errors related to the density correction for TAS (i.e. $\mathrm{IAS_T} = \mathrm{IAS}$), Eq. 1.9 then simplifies to:

$$\% \ \varepsilon_{\mathrm{TAS}} = \sqrt{\frac{\rho}{\rho_\mathrm{T}}} - 1 \tag{1.10}$$

Substituting the Ideal Gas Law

$$\rho = \frac{\mathrm{P}}{\mathrm{RT}} \tag{1.11}$$

into Eq. 1.10 and simplifying yields:

$$\% \ \varepsilon_{\mathrm{TAS}} = \sqrt{\frac{\mathrm{PT_T}}{\mathrm{P_T T}}} - 1 \tag{1.12}$$

Eq. 1.12 shows the relation between errors in temperature and pressure to the percent error in the true airspeed. This equation can be used to calculate the maximum error in the true airspeed based on the specifications of the temperature and pressure sensors being used. The MS8607 [68] sensor from TE Connectivity is a low cost MEMS style sensor that is being used throughout this project. According to the manual, the maximum error in the absolute pressure is 2 mbar (at 820 mbar, which is a typical flight pressure) while the maximum error in temperature is 1 °C (at 25 °C). Eq. 1.12 therefore predicts a maximum error of approximately 0.3% in the density correction for the true airspeed. The maximum expected errors in temperature and pressure will have a negligible effect on the true airspeed error. However, it should be noted that this error only applies to the density correction from IAS to TAS, and any errors in IAS will also translate to errors in TAS.

It is also important to note that not all airdata systems have access to a temperature sensor, and may use the Standard Atmosphere [47] as a way to perform the density correction from IAS to TAS. A useful way of looking at the effects of the temperature errors is investigating the sensitivity of the percent error to changes in the temperature error. First, the measured temperature is defined in terms of the true temperature, plus an error term, $\Delta \mathrm{T}$:

$$\mathrm{T} = \mathrm{T_T} + \Delta \mathrm{T} \tag{1.13}$$

Combining Eq. 1.13 with Eq. 1.12, and taking the first derivative with respect to $\Delta \mathrm{T}$, yields a first-order look at the sensitivity, as shown in Eq. 1.14:

$$\frac{\partial \varepsilon_{\mathrm{TAS}}}{\partial \Delta \mathrm{T}} = -\frac{\sqrt{\mathrm{T_T}}}{2\sqrt{(\mathrm{T_T} + \Delta \mathrm{T})^3}} \left(\sqrt{\frac{\mathrm{P}}{\mathrm{P_T}}} \right) \tag{1.14}$$

Similar to Eq. 1.13, P can be written in terms of $\mathrm{P_T}$ and the error in the pressure measurement, $\Delta \mathrm{P}$. For the MS8607 sensor, $\Delta \mathrm{P_{max}} \ll \mathrm{P_T}$, which suggests $\sqrt{\frac{\mathrm{P}}{\mathrm{P_T}}} \approx 1$. Additionally, with the assumption that $\Delta \mathrm{T_{max}} \ll \mathrm{T_T}$, and only focusing on the error magnitudes, Eq. 1.14 reduces to:

$$\left| \frac{\partial \varepsilon_{\mathrm{TAS}}}{\partial \Delta \mathrm{T}} \right| \approx \frac{1}{2\mathrm{T_T}} \tag{1.15}$$

This shows that the sensitivity of the percent error due to the error in the temperature measurement can be approximated using only the true temperature. At a true temperature of 20 °C, this sensitivity is 0.17 %/°C. Clearly the expected temperature errors from the sensors in use will have a negligible effect on the TAS calculation. However, using the Standard Atmosphere can lead to large TAS errors. For example, during the summer, it is possible for the true temperature to reach 30 °C or greater in the eastern plains of Colorado, while the Standard Atmosphere predicts a temperature of only 5 °C. This difference of 25 °C will lead to an error of approximately 4% in the TAS calculation. Therefore, it is important to have a temperature sensor when performing the TAS calculation, but it does not need to be exceptionally accurate.

With the percent error in density defined in the same manner as Eq. 1.8, simplification leads to:

$$\%\ \varepsilon_\rho = \frac{\mathrm{P_T T}}{\mathrm{P T_T}} - 1 \tag{1.16}$$

Substituting Eq. 1.13 into Eq. 1.16, and taking the derivative, yields the sensitivity of $\%\ \varepsilon_\rho$ to errors in the true temperature:

$$\frac{\partial \varepsilon_\rho}{\partial \Delta \mathrm{T}} = \frac{\mathrm{P_T}}{\mathrm{P T_T}} \approx \frac{1}{\mathrm{T_T}} \tag{1.17}$$

At 20 °C, the sensitivity is only 0.34 %/°C. The pressure measured at the ports of a FADS will vary based on density, so it is important to have a density measurement. Similar to the TAS case, it is important to have the ability to measure temperature (as opposed to using the Standard Atmosphere), but the errors in the temperature measurements will only have a small effect on the density errors.

As mentioned previously, the MS8607 sensor has maximum specified errors of 2 mbar and 1 °C (at 820 mbar and 25 °C, respectively). The definition of the percent error in true airspeed is kept the same, and is shown in Eq. 1.9. Using the maximum stated sensor errors and the ideal gas law, $\sqrt{\frac{\rho}{\rho_\mathrm{T}}} = 1.006$. With the maximum expected errors, the measured density will only be off by 0.6%; therefore, $\rho \approx \rho_\mathrm{T}$. Making the assumption that $\rho = \rho_\mathrm{T}$, Eq. 1.9 now simplifies to:

$$\%\ \varepsilon_\mathrm{T} = \frac{\mathrm{IAS_T} - \mathrm{IAS}}{\mathrm{IAS}} \tag{1.18}$$

The error in the indicated airspeed, ΔIAS, is defined as:

$$\Delta\mathrm{IAS} = \mathrm{IAS_T} - \mathrm{IAS} \tag{1.19}$$

Combining the definition of ΔIAS with Eq. 1.18 yields the following form of the percent error:

$$\%\ \varepsilon_\mathrm{T} = \frac{\Delta\mathrm{IAS}}{\mathrm{IAS_T} - \Delta\mathrm{IAS}} \tag{1.20}$$

Taking the derivative of Eq. 1.20 shows how sensitive the percent error in TAS is to changes in IAS error:

$$\frac{\partial \varepsilon_{\mathrm{T}}}{\partial \Delta \mathrm{IAS}} = \frac{\mathrm{IAS}_{\mathrm{T}}}{(\mathrm{IAS}_{\mathrm{T}} - \Delta \mathrm{IAS})^2} \tag{1.21}$$

In general, it is not possible to assume $\Delta\mathrm{IAS} \ll \mathrm{IAS}_{\mathrm{T}}$. As an example, the Skywalker UAS has a flight speed on the order of 20 m/s, while the Aeroprobe MHP has a specified maximum error of 1 m/s. Assuming IAS = 20 m/s and ΔIAS = 1 m/s, the sensitivity of the percent error in TAS to errors in IAS is 5.5 $\%/\frac{\mathrm{m}}{\mathrm{s}}$. Based on these results, errors in the IAS, as opposed to errors in the temperature sensor, will likely lead to larger errors in the TAS. This underscores the importance of an accurate airspeed measurement, as the true airspeed is required during for the transformation to the inertial wind.

1.4 Conclusion

Small unmanned aircraft offer many unique advantages for performing *in situ* measurements, and their use has become more widespread in recent years. A range of wind sensing methods for UAS exist: from direct methods that combine the dynamics of the aircraft with inertial sensors and pitot-static probes, to flush airdata systems and multi-hole probes whose relative wind measurements can be transformed into the inertial frame. Many of the original calibration instruments for flush airdata systems are either infeasible or inappropriate when the airdata system is installed on an sUAS. This is due in part to the relative simplicity of sUAS, especially when limited to slow airspeeds in the incompressible range. When examining wind sensing errors, it was determined that it is important to have a reference temperature sensor; however, it does not need to be exceptionally accurate. The errors in density and TAS are not that sensitive to inaccuracies in the temperature measurement. Errors in the indicated airspeed are likely to play a larger role.

The motivation for developing the distributed flush airdata system came from experience with multi-hole probes and the advantages offered by flush airdata systems. The method presented in this dissertation allows for the flush airdata system to be distributed across the entire airframe, and describes an approach for determining port locations through the use of computational fluid dynamics. Investigating sources of error when transforming to the inertial frame and comparisons with commercial systems led to the goal of 1 m/s in airspeed and 1° in angle of attack and sideslip. The system presented uses a reference multi-hole probe to perform an in-flight calibration. This calibration approach allows a single reference instrument to be used to calibrate an entire fleet of aircraft with flush airdata systems.

The contribution of the work presented in this dissertation includes the development of a method for determining port locations with arbitrary airframe shapes and the calibration of a flush airdata system for small UAS in flight. The port selection method is described in Chapter 3, along with the process of integrating the hardware into a new airframe in preparation of flight testing. Results from these flight tests are examined in Chapter 4, along with a discussion of the in-flight calibration. A summary of the work performed and results obtained, along with a discussion of suggested improvements to the method as described, appear in Chapter 5. Early work involving the distributed flush airdata system is discussed in Chapter 2, along with new results comparing neural networks to the original nonlinear least squares approach.

Chapter 2

Wind Tunnel Results for a Distributed Flush Airdata System

2.1 Introduction

Small unmanned aircraft systems (sUAS) have proven effective at making *in situ* meteorological measurements within the atmospheric boundary layer (e.g., [24, 3, 58]), including operations in, and near, supercell thunderstorms [12, 61]. This demonstrated ability to operate in both quiescent and dynamic environments uniquely qualifies sUAS for *in situ* wind measurements. Size, weight, and power requirements preclude the integration of some proximal wind-measurement tools used on manned aircraft, such as radar, into an sUAS. Many other approaches have been explored, such as one that uses only the inertial sensors already in use for control of the aircraft [44], a strip of pressure sensors on the wing [4], and multi-hole probes, which are a modification of the classic pitot-static probe to enable three-dimensional wind measurements [70, 29].

The multi-hole probe (MHP) has been used effectively for wind measurements from sUAS [23, 32, 72]. It is highly accurate for relative-wind measurements (errors less than 1 m/s airspeed and 1° angle of attack and sideslip), and miniaturized commercial versions that can be integrated into sUAS are available [1]. By combining the aircraft state with the relative wind measurement (from an MHP or some other relative wind instrument), it is possible to estimate the inertial winds (for example, in the North-East-Down frame). The aircraft state is generally estimated with an onboard inertial measurement unit (IMU) that provides an estimate of the instantaneous orientation and rotation rates coupled with a GPS receiver that provides the inertial position and velocity of the aircraft. A description of this process is presented in [49].

Figure 2.1: Aeroprobe MHP mounted on the X-8 Skywalker UAS.

As the Aeroprobe MHP is a commercial off the shelf (COTS) system, it is effectively "plug-and-play" where the user is responsible for installing the probe in a suitable location on the aircraft, but is not required to modify the software. However, this simplicity means that an MHP with an accompanying airdata computer can cost several times that of the sUAS airframe on which it is being flown. Additionally, the MHP must be mounted such that it has access to the freestream (typically sticking out from the nose), which can make it vulnerable to damage, especially on sUAS without landing gear (real world examples includes clipping a wing in tall grass or skipping on a rock upon landing, both of which can lead the aircraft to impact the ground probe first). One configuration that has been used previously is shown in Fig. 2.1.

Flush airdata systems (FADS) are an alternative to the MHP that rely on the same physical principles. FADS effectively turn part of the airframe into an MHP. They have been implemented on a range of aircraft, from the Space Shuttle [36], to an F-18 [80], to other manned aircraft [71, 30, 31], and even on an sUAS [54]. All of these FADS have been located on the nose of the aircraft, which is not always a viable option for sUAS due to the wide variety of shapes [11]. While a FADS also requires significantly more user input than a COTS MHP, it does remove the need for an exposed probe.

This chapter presents a modification to the “conventional” FADS configuration: rather than restrict the pressure ports on the nosecone, the ports are distributed across the aircraft. A method is developed to determine suitable locations for an arbitrarily shaped fixed-wing aircraft. With the use of small, MEMS-style COTS pressure sensors,[1] this method also reduces the hardware cost by an order of magnitude compare to the MHP. While the selection of sensor locations is not the focus of this paper, Section 2.2 will provide a short background on how the locations were chosen. Section 2.3 details the wind tunnel model and how it was calibrated. Finally, Section 2.4 presents the results[2] and highlights how nonlinear least squares and neural networks were used to produce estimates of the angle of attack and sideslip from the pressure measurements, concluding with a comparison of the accuracy of the two estimation methods.

2.2 Sensor Location Selection

Both [80] and [54] show how potential flow can be used to determine the layout of the pressure ports on a nosecone. However, not all sUAS have nosecones (one such example shown in Figs. 2.2 and 2.3), and of the ones that do, the nosecone could be rendered unusable if the aircraft has a propeller installed on the nose. Additionally, without an *a priori* analysis, there is no guarantee that the nosecone is the optimal location for a FADS, particularly for unconventional airframe designs. Due to the variety of sUAS shapes, there is a need for a robust method of sensor location selection. A brief background on how computational fluid dynamics (CFD) can be used is provided here; [38] provides a more detailed discussion.

The Eagle Owl sUAS (Figs. 2.2 and 2.3) is used here because of its unconventional, but simple, geometry. STAR-CCM+[3] was used to predict how the pressure on the surface changes with varying angle of attack (α; defined as pitching up relative to the oncoming airflow) and sideslip (β; defined as yawing to the right relative to the oncoming airflow). This enables identification of the locations

---

[1] Microelectromechanical systems (MEMS); in the case of the MS5611-01BA, a piezo-resistive sensor for pressure [69]

[2] The results presented in this chapter are a more thorough update to the results presented in [37]

[3] CFD simulation software [66]

with highest sensitivity to changes in the wind angles. Steady-state simulations were run with α ranging from $-5°$ to $15°$ (in $1°$ increments), β ranging from $0°$ to $5°$ (in $1°$ increments), and the airspeed fixed at 15 m/s. The reference pressure for the simulations was set to 830 hPa (roughly the pressure at ground level in Boulder, Colorado, the location of the wind tunnel experiments) and the temperature was set to 20 °C. An ideal gas was assumed, and the K-ϵ model (a description of which can be found in [45]) was used to simulate turbulence. The simulation model contained approximately 185,000 grid points on the surface of the Eagle Owl, with the largest dimension of the cells being in the 1-4 mm range.

High priority was placed on aircraft locations that had a large range in pressure (the sensor noise/bias would therefore have less of an effect on the estimate of α and β), as well as locations that experienced a smooth/predictable pressure response to variations in α and β. To prioritize these two aspects, a simple cost function was implemented:

$$\mathrm{J_1} = \mathrm{RMSE}/\mathrm{P_r} \tag{2.1}$$

where RMSE is the root mean squared error of the fit between pressure, α, and β, and $\mathrm{P_r}$ is the difference of the maximum and minimum pressure experienced over all 126 simulations (for a specified grid point). The value of this cost function can be seen for all the grid points in Fig. 2.2 (a). A lower value implies either a more predictable response, or a greater pressure range. The leading edges of the top airfoil are where $\mathrm{J_1}$ is lowest, followed by the leading edges of the bottom airfoil. These locations are insensitive to changes in β (compared to changes in α), which makes them poorly suited for the determination of sideslip (this behavior was apparent in the wind tunnel results as well). A second cost function was then produced, with a greater emphasis on sensitivity to β:

$$\mathrm{J_2} = -\left(\frac{\mathrm{RMSE} - \mathrm{RMSE_l}}{\mathrm{RMSE_u} - \mathrm{RMSE_l}}\right) \times 0.2 + \left(\frac{\mathrm{P_{r_\beta}} - \mathrm{P_{r_{\beta_{min}}}}}{\mathrm{P_{r_{\beta_{max}}}} - \mathrm{P_{r_{\beta_{min}}}}}\right) \times 0.8 \tag{2.2}$$

where $\mathrm{P_{r_\beta}}$ is the average range of pressure across the sideslips (averaged over all β for a given

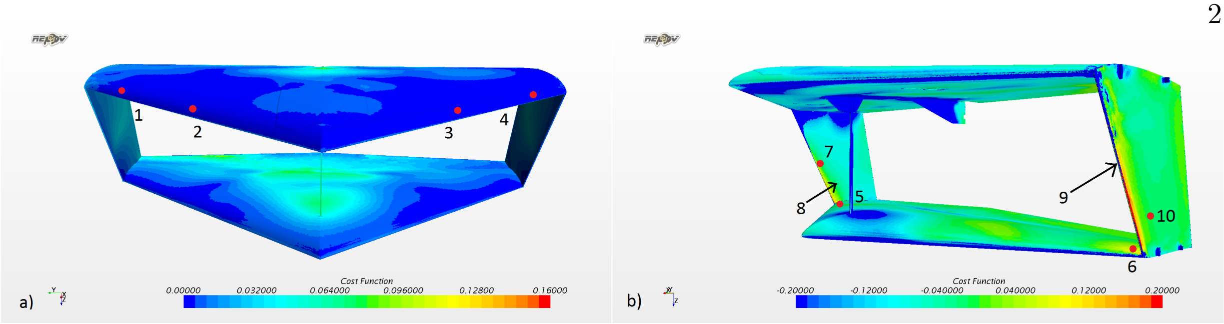


Figure 2.2: Plots showing the values of the two cost functions used for the selection of pressure port locations. Locations for the J_1 cost function (to be minimized) are shown in (a) and for the J_2 cost function (to be maximized) in (b). Arrows show approximate locations of sensors that are hidden by the current view.

α), $Pr_{\beta_{min}}$ and $Pr_{\beta_{max}}$ are the minimum and maximum average sideslip ranges, and $RMSE_l$ and $RMSE_u$ are the lower and upper limits of acceptable RMSE values. The lower limit $RMSE_l$ was set to 0.0609 as that was the lowest value achieved, while $RMSE_u$ was set to 1.0 as an upper limit. The cost function has larger values for locations that are better suited for determining sideslip (Fig. 2.2 (b)). The leading edges of the side plates are best suited for measuring sideslip. Using both of these cost functions, and excluding certain areas (such as the underside of the aircraft and the extreme forward portions of the leading edges), ten locations were chosen and are superimposed on the cost functions in Fig. 2.2. Since locations 1-4 were chosen with J_1, it is predicted that these locations will have the greatest influence on the measurement of α, with locations 5-10 (chosen using J_2) being the most important to determining β.

2.3 Wind Tunnel Tests

2.3.1 Wind Tunnel Model

To investigate the feasibility of a distributed FADS for obtaining wind measurements, a 2/3-scale model of the Eagle Owl airframe was fabricated and used in the low speed wind tunnel at the National Center for Atmospheric Research (NCAR) in Boulder, Colorado. The model has a wingspan of 0.6 m and a height of 0.2 m. It is constructed from EPS foam, bass wood, and epoxy. The nearly completed model can be seen in Fig. 2.3 (a), while the finished model can be seen

Table 2.1: Technical specifications of MS5611-01BA pressure sensor

Operating range (pressure)	10 to 1200 mbar
Operating range (temperature)	-40 to +85 °C
Accuracy (25 °C, 750 mbar)	-1.5 to +1.5 mbar
Error band (-20 °C to 85 °C, 450 to 1100 mbar)	-2.5 to +2.5 mbar

mounted in the wind tunnel in Fig. 2.3 (b). For these experiments, the pressure was recorded with Measurement Specialties MS5611-01BA pressure sensors [69], with the technical specifications reported in Table 2.1. These sensors were chosen for their low cost,[4] high resolution, and availability.

For simplicity, the pressure sensors were housed off the aircraft and outside the wind tunnel. Holes were drilled to allow 3.175 mm outer diameter Tygon tubing to run from the underside/interior face of the wings/side plates to the locations chosen through CFD. Slots were cut into the undersides/interior face to route the tubing while maintaining a smooth surface to minimize flow disturbance. A wooden spar was installed along the centerline of the lower wing; this connected to a carbon fiber rod which extended the model in front of the vertical support in the wind tunnel. The final step involved running the tubing cleanly off the model and out of the wind tunnel.

2.3.2 Calibration

The first phase of the experiment involved collecting data for calibration. As mentioned previously, only α and β were varied in the tunnel. The range of (α,β) orientations is shown in Fig. 2.4, with the blue dots representing orientations used for calibration and red dots for validation. For each orientation, 300 samples were taken, where a “sample” refers to a pressure measurement taken simultaneously from all 10 sensors.

It is assumed that the airflow through the tunnel is parallel to the walls through the test section. Therefore, all α measurements are relative to the floor of the test section. For clarity, α is defined as the angle between the wingtip chord line of the top wing and the flow through the tunnel, with the positive direction being nose up. An AccuMaster 7434 digital inclinometer was

---

[4] The cost is on the order of $10 per sensor, as of July 2017

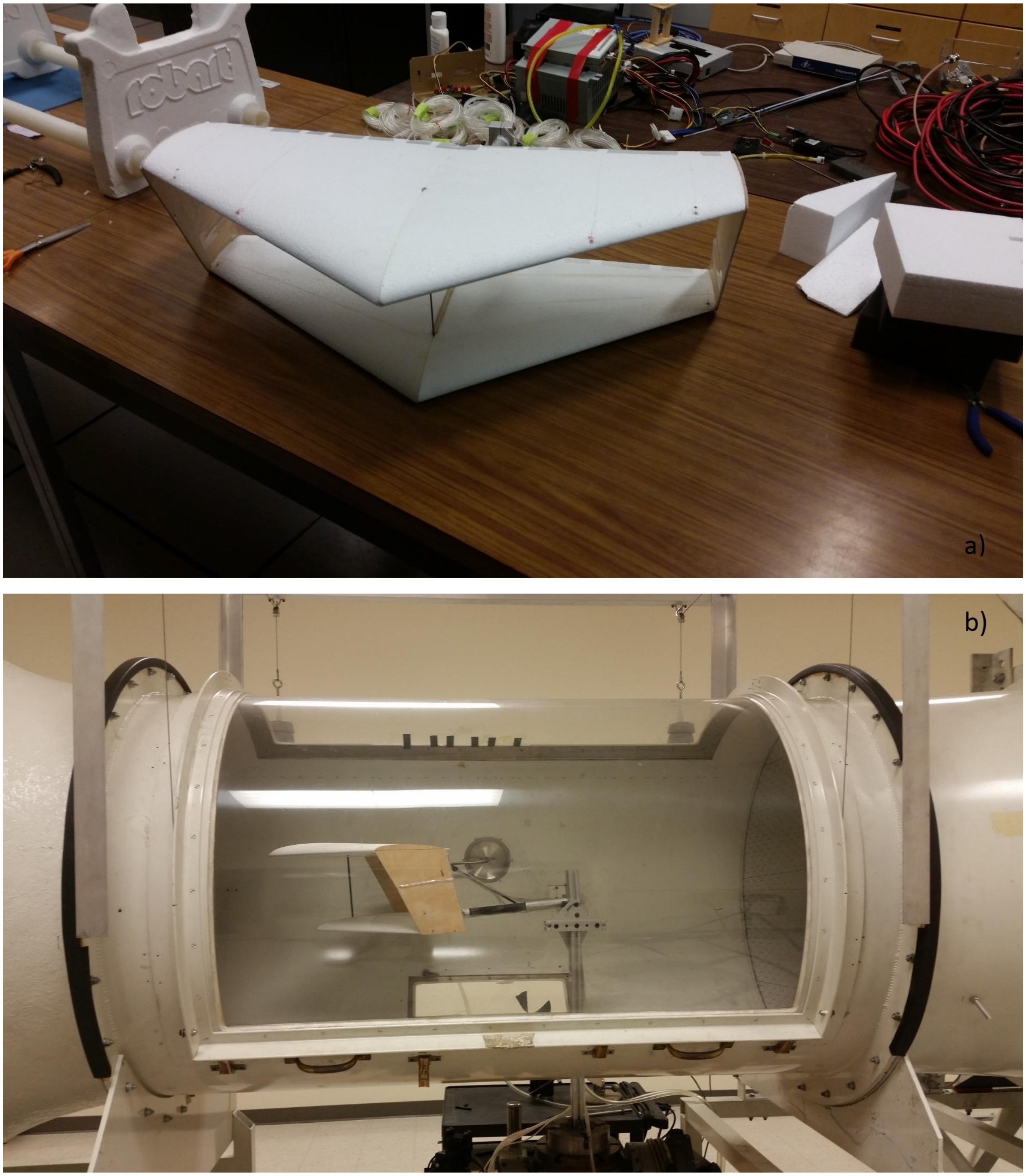


Figure 2.3: Completed wind tunnel model. Note the tubing installation in (a), while (b) shows the finished version in the wind tunnel.

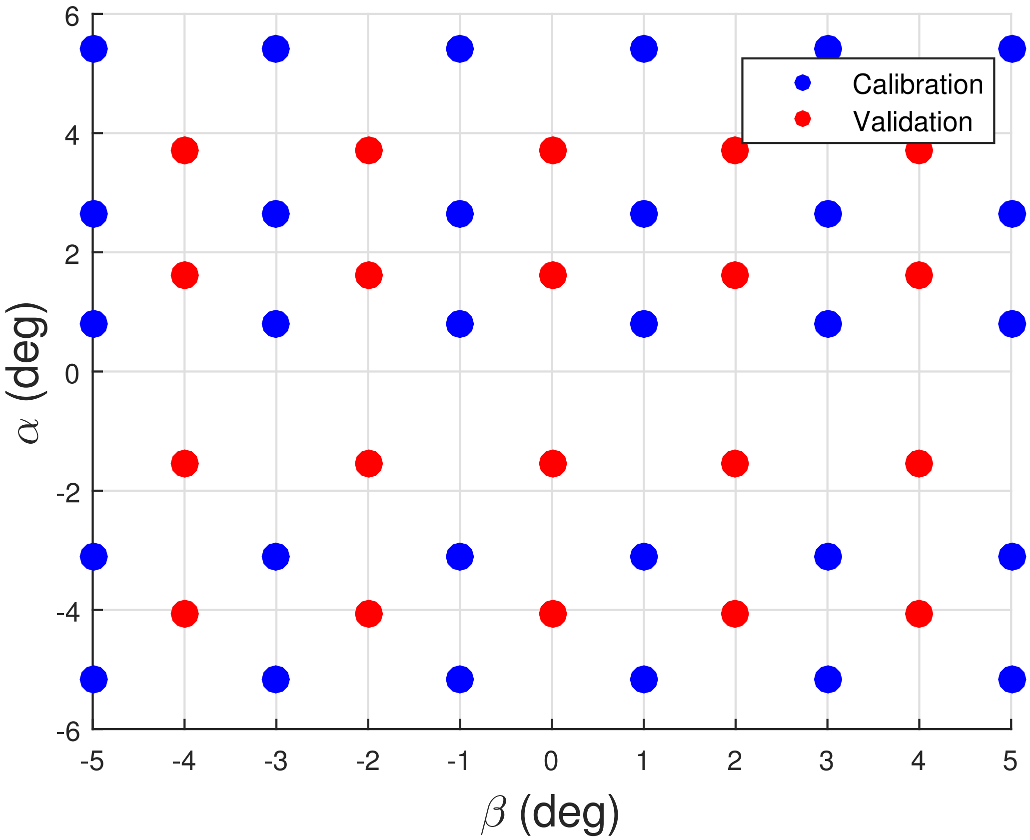


Figure 2.4: Aircraft (α, β) orientations used for wind tunnel calibration and validation tests.

used to measure the angle between the model and tunnel floor. It features an accuracy of 0.2° and resolution of 0.05°. The definition of β is the angle between the centerline of the model and the airflow. A positive sideslip is a rotation to the right from the incoming freestream wind vector. Rotating the vertical support changes β, and was measured on an analog vernier scale to about 0.1° accuracy.

Before and after each fixed α, the room pressure was measured. The average of the before and after measurements was calculated and represents the freestream pressure. This average is then subtracted from each individual sensor, yielding ΔP. All the results presented will be in terms of ΔP, and the relation between α, β, and ΔP can be seen in Fig. 2.5 for several locations. For example, at location 3, increasing α reduces the pressure compared to the freestream pressure. As will be explained in the next section, the surface fits shown in Fig. 2.5 will be used for the nonlinear

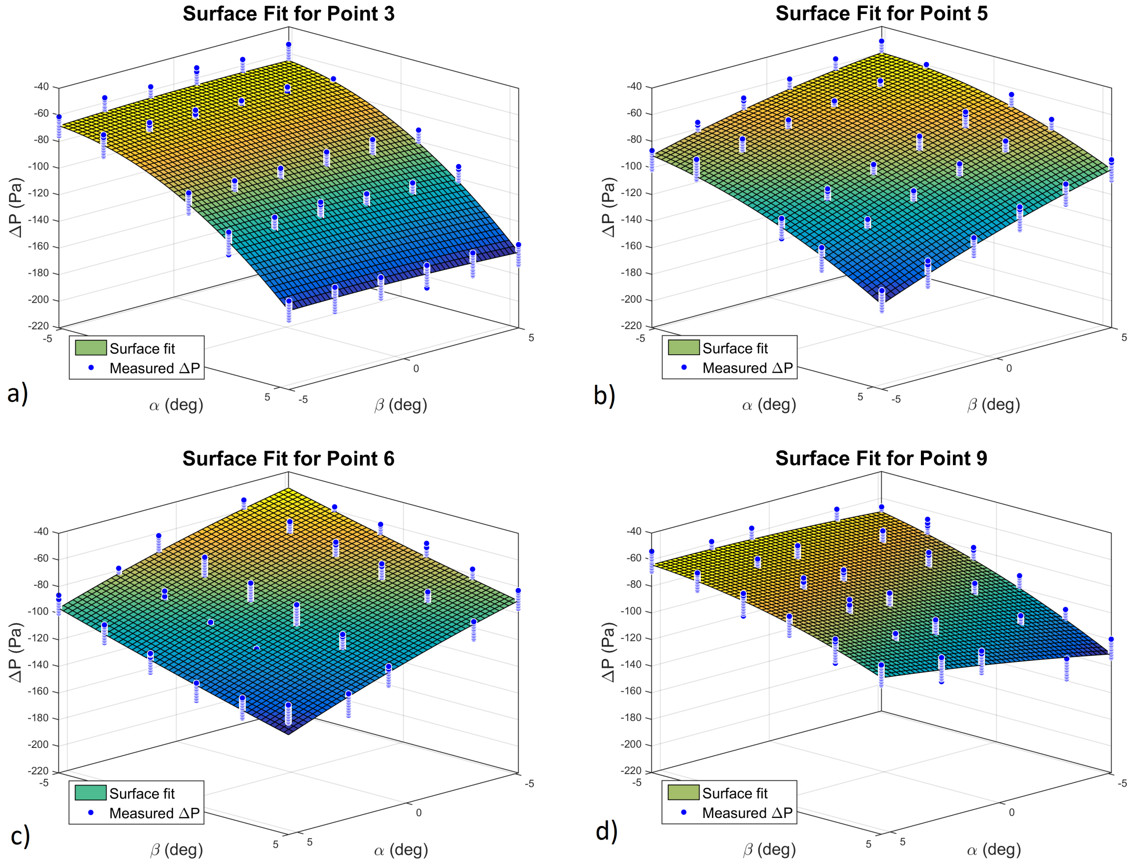


Figure 2.5: Measurement models used with nonlinear least squares. The surface fits are second-order polynomials in both directions.

least squares method, but are not needed for the neural networks.

2.4 Results

2.4.1 Nonlinear Least Squares

The first method investigated for estimating angle of attack α and sideslip β from pressure measurements was nonlinear least squares (NLS). This approach, as outlined in [8], attempts to minimize the residuals, which are the difference between the true pressure measurements and the predicted pressure measurements, by minimizing the cost function:

$$\mathrm{J} = \frac{1}{2}\mathbf{e}^{\mathrm{T}}\mathbf{W}\mathbf{e} = \frac{1}{2}[\mathbf{y} - \mathbf{f}(\hat{\mathbf{x}})]^{\mathrm{T}}\mathbf{W}[\mathbf{y} - \mathbf{f}(\hat{\mathbf{x}})] \tag{2.3}$$

with the residual vector represented by $\mathbf{e}$ and the weighting matrix by $\mathbf{W}$. This weighting matrix assigns a weight to each sensor based on the noise in the measurements; i.e., a sensor with less noise will be trusted more than a sensor with greater noise, and therefore contribute more to the final solution. The ΔP measurements from all 10 sensors are represented by $\mathbf{y}$. The estimate of α and β is the vector $\hat{\mathbf{x}}$. The nonlinear relationship between α, β and ΔP for sensor i is $\Delta \mathrm{P} = f_i(\alpha, \beta)$, where $f_i(\alpha, \beta)$ can be written as a second order polynomial:

$$\Delta \mathrm{P} = f_i(\alpha, \beta) = a_i + b_i\alpha + c_i\beta + d_i\alpha^2 + e_i\alpha\beta + g_i\beta^2 \tag{2.4}$$

with the individual coefficients determined with the MATLAB *fit* function. All 10 f_i relationships are stored in the $\mathbf{f}$ term, with f_3, f_5, f_6 and f_9 visually represented in Fig. 2.5. Additionally, the Gauss-Newton method was used for finding the minimum of J, and all estimates are produced with an initial guess of $(\alpha = 0°, \beta = 0°)$.

2.4.2 Neural Networks

The independent variables are α and β, while pressure (or more specifically, ΔP) is the dependent variable. The first step in the least squares method is to determine how the dependent variable is related to the independent variables for each sensor $(\alpha, \beta \rightarrow \Delta \mathrm{P})$, then it is possible to work in reverse so that by measuring only the dependent variable, the independent variables can be estimated $(\Delta \mathrm{P} \rightarrow \alpha, \beta)$. Neural networks, on the other hand, can directly relate the measured dependent variable to the independent variables to be estimated in one step. The networks are trained with the data from the calibration phase, resulting in a simple function that accepts ΔP and calculates α and β.

The layout of a network used for this project is shown in Fig. 2.6. The output of layer i is given by:

$$\mathbf{a}_i = f_i\left(\mathbf{W}_i\mathbf{p}_i + \mathbf{b}_i\right) \tag{2.5}$$

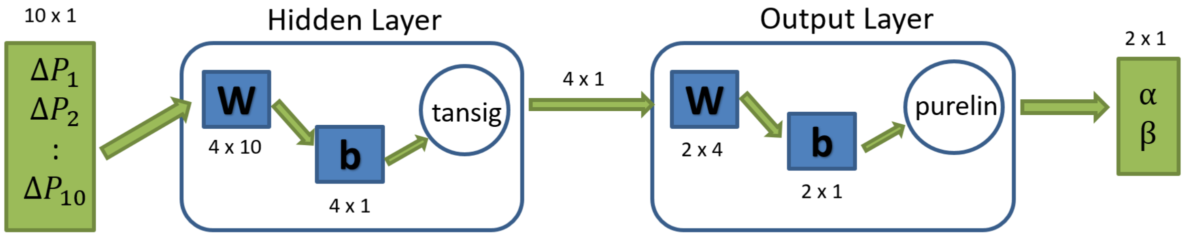


Figure 2.6: Schematic showing the layout of a single neural network.

where for the i-th layer, $\mathbf{W}_i$ is the weight matrix, $\mathbf{b}_i$ the bias vector, $\mathbf{p}_i$ is the input, and f_i is the transfer function. The ΔP measurements are the inputs for the hidden layer, which uses the MATLAB *tansig* transfer function with Eq. 2.5. These values are then the inputs into the output layer, which uses the MATLAB *purelin* transfer function to output the values of α and β.

The neural networks were trained using the MATLAB Neural Network Toolbox (the foundation of which is provided by [20]). Two ways the user can affect the accuracy of the results are by choosing the training method and the number of hidden layer neurons. Three of the training methods used for regression in the toolbox are Levenberg-Marquardt ([21, 42]), Bayesian Regularization ([16, 41]) and Scaled Conjugate Gradient ([46, 6]). Among these methods, Bayesian Regularization is the only one specifically designed to generalize well. Since the orientations used to collect the training data were different than the orientations used for validation, being able to generalize outside of the training points was of high priority, hence the reason why Bayesian Regularization is the chosen training method.

In addition to choosing the training method, the number of neurons in the hidden layer must also be specified. Too few neurons and the network cannot handle the complexity of the relationship between ΔP, α and β (underfitting the data); too many neurons and it will not generalize well outside of the training points (overfitting the data). The optimal number of hidden layer neurons to use can be seen in Fig. 2.7. Four neurons leads to the smallest mean-square error for α and β simultaneously. The number of neurons in the output layer are determined by the number of outputs, and as such, are not user changeable.

Neural networks are initialized with random weights and biases. Even if the network is

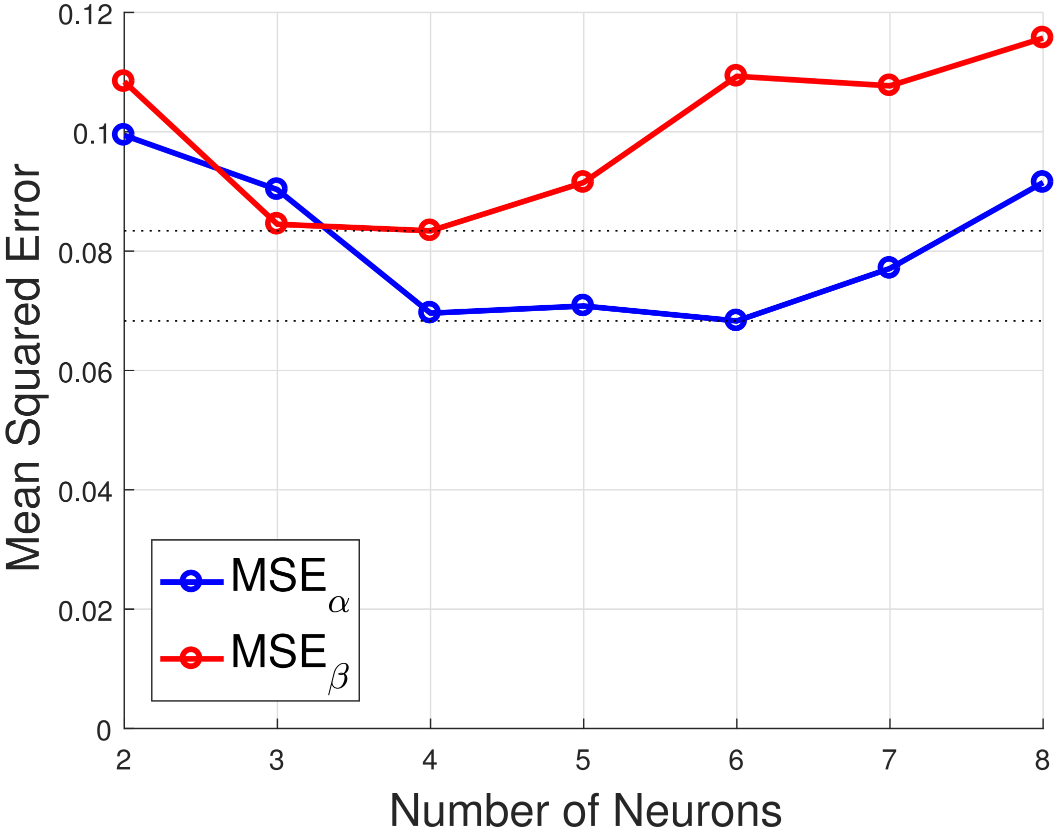


Figure 2.7: Mean squared errors (MSE) using different numbers of neurons in the hidden layer.

retrained using the same data, it is possible to produce different results. Therefore, it is possible to use an ensemble of networks and get a range of outputs. The number of networks to use in the ensemble is shown in Fig. 2.8. Convergence starts around 30-40 networks. The only penalty incurred for using more networks is a larger amount of time training them. For all the results presented in this paper, an ensemble size of 30 networks was used.

2.4.3 Results

While 300 samples were taken at each validation orientation, only a single sample was used at a time to produce an estimate. This led to 300 estimates for each orientation. The case where the true orientation was $(-4.05°, -4.0°)$ is shown in Fig. 2.9 (a) and (b) (for NLS and neural networks, respectively). The black diamond shows the true orientation, and each blue dot is an estimate

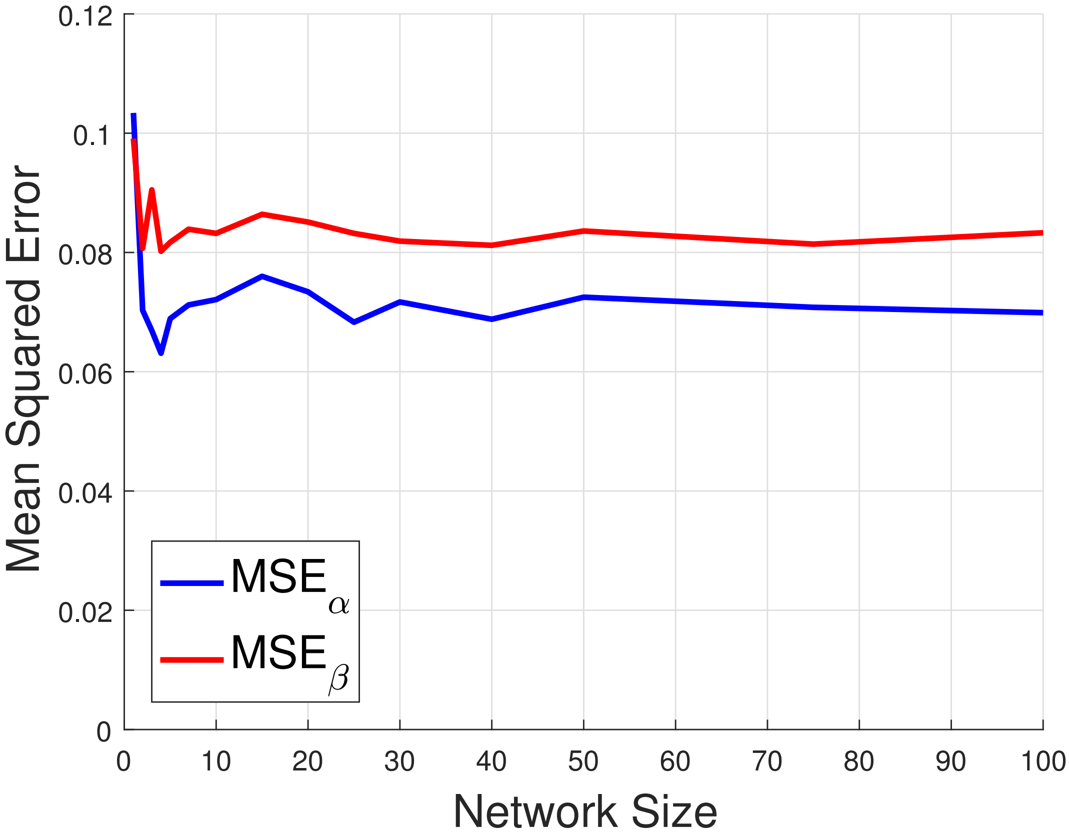


Figure 2.8: Convergence of the mean squared error as a function of the ensemble size.

produced by using one of the pressure samples. It is immediately clear that for this particular orientation, NLS has a significantly stronger bias in the estimates than the neural networks. In addition, the neural network estimates are grouped tighter than the NLS estimates. The estimates produced for all 20 test orientations are shown in Fig. 2.9 (c) and (d). For the lower α values, there is a clear bias in NLS, although as α increases, the bias decreases.

With the true orientation expressed as $\mathbf{x}_{\text{true}}$, the error for an estimate is calculated with:

$$\varepsilon = \hat{\mathbf{x}} - \mathbf{x}_{\text{true}} \tag{2.6}$$

Each test orientation therefore has 300 errors associated with the estimates. The statistics of those errors are shown in Fig. 2.10. The mean error and two standard deviation (σ) bounds for α are shown in Fig. 2.10 (a) and (b) for NLS and neural networks, respectively. For example, when the

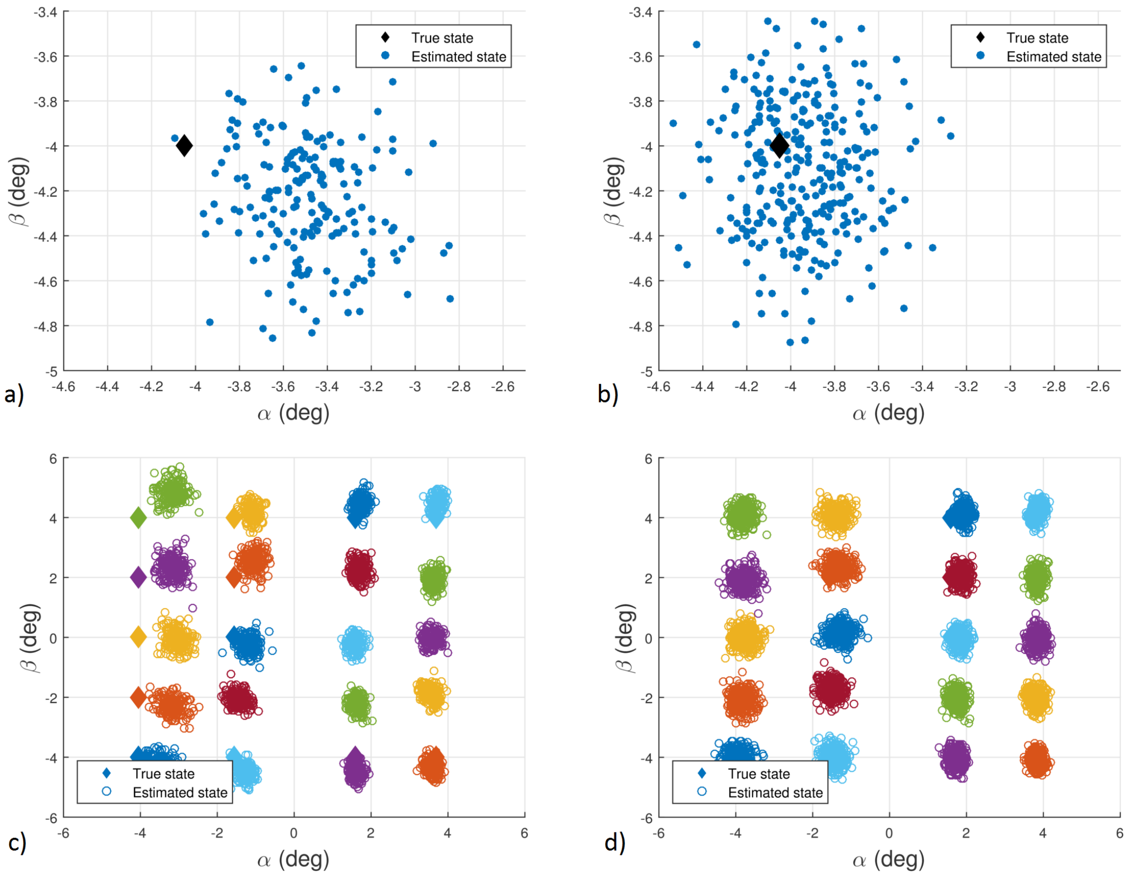


Figure 2.9: Scatter plots showing the estimates produced by nonlinear least squares and the ensemble of neural networks. Estimates from NLS (a) and the neural networks (b) are shown for the case where the true orientation is ($\alpha = -4.05°$, $\beta = -4.00°$). The estimates for all the validation orientations can be seen in (c) and (d) for NLS and neural networks, respectively.

true orientation is $(-4.05°, -4.00°)$, NLS has a mean error of almost 0.6°, while the neural networks have a mean error of only about 0.1°. It is also easier to see the trend mentioned earlier for NLS where it is less biased for higher angles of attack. It should also be noted that both methods almost always overestimate the angle of attack.

The mean errors and 2σ bounds for β are shown in Fig. 2.10 (c) and (d). Again, it is immediately apparent that the neural networks produce significantly smaller mean errors than NLS. For NLS, the mean errors tend to be biased in the same direction as the true β, and scale with the true β as well. The most positive sideslip tested generally has the most positive bias to the estimates, while the most negative sideslip tested almost always has the most negative bias.

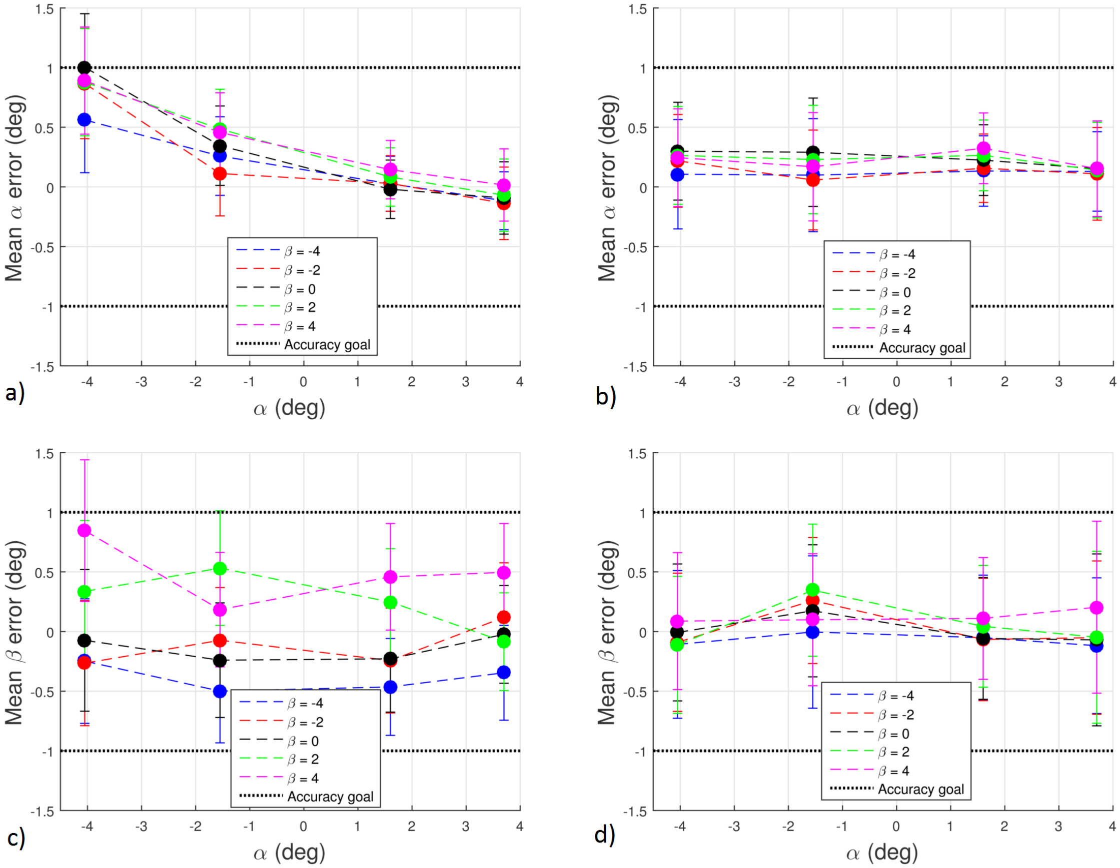


Figure 2.10: Comparison of mean errors and 2σ bounds. Errors in α for NLS (a) and neural networks (b), and errors in β for NLS (c) and neural networks (d).

This trend is significantly weaker when using neural networks, but still appears for some of the validation orientations.

While all the sensors worked the entire time, it is possible that a sensor might fail at some point. To investigate the effects of a failed sensor, estimates were produced ignoring the measurements from a single sensor at a time. The resulting mean squared errors (MSE) are shown in Fig. 2.11 for neural networks. The case of NLS with failed sensors was not considered. Note that the dashed lines represent the baseline MSE with all ten sensors being used for estimates of the orientation. It will also be useful to refer back to Fig. 2.5 to see why some sensors have a larger effect on α or β.

When sensors 2, 3 and 4 are ignored, there is relatively little effect on the β errors. This

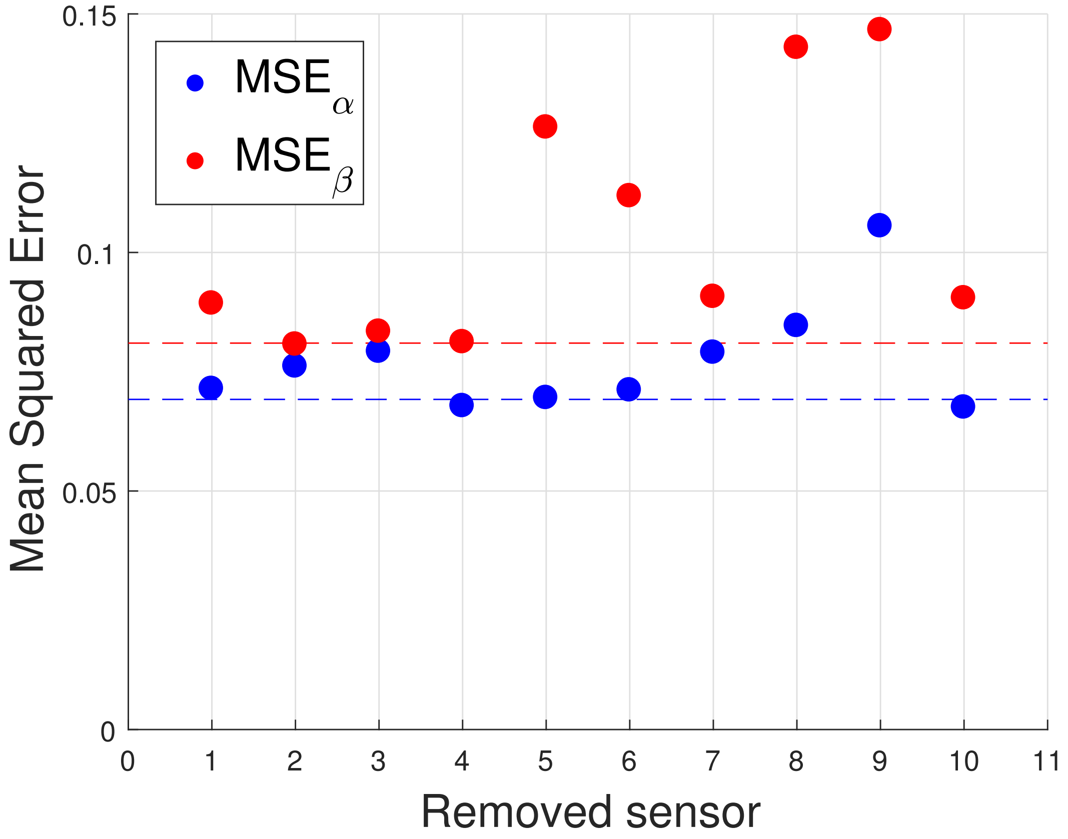


Figure 2.11: Mean squared error for the removal of an individual sensor.

makes sense as the pressure at these locations is nearly independent of β (sensors 2 and 4 have nearly identical surfaces to sensor 3). However, the pressure at location 1 is independent of β at low α, but as α is increased, the pressure does become dependent on β, hence why MSE_β jumps when sensor 1 is removed. It is unsurprising that MSE_β increases after removing sensors 5-10 as these locations were chosen specifically because they are more sensitive to changes in β. Locations 1-4 were chosen for their sensitivity to α; it is therefore surprising that removing sensors 1 or 4 has minimal impact on MSE_α. Perhaps the most surprising result is when sensor 9 is removed. This location was chosen for its sensitivity to β, yet this sensor has the largest effect on MSE_α.

Finally, it is also interesting to look at how the errors for both methods are distributed. Histograms of the errors for all the validation orientations are shown in Fig. 2.12. The errors from neural networks are well represented by a Gaussian distribution, although only the β errors have

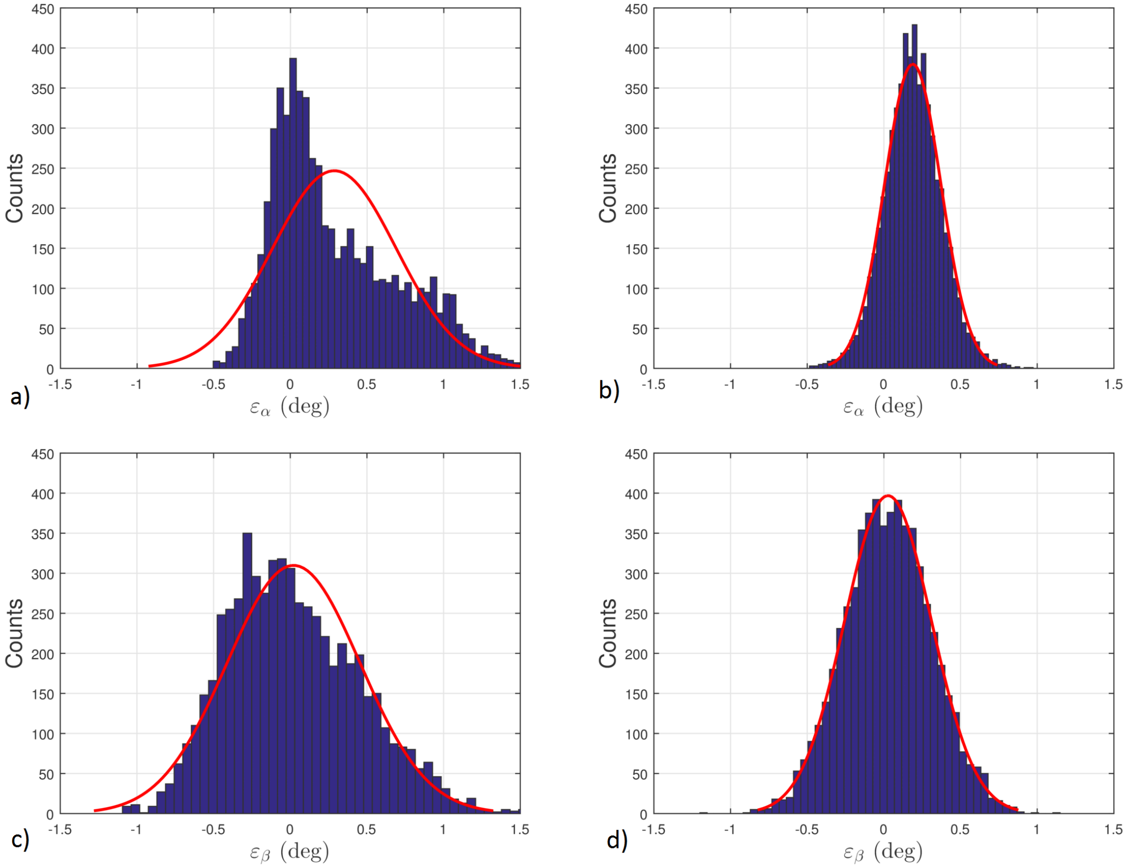


Figure 2.12: Histogram of all the errors from nonlinear least squares, (a) and (c), and neural networks, (b) and (d).

zero mean. All of the estimates of α were within 1° of the truth, while only two out of the 6000 total estimates for β were greater than 1° from the truth. For NLS, the β errors are close to a zero-mean Gaussian, but the α errors are clearly not representative of a Gaussian distribution. Additionally, 93.1% of the α errors and 98.0% of the β errors are within 1° of the truth.

2.5 Conclusion

The wind tunnel results presented show that a distributed flush airdata system is indeed a viable method of wind sensing from small UAS. Different locations contribute to the final estimate in varying amounts, so it is important to choose suitable locations. It was also quite clear that using an ensemble of neural networks leads to better results than the nonlinear least squares approach.

While neural networks performed better, both methods achieved mean errors that were less than 1°, which is the maximum error specified by commercial multi-hole probe systems (specifically, the Aeroprobe 5-hole probe). Lastly, for the neural networks, 100% of the errors in the angle of attack α and 99.97% of the errors in sideslip β were less than 1°.

The wind tunnel tests served to verify the presented approach to implementing a distributed flush airdata system. To provide the full relative wind vector, the FADS must also measure airspeed. Chapter 3 outlines an updated approach for choosing locations on a new airframe, including locations for the measurement of airspeed. Due to size constraints, it is not always possible to calibrate the FADS in the wind tunnel. Chapter 4 presents an alternative calibration approach, performed in-flight, along with results from flight testing of the updated flush airdata system.

Chapter 3

Distributed Flush Airdata System for the Skywalker UAS

3.1 Introduction

Chapter 2 summarized the early developments of the distributed flush airdata system. There were numerous simplifications applied, such as only focusing on the determination of the angle of attack (α) and the angle of sideslip (β), running simulations at a fixed airspeed and density, and only testing inside a wind tunnel. This descoping allowed for faster progress for the proof of concept phase. However, the prior work must be expanded upon before moving on to flight tests.

Section 3.2 will discuss the new method of determining port locations. This new method differs from the old one in that it deals directly with the pressure coefficient gradients, as opposed to the heuristic approach outlined in Chapter 2. Section 3.3 will outline the process of assembling and sealing the circuit boards for measuring the pressure at the ports. Section 3.4 will detail the hardware integration with the Skywalker UAS. Finally, Section 3.5 will discuss the conclusions and lessons learned with particular approach to determining the port locations and assembly of the hardware.

3.2 Selecting Pressure Port Locations

Certain locations on the airframe are more sensitive to changes in the flow conditions than other locations. Therefore, it is important to determine the locations that are better suited for a flush airdata system before installation can even begin. Potential flow has been used previously on a simple shape to determine the most sensitive locations [80, 70, 54], but is significantly more difficult to apply to a full airframe.

Similar to the method outlined in Chapter 2 and [38], computational fluid dynamics (CFD) will be used to identify the most sensitive locations. However, the approach outlined in the current chapter has several notable differences compared to the previous approach: dealing directly with pressure coefficients instead of cost functions, simulating multiple airspeeds and altitudes (synonymous with density), and determining locations for measuring static and stagnation pressures. While CFD was also used to determine port locations in [65], the method outlined in this chapter differs in that 3D simulations were performed in order to account for pressure changes due to varying sideslip angles. The basic premise of looking for locations with steep gradients is the same between the outline in this chapter and [65]. Certain locations on the Skywalker are also excluded for reasons that will be explained later.

3.2.1 Skywalker UAS

The X-8 Skywalker UAS is one of the primary instrument testing platforms for the Research and Engineering Center for Unmanned Vehicles. It is readily available, offers a large amount of payload space, and as of November 2016, the airframe can be purchased for \$200-\$250. The foam construction allows for easy hardware integration and quick repairs in the event of hard landings. The Skywalker has a cruise speed around 17 m/s, and an endurance of approximately 45 minutes when using a 9,000 mAh battery.

Two other features of the Skywalker that are ideal for a distributed FADS are the rear propeller and the winglets. As was shown in Chapter 2, locations closer to the leading edges are more sensitive to flow angle changes. Having a propeller in the rear allows the entire front of the airframe to potentially be used for port locations. Additionally, the winglets add a vertical surface, thus increasing the sensitivity to changes in β. A Skywalker making a low pass during test flights is shown in Fig. 3.1.

Figure 3.1: Skywalker UAS performing a low pass.

3.2.2 CFD Simulations

STAR-CCM+ [66] was the CFD software package used for the simulations. For the simulations, a polyhedral mesh with cell sizes ranging from about 1.5 mm near the leading edges to nearly 6.5 mm on the flatter parts of the airframe was generated. The cells on the surface of the Skywalker are seen in Fig. 3.2, focusing on the fuselage and portions of the wings (a) and the winglets (b). Five prism cell layers are generated above the surface to help resolve the boundary layer. A total of approximately 110,000 cells are on the surface of the simulation model.

As discussed previously, the pressure on the surface of the aircraft can be defined as a function of α, β, freestream airspeed (V_∞), freestream static pressure (P_∞), and freestream density (ρ_∞) (where the freestream temperature T_∞ is coupled with P_∞ and ρ_∞ through the ideal gas law). Under US Standard Atmosphere [47] conditions, P_∞ and ρ_∞ decrease with altitude at a prescribed rate. To reduce the dimensionality of the simulations, P_∞ and ρ_∞ are combined together into the altitude dimension, thus reducing the simulations down to four dimensions: α, β, V_∞, and the

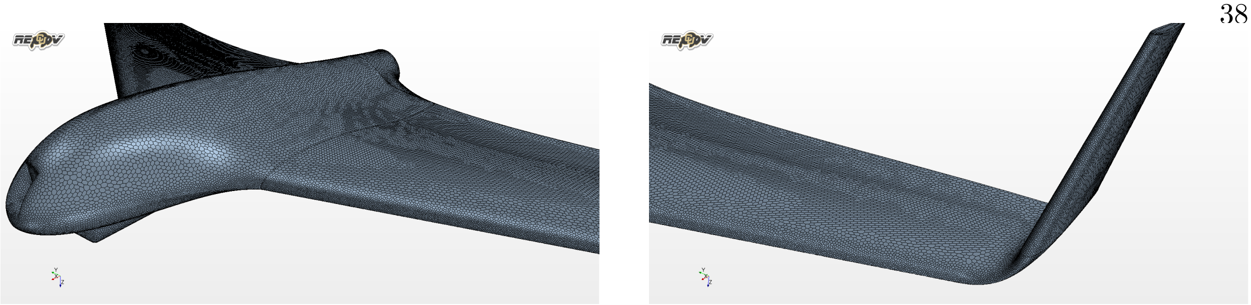

(a) (b)

Figure 3.2: Close-up views of the polyhedral volume mesh generated. The cell size decreases at locations with a higher radius of curvature, such as the leading edges.

Table 3.1: Thermodynamic properties for simulations

h	P_∞ (Pa)	ρ_∞ (kg/m^3)	T_∞ (°C)
1829 m (6000 ft)	81200	1.0239	3.15
2438 m (8000 ft)	75250	0.9629	-0.85

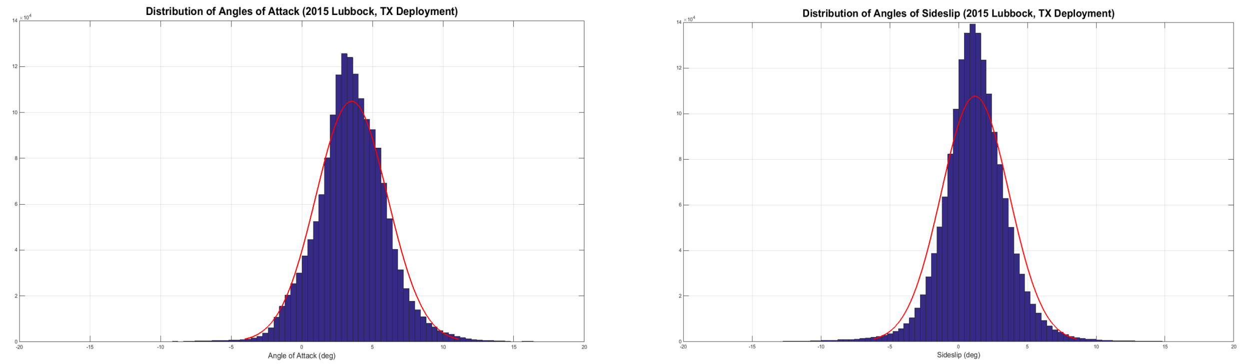
altitude (h). The altitudes used and corresponding relevant thermodynamic properties are shown in Table 3.1. For V_∞, simulations were performed at 14, 17, and 20 m/s.

The data obtained during the 2015 Lubbock, Texas, field campaign proved useful for determining the range of α and β values to use. For two weeks during the month of June, the Tempest UAS was flown with a five-hole multi-hole probe (MHP) from the Aeroprobe Corporation. The distribution of α and β measured by the MHP during the campaign is seen in Fig. 3.3 and represent approximately 1.5 $\times 10^6$ data points. A Gaussian fit in red is shown for each distribution. For β, three standard deviations covers the range from $-6°$ to $8°$, though it is clear the distribution is not zero mean. The mean β is expected to be zero, though this offset could be due to MHP misalignment. Aircraft typically have some small positive α during steady level flight, so a nonzero mean for α makes sense. The 3σ range for α is from $-4°$ to $12°$. Based on the Lubbock 2015 measurements, α is varied from $-4°$ to $12°$ in $2°$ increments in the simulations, while β ranges $-6°$ to $6°$ in $1.5°$ increments. The variations in α, β, V_∞ and h led to 486 simulations being performed.

Pressure distributions across the aircraft at trim condition for various altitudes and airspeeds are shown in Fig. 3.4. The pressure for simulation results is defined as the difference between the

Table 3.2: Simulation test matrix

α	$[-4°, -2°, 0°, 2°, 4°, 6°, 8°, 10°, 12°]$
β	$[-6°, -4.5°, -3°, -1.5°, 0°, 1.5°, 3°, 4.5°, 6°]$
V_∞	$[14, 17, 20]$ m/s
h	$[1829, 2438]$ m



(a) Distribution of α

(b) Distribution of β

Figure 3.3: Distribution of the recorded angles of attack and sideslip during the 2015 Lubbock, Texas, deployment.

absolute and freestream pressures. Additionally, for the simulations, trim condition is taken to be $\alpha = 2°$ and $\beta = 0°$. The pressure behaves as expected on the wings: an increase in airspeed leads to a decrease in pressure on the top surface of the wings, with the decrease being greatest closer to the leading edge. These pressure distributions are more useful to see how the pressure behaves on the fuselage and near the airframe irregularities (such as the carved out portion in the nose).

Oftentimes, the pressure coefficient ($\mathrm{C_p}$) is utilized to incorporate the pressure changes due to varying ρ_∞ and V_∞. The definition of $\mathrm{C_p}$ is shown in Eq. 3.1:

$$\mathrm{C_p} = \frac{\mathrm{p} - \mathrm{p}_\infty}{\mathrm{q}_\infty} \tag{3.1}$$

with q_∞ representing the dynamic pressure, which is defined as:

$$\mathrm{q}_\infty = \frac{1}{2}\rho_\infty \mathrm{V}_\infty^2 \tag{3.2}$$

Even at a fixed location on the same airfoil with a constant α and β, the value of $\mathrm{C_p}$ will change as the freestream Mach number (M_∞) changes. Different compressibility corrections can be applied, one of which is the Prandtl-Glauert rule [2]. The C_p defined in Eq. 3.1 is the low-speed, or incompressible, definition, and can also be represented by $C_{p,0}$. The Prandtl-Glauert rule adds a correction term in the denominator, as shown in Eq. 3.3:

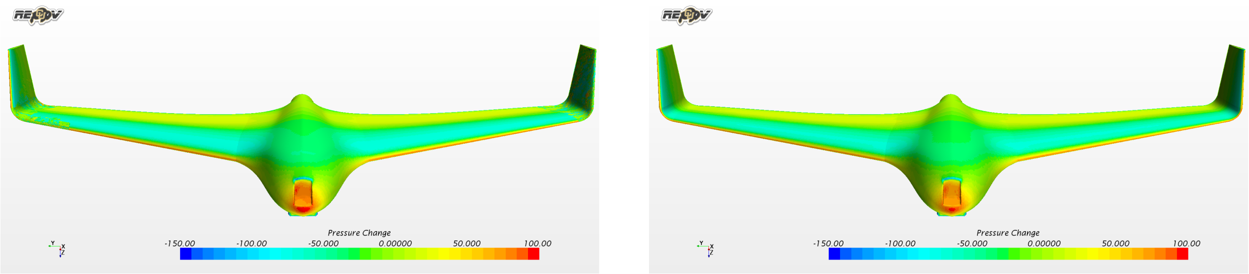
$$C_p = \frac{C_{p,0}}{\sqrt{1 - M_\infty^2}} \tag{3.3}$$

This rule is reasonably accurate for M_∞ values up to 0.7. At $M_\infty = 0.3$ (generally accepted as the upper limit at which air is considered incompressible, [2]), $C_{p,0}$ is within 5% of the true C_p. For the CFD simulations, M_∞ ranged from a minimum of 0.04 to a maximum of 0.06, which leads to a maximum error in C_p of 0.2%. As the simulations are clearly in the incompressible range, it is assumed that C_p is unaffected by changes in speed or density. Therefore, all subsequent results will be presented in the form of C_p to reduce the dimensionality of the problem.

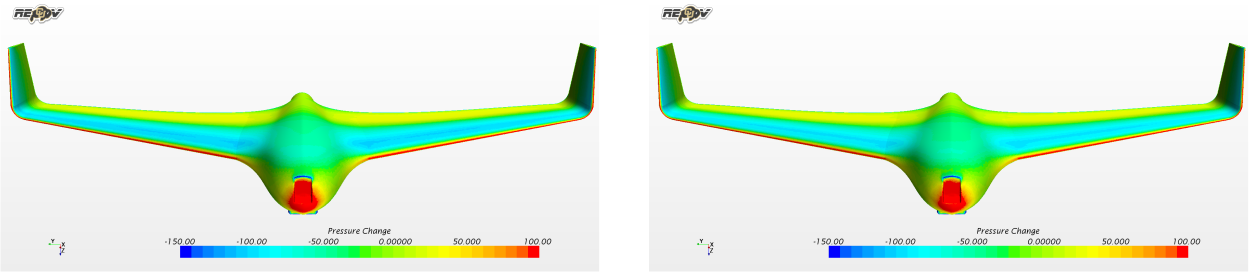
For a single α and β orientation, six simulations were performed: at altitudes of 1829 and 2438 m, and at each altitude, airspeeds of 14, 17 and 20 m/s. At each of these six simulations, the pressure coefficient is calculated across the aircraft (one of the outputs of the simulations is pressure on the surface, so it is a trivial task to convert pressure to pressure coefficient). The pressure coefficients from the different altitudes and airspeeds can then be averaged into a mean pressure coefficient.[1] This C_p is shown for several different α and β orientations in Fig. 3.5. As α increases, C_p decreases in the vicinity of the leading edges. It is also easy to see that locations closer to the leading edges are more sensitive to changes in α than the rest of the aircraft. The same can be said about the leading edges of the winglets for changes in β. Additionally, the area of decreased C_p changes sides on the winglets based on the whether β is positive or negative, and is most visible in Fig. 3.5 (e) and (f). Finally, note that changing α does have an effect on C_p across the winglets, but changing β has virtually no effect on C_p across the wings.

---

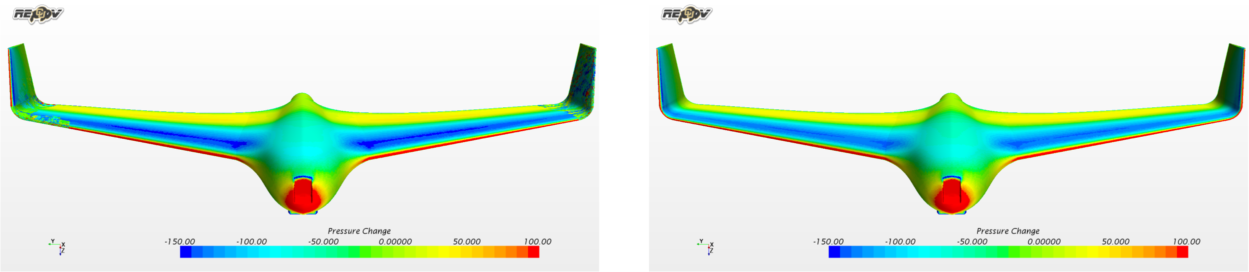
[1] For simplicity, the mean pressure coefficient will be represented by C_p throughout the rest of the chapter. When the regular pressure coefficient is used, the altitude and airspeed will also be specified.



(a) h = 1829 m, V_∞ = 14 m/s (b) h = 2438 m, V_∞ = 14 m/s



(c) h = 1829 m, V_∞ = 17 m/s (d) h = 2438 m, V_∞ = 17 m/s



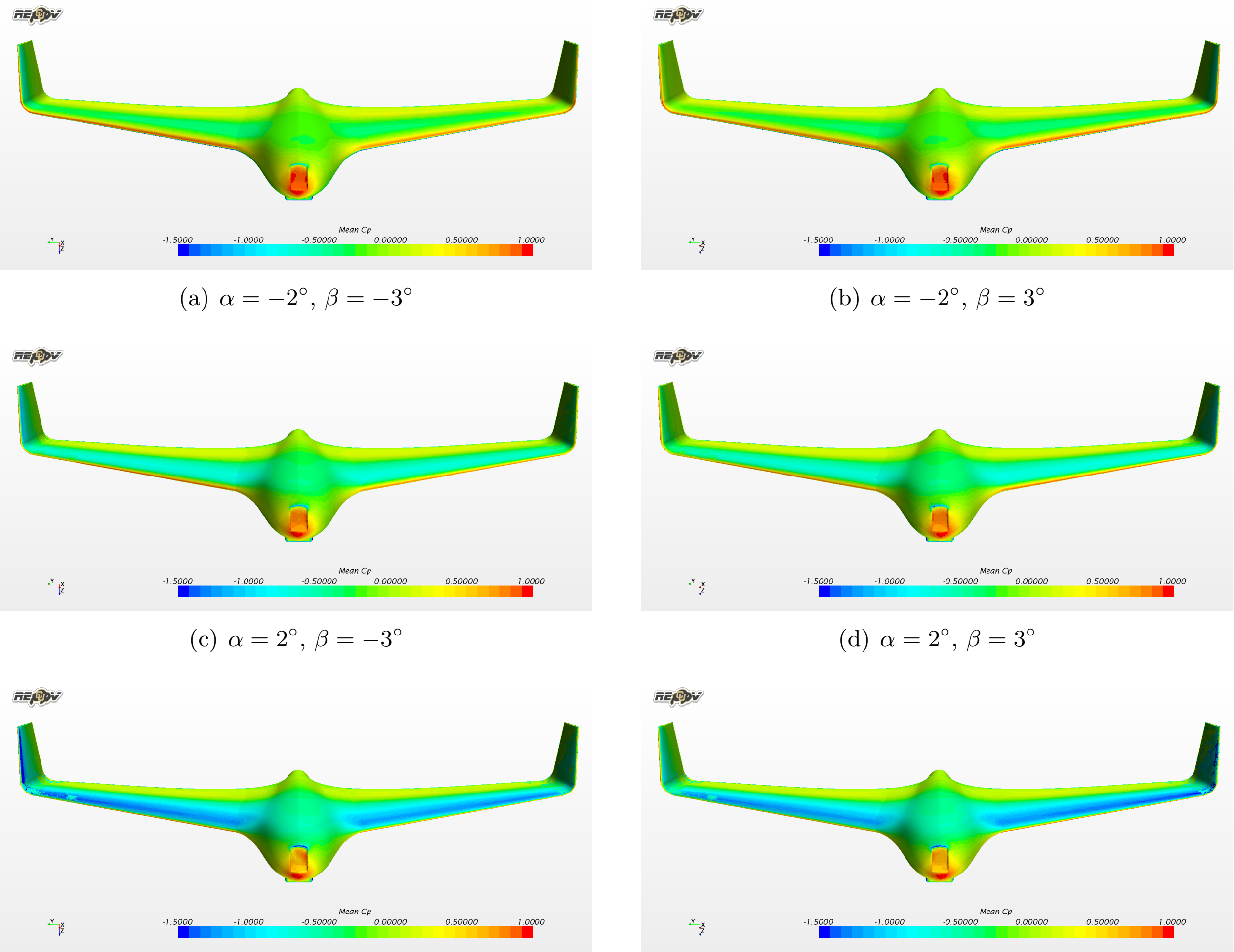
(e) h = 1829 m, V_∞ = 20 m/s (f) h = 2438 m, V_∞ = 20 m/s

Figure 3.4: Pressure changes for varying altitudes and freestream velocities. All plots are at trim condition.

3.2.3 Locations Best Suited for α and β

Previously, the port locations were chosen through the use of cost functions that prioritized locations that experienced a large and predictable range in pressure for changes in the flow angles. A simpler approach is outlined here utilizing the derivative of the mean pressure coefficient with respect to α and β.

Point 67914 will serve as the example to demonstrate the new methodology for port location selection. Point 67914 is located on the extreme forward portion of the leading edge of the right wing. The value of C_p is shown in Fig. 3.6 (a). As the mean pressure coefficient has been reduced



(a) $\alpha = -2°$, $\beta = -3°$

(b) $\alpha = -2°$, $\beta = 3°$

(c) $\alpha = 2°$, $\beta = -3°$

(d) $\alpha = 2°$, $\beta = 3°$

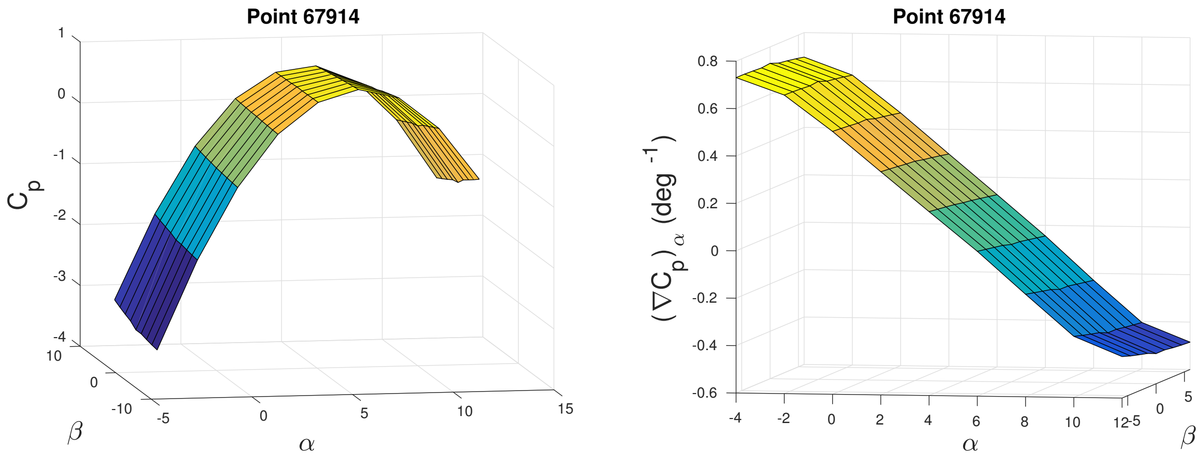
(e) $\alpha = 6°$, $\beta = -3°$

(f) $\alpha = 6°$, $\beta = 3°$

Figure 3.5: The mean pressure coefficient at several different angles of attack and sideslip.

to a function of α and β, the gradient of $\mathrm{C_p}$ will consist of the derivative in the α and β directions, represented by $(\nabla \mathrm{C_p})_\alpha$ and $(\nabla \mathrm{C_p})_\beta$, respectively. For point 67914, $(\nabla \mathrm{C_p})_\alpha$ is shown in Fig. 3.6 (b). The gradient of $\mathrm{C_p}$ shows exactly how sensitive that particular point is to changes in α and β. Integrating to find the volume under both $(\nabla \mathrm{C_p})_\alpha$ and $(\nabla \mathrm{C_p})_\beta$ leads to a scalar "total sensitivity" value for both the α and β directions. Locations with a higher sensitivity magnitude over a larger area in (α, β) space will lead to larger total sensitivity values. The individual points on the surface of the aircraft are then ranked based on their total sensitivity values.

The total α sensitivity is shown in Fig. 3.7. Unsurprisingly, the leading edge of the wings is where the total α sensitivity is greatest. While other flush airdata systems are typically located



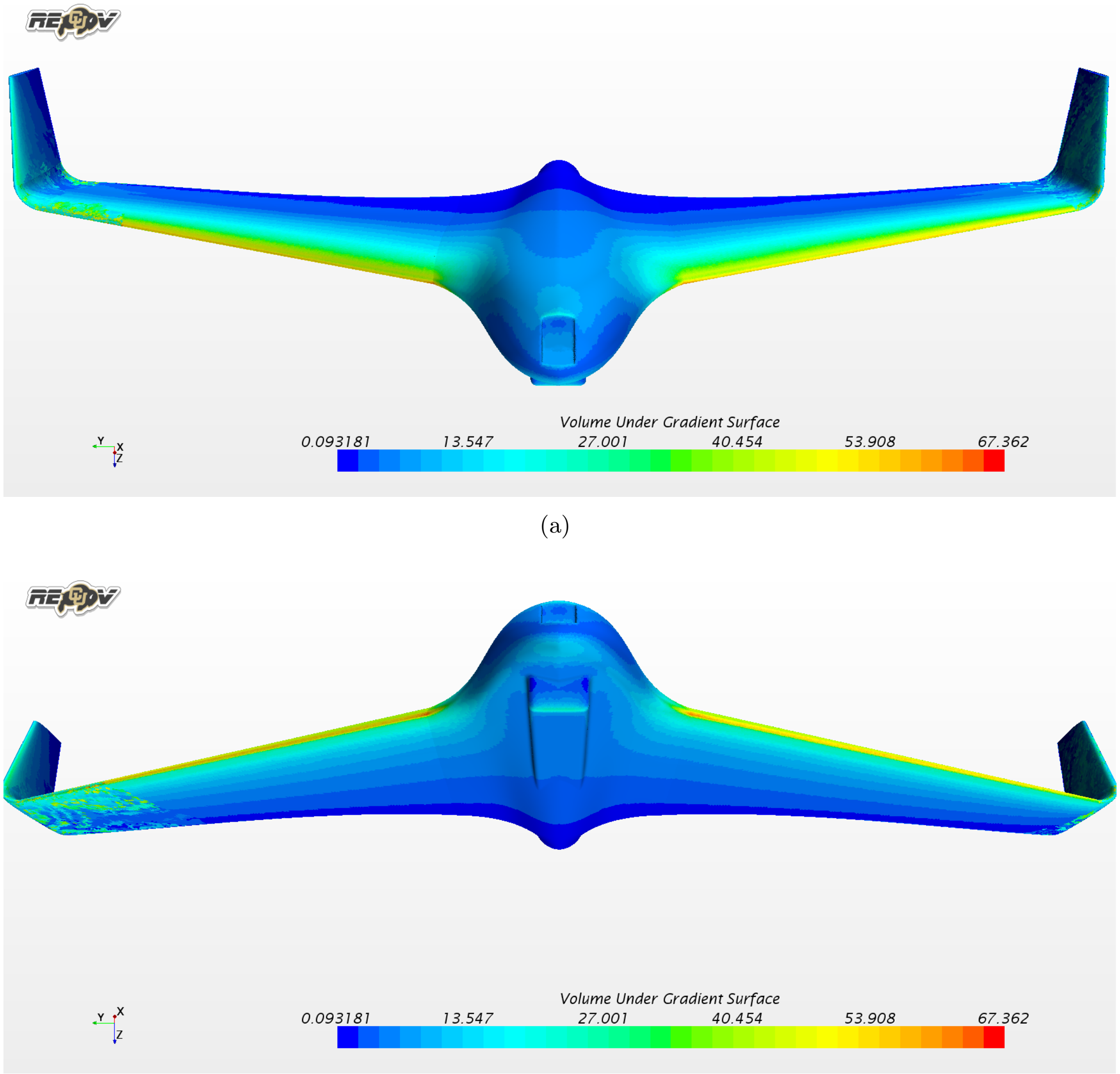
(a) (b)

Figure 3.6: Example of a point on the extreme forward portion of the leading edge.

in the nose, at least for the Skywalker, the nose is not the most sensitive location to changes in α. As such, it is possible to make the case that the leading edges are a more ideal port location than the nose is. It is also noticeable in the image that some areas near the winglet look like paint splatters and are not as smooth as the rest of the aircraft. This is a result of artifacts in some of the simulations. This is most easily shown in Fig. 3.5. Some of the simulations in this figure (e,f) show the same splotches, but the rest do not. As the total sensitivity approach incorporates all the simulations, the artifacts end up appearing in the final plot as well. In the end, this has no effect on the results. While the left winglet is affected, the right winglet is not (assuming symmetry, only one winglet is actually required to choose locations). The winglet transition area is affected on both wings, but, as discussed later, is excluded from consideration anyway. Finally, while a portion of the right wing is affected, the entirety of the left is unaffected, up to the transition point. Therefore, while some of the simulations exhibit unexpected results, it is not necessary to redo them.

The total β sensitivity is shown in Fig. 3.8. It is immediately apparent that the leading edges of the wings are not very sensitive to changes in β, but that the winglets are very sensitive. It is hard to see, but the interior face of the winglets are more sensitive than the exterior face. This is likely due to some asymmetry in the winglet shape.

When comparing the results of the α and β total sensitivities, there are a few key results to

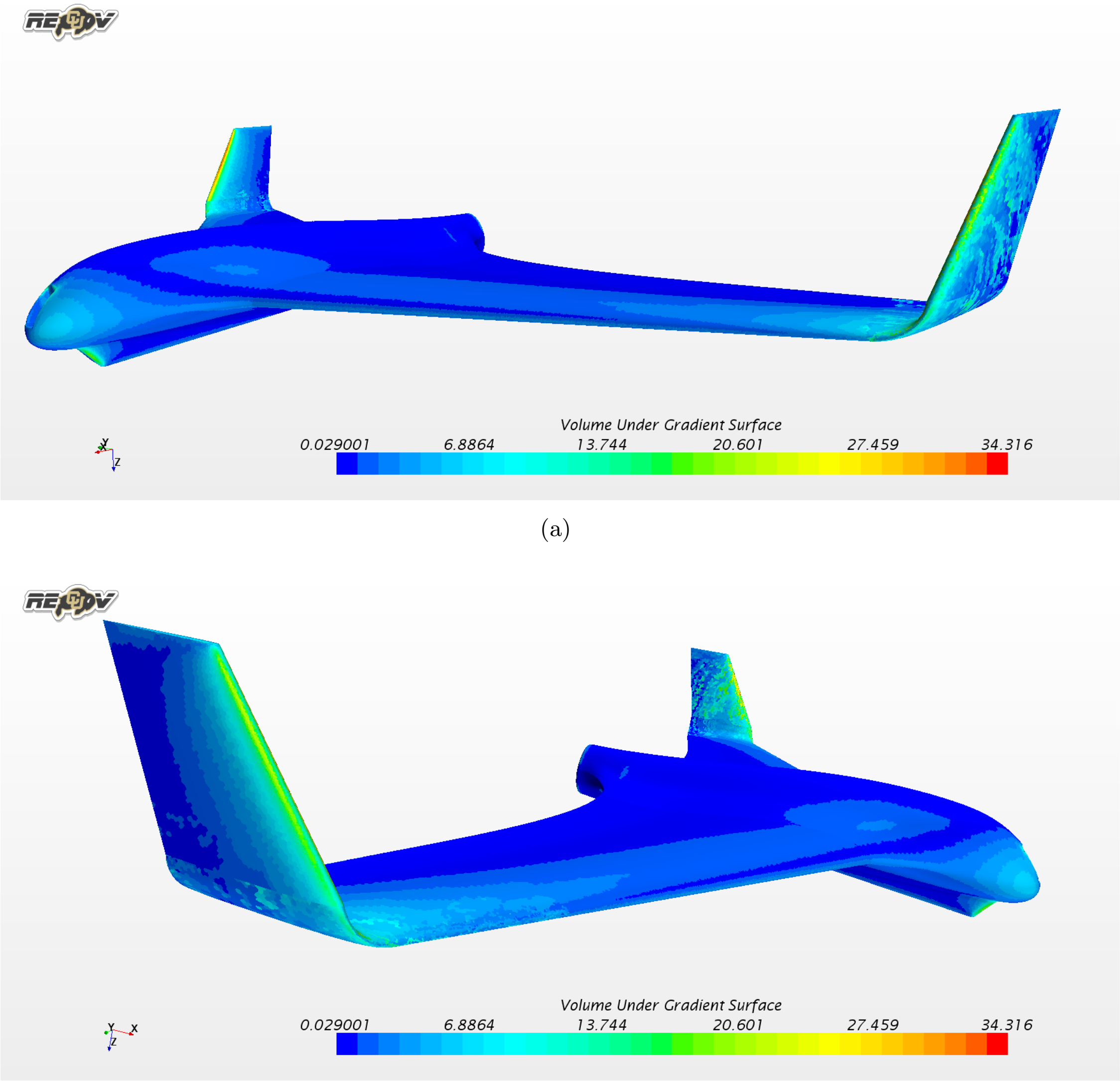


(a)

(b)

Figure 3.7: Volume under the α gradient surface for the mean pressure gradient.

note. First, the peak α total sensitivity is greater than that of the peak β total sensitivity. This can be explained by a combination of a greater range in the simulated angles of attack and that the wing exhibits a greater pressure response compared to the winglet (due to the wing being designed to provide lift), meaning greater values for $(\nabla \mathrm{C_p})_\alpha$. The second key result is for the nose of the aircraft: α sensitivity is highest on the top and bottom, while β sensitivity is highest on the sides of the nose. This is the same result as could be found using potential flow theory on a hemisphere (though in both the α and β case, the total sensitivity on the nose is noticeably less than on the



(a)

(b)

Figure 3.8: Volume under the β gradient surface for the mean pressure gradient.

wings or winglets, respectively). Finally, the winglets have elevated levels of total α sensitivity compared to the majority of the aircraft. It is hypothesized that this is due to the backward slant of the winglets. Furthermore, a subsequent investigation should be conducted into whether the winglets could serve as the only area for port locations since it exhibits good sensitivity for both α and β.

A scatter plot for both α and β total sensitivities is shown in Fig. 3.9. A better sense of the scales for both sensitivities can be gained through this plot. While the sensitivities are plotted by

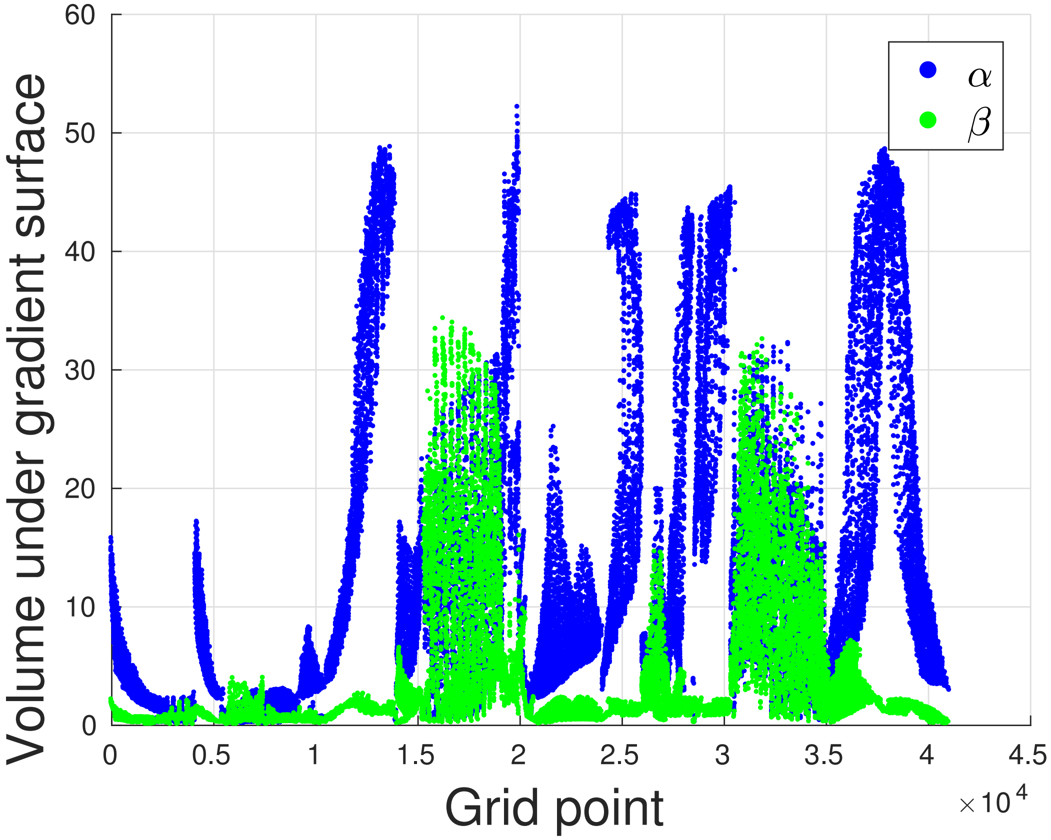


Figure 3.9: Volume under the pressure coefficient gradient surfaces.

point number, it is still possible to identify some aircraft locations. For example, the two distinct peaks for the β sensitivity are known to occur on the winglets. It is also interesting to note that at these peaks, the α sensitivity is nearly equivalent to the β sensitivity.

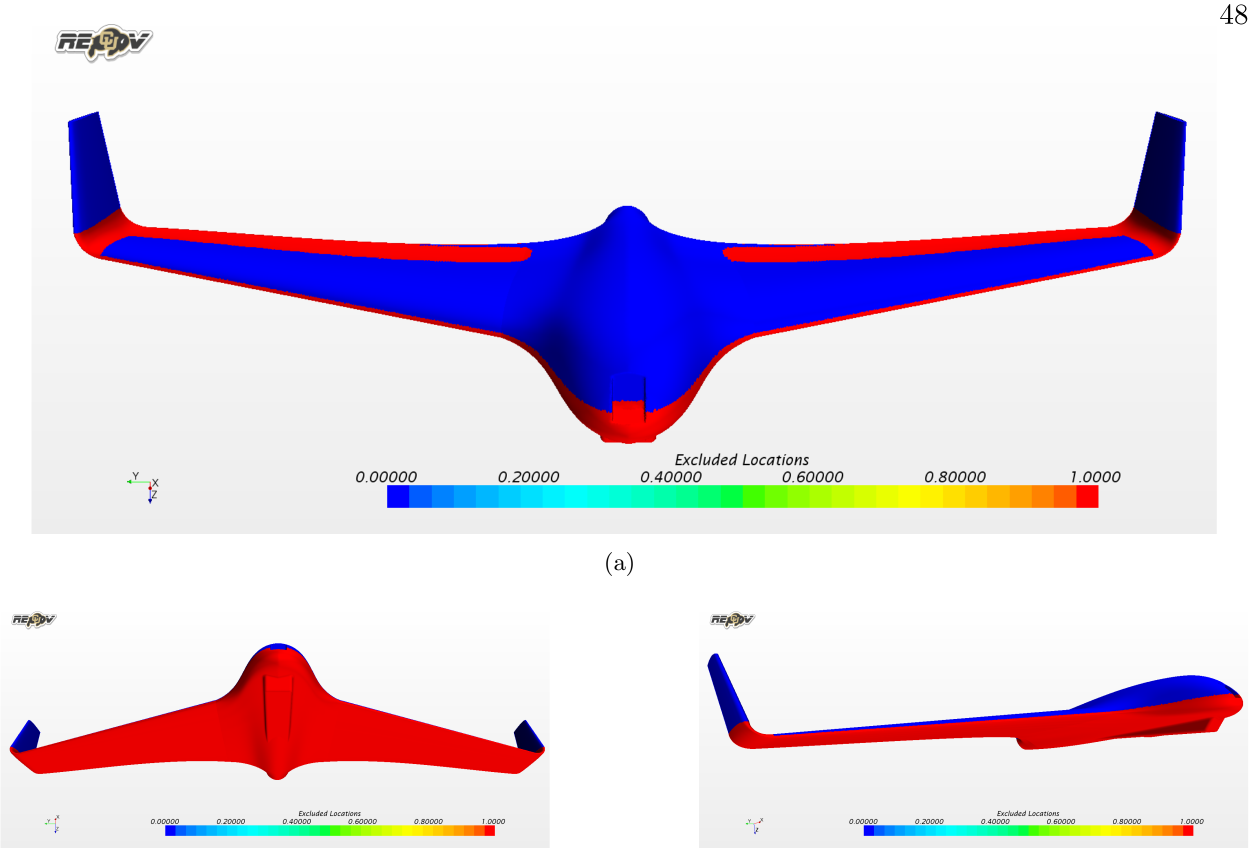
With an α and β total sensitivity for each point on the aircraft, it is now possible to sort the points to start determining port locations. A total of 16 port locations will be chosen: eight that are better suited for α and another eight that are better suited for β. There are two main reasons for choosing 16 locations: 1) a large number of sensors allows for some to be ignored for the purpose of determining how many sensors are actually required for the accuracy goal, and 2) extra sensors also serve as a buffer against hardware issues (such as sensors not sealed properly, or sensor measurement dropouts in flight).

Point 67914 is located near the very front of the leading edge of the wing. The mean pressure coefficient and $(\nabla \mathrm{C_p})_\alpha$ for this point is shown in Fig. 3.6. The plot of $\mathrm{C_p}$ shows a maximum value. This maximum occurs because the stagnation point is either above or below point 67914 depending on the value of α. The drawback to having a maximum (or minimum) is that in the vicinity of the extrema, the gradient will be extremely small. This is evidenced by the gradient in Fig. 3.6 (b). There is clearly a range of α where the gradient (i.e. the sensitivity) is close to zero. Sensor errors will have a more adverse effect in the ranges of low sensitivity, so it is advantageous to avoid locations that experience extrema for the $\mathrm{C_p}$ responses.

To avoid selecting locations with extrema, some areas of the Skywalker were excluded from consideration. The excluded zones are displayed in Fig. 3.10. Based on the minimum α used during simulation, the stagnation point will hit some maximum high point just above the leading edge. Locations below this high point are excluded. This ensures no locations will experience extrema. As the Skywalker is a belly-lander, all locations on the underside of the aircraft are also excluded from consideration. This was done to avoid damage to the ports or clogging due to dirt. The transition area between the wing and winglet was also excluded due to possible flow irregularities or vortices. Portions of the trailing edges were excluded due to the way the extrema exclusion was applied.

With the exclusion zones in place, the total α sensitivities are reranked. The $\mathrm{C_p}$ response for Port 1 (point 20241 in the simulations) is shown in Fig. 3.11 (a). At the low end of the α range, the slope in the α direction is minimal, which is also seen in Fig. 3.11 (b). However, the gradient is never actually equal to zero, and the gradient magnitude increases along with α. As expected though, the gradient in the β direction, shown in Fig. 3.11 (c), is nearly zero the entire time, and the gradient magnitude is an order of magnitude less than that in the α direction. The remaining seven locations for α exhibit very similar $\mathrm{C_p}$ responses.

When ranking the locations based on the total β sensitivity, none of the chosen port locations experienced extrema, hence the reason why no exclusion zones were applied to the winglets. The $\mathrm{C_p}$ response for Port 9 (point 35104) is presented in Fig. 3.12 (a). It is easily noticed that the gradient magnitude in the α direction is significantly smaller than for Port 1, but the gradient magnitude



(a)

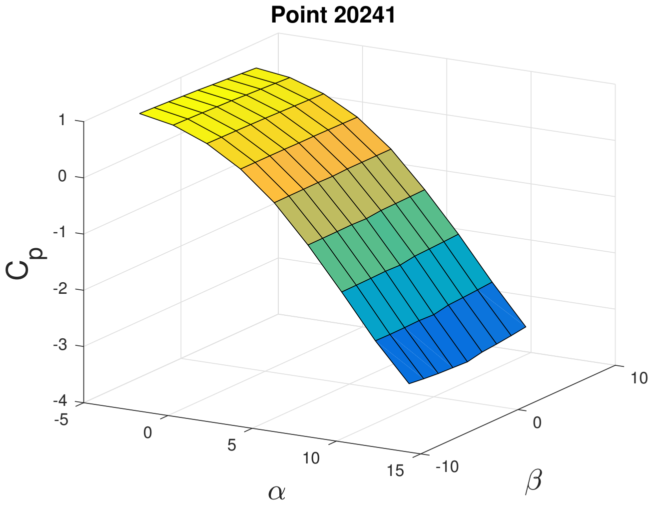
(b)

(c)

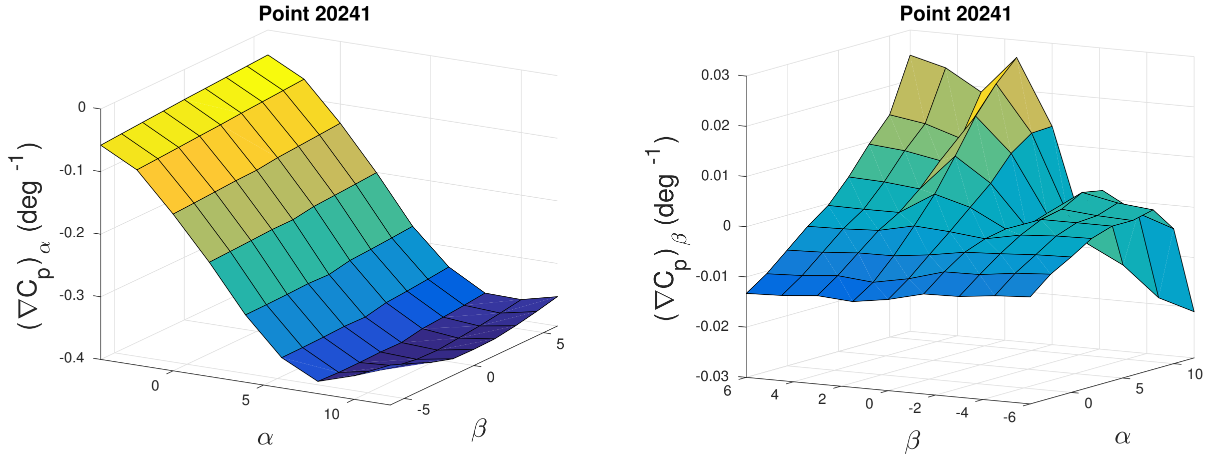
Figure 3.10: Areas of the Skywalker excluded from consideration for port locations. Red areas are excluded while blue areas are included.

in the β direction is an order of magnitude larger than in Port 1. Additionally, the minimum value for $(\nabla \mathrm{C_p})_\beta$ occurs at one of the far corners in the simulated (α,β) space. During more typical flight conditions, the sensitivity is much greater. The remaining locations on the winglets experience similar responses.

The final locations for ports that are more sensitive to α are shown in Fig. 3.13, while the ports that are more sensitive to β are shown in Fig. 3.14. As is displayed in both figures, all of the ports are slightly set back from the forwardmost portion of the respective leading edges. For the ports on the wings, a minimum distance of 5 cm between each port was specified, whereas for the winglets, the minimum distance was set to 3.75 cm. The ports are also symmetric about the centerline of the aircraft. For the α ports, the top 4 locations ranked by total sensitivity whose



(a) Mean pressure coefficient



(b)

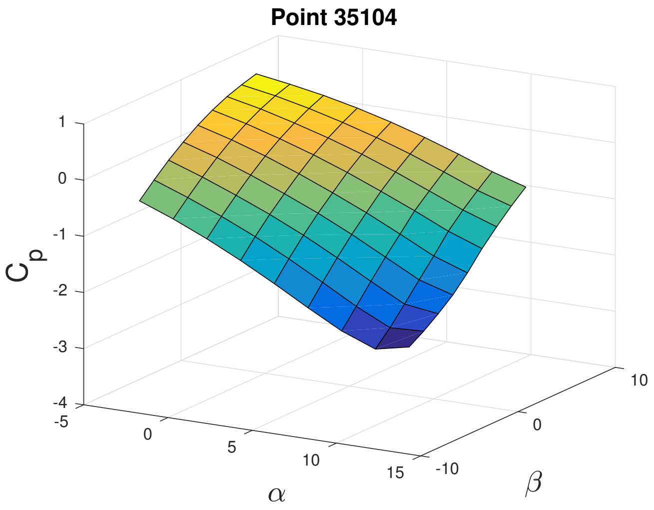
(c)

Figure 3.11: Mean C_p as a function of α and β (a), along with the gradient of the mean C_p in the α (b) and β (c) directions for Port 1.

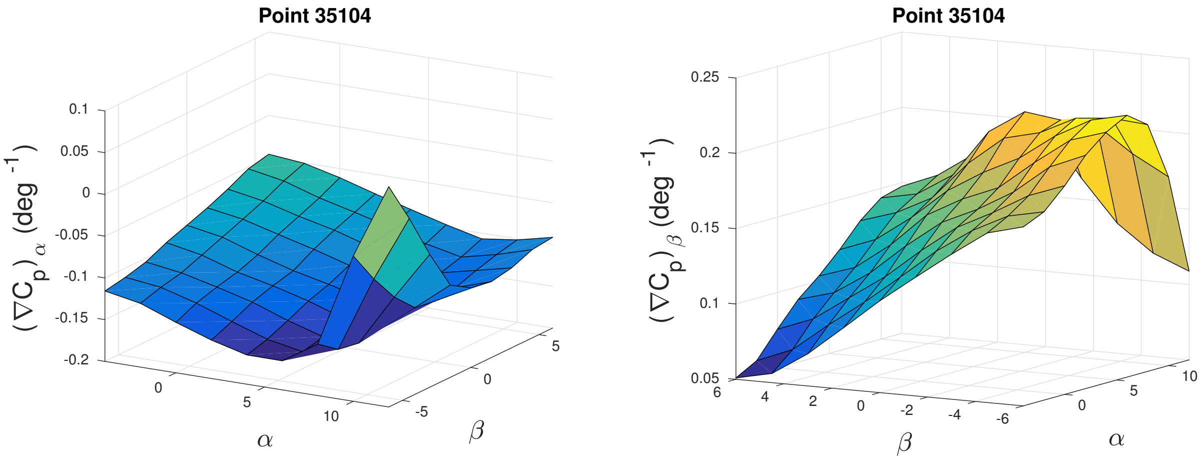
symmetry points also met the minimum distance condition became Ports 1-8. Ports 9-16 were chosen in the same manner.

3.2.4 Stagnation and Static Pressure Ports

The simulation results obtained through CFD are also used for establishing suitable locations for stagnation and static pressure ports. Stagnation and static ports are central to how pitot-static probes work, and all commercially available MHPs have them. The stagnation pressure is the



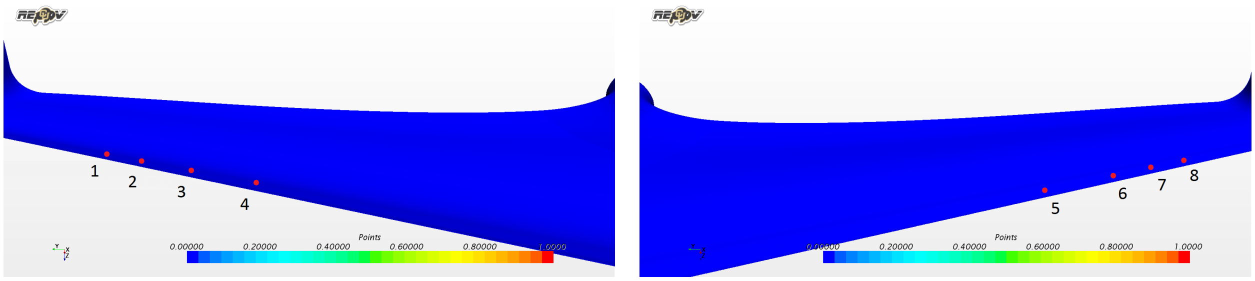
(a) Mean pressure coefficient



(b)

(c)

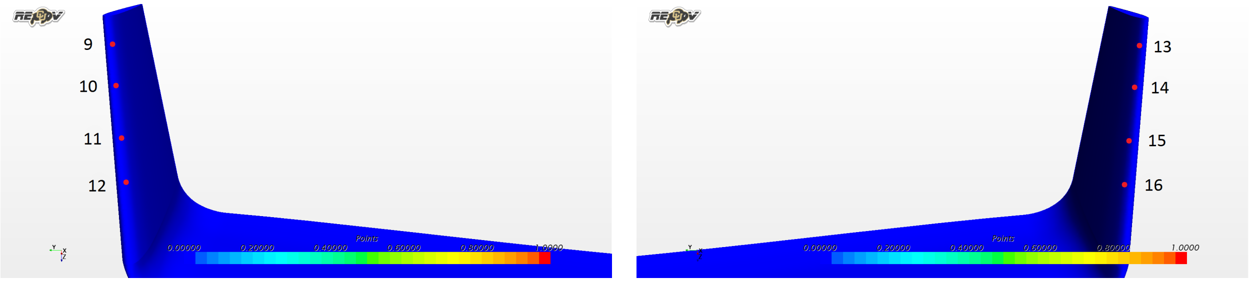
Figure 3.12: Mean C_p as a function of α and β (a), along with the gradient of the mean C_p in the α (b) and β (c) directions for Port 9.



(a) Right wing

(b) Left wing

Figure 3.13: Locations of Ports 1-8 which are better suited for the determination of α.



(a) Right wing

(b) Left wing

Figure 3.14: Locations of Ports 9-16 which are better suited for the determination of β.

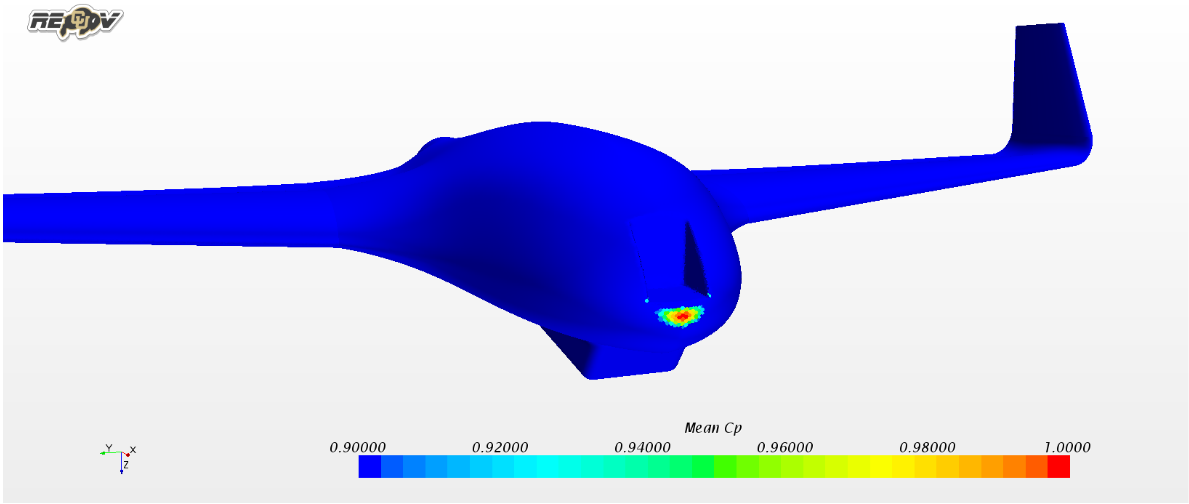


Figure 3.15: Mean C_p at trim condition with scale modified to emphasize locations equal to the stagnation pressure.

pressure of the fluid isentropically brought to rest, and is equal to the dynamic plus static pressure. As such, the value of C_p is equal to 1 at stagnation points. The distribution of the mean C_p at trim conditions $(2°, 0°)$ is displayed in Fig. 3.15. The color scale has been modified to highlight locations where C_p is close to, or equal to, 1. The exact stagnation point is a function of both α and β; as such, only trim condition was utilized for determining the stagnation point. The center of the high-C_p bubble, where $C_p = 1$, was therefore chosen for the stagnation port.

To find suitable static ports, areas with $C_p = 0$ were searched for. A distribution of C_p at trim condition with the color scale tweaked to emphasize areas around $C_p = 0$ is presented in Fig. 3.16. There are numerous different bands across the aircraft where the pressure equals the freestream pressure. Areas towards the rear of the aircraft are excluded from consideration due to

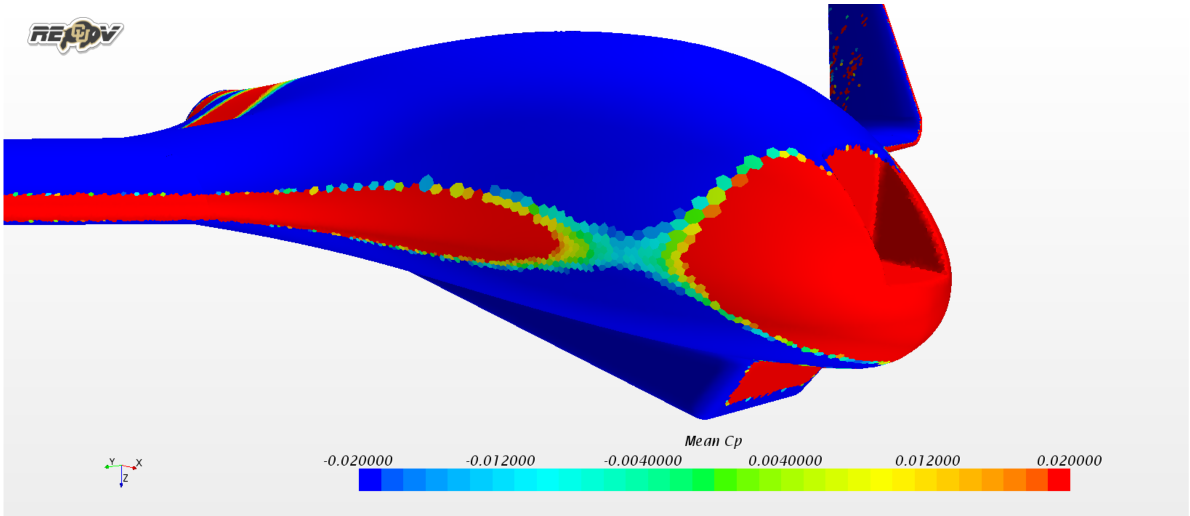


Figure 3.16: Mean C_p at trim condition with scale modified to emphasize locations equal to static pressure.

the proximity of the control surfaces (this would change the measured pressure based on the control surface deflection). Out of the remaining suitable areas, locations that require the least amount of pneumatic tubing are typically chosen [19]. As such, the forwardmost location on the $C_p = 0$ band from the wing is the selected port location. For reference, the port can also be seen in Fig. 3.21.

3.3 PTH Circuit Board Assembly

The sensor chosen for measuring the pressure at the selected ports is the MS8607-02BA01 from Measurement Specialties. The MS8607 [68] was chosen for its ability to measure pressure, temperature, and humidity (PTH); its low cost (∼$10 per sensor); and high resolution for a MEMS-style sensor at that price point. This is also an absolute pressure sensor; while a differential pressure sensor can result in greater accuracies, they are significantly larger than the MS8607. Space is at a premium within the wings of the Skywalker, and a smaller sensor package was given higher priority. Additionally, differential pressure sensors would require running pneumatic tubing from the wings to the fuselage as the measured pressures are referenced against the static pressure. With the current setup, only electrical wires are passed from the wings to the fuselage, allowing the connections to be simpler. Relevant technical specifications can be found in Table 3.3.

The circuit boards were designed to house four MS8607 sensors each (this is related to why eight port locations were chosen per wing). The circuit diagrams showing the layout and electrical connections are shown in Fig. 3.17. A TCA9548A from Texas Instruments [26] handles the data streaming over an I^{2}C connection from the MS8607 sensors. There is also an MPU-6050 6-axis inertial measurement unit (IMU) manufactured by InvenSense [28]. While the IMU is not directly related to the distributed FADS project, it will eventually be used to investigate differences between IMUs distributed with the pressure sensors versus a standalone IMU in the fuselage (currently where IMUs are positioned during flight). The assembled circuit board is laid out in Fig. 3.18.

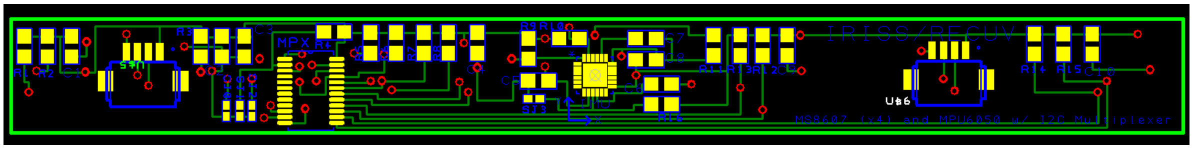
The PTH sensors need to be sealed individually before the circuit boards can be installed into the aircraft. The process of sealing the circuit boards is outlined in Fig. 3.19. Two layers of acrylic are used to seal the sensors individually, and Tygon tubing is then connected to brass tubing in the top acrylic layer.

Each circuit board is tested upon completion of the sealing process. Short sections of Tygon tubing are attached to each port on a circuit board. The free end of the tubing is then sealed with silicone, creating an airtight reservoir that an individual sensor is connected to. With the Ideal Gas Law, it is known how air in a sealed reservoir will react to changes in temperature. The circuit boards are stored in a refrigerator until they reach ambient temperature ($\sim 5°$ C). With the boards at a lower temperature than when the tubing was sealed, the pressure inside decreases. When the circuit boards are removed from the refrigerator (and start warming up), the pressure will increase.

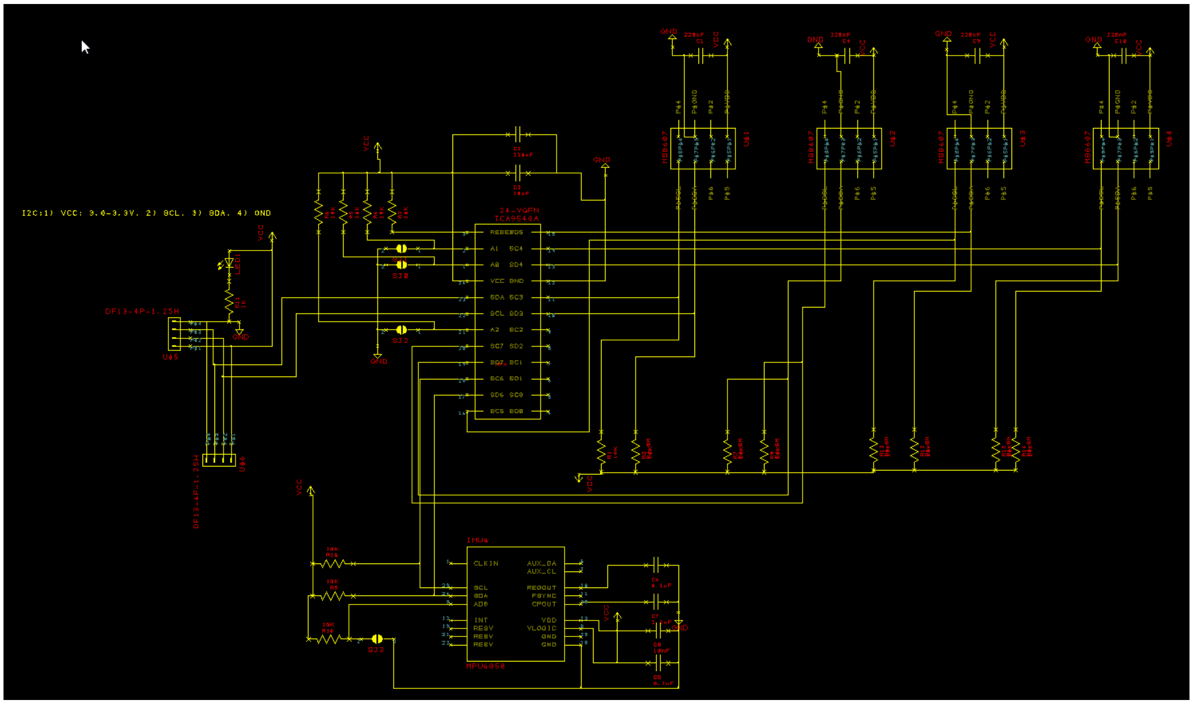
Both trends are displayed in the test results shown in Fig. 3.20. Boards 001 and 004 were both sealed and tested together, while several other boards were left open to the environment and

Table 3.3: Technical specifications of MS8607-02BA01 PTH sensor

Operating range	10 to 2000 mbar, 0% to 100% RH, -40 to 85 °C
High resolution mode	1.6 Pa, 0.04% RH, 0.01 °C
Absolute accuracy (25 °C)	±200 Pa, ±3% RH, ±1 °C
Relative accuracy (25 °C)	10 Pa
Response time	< 5 ms pressure, 5 s RH

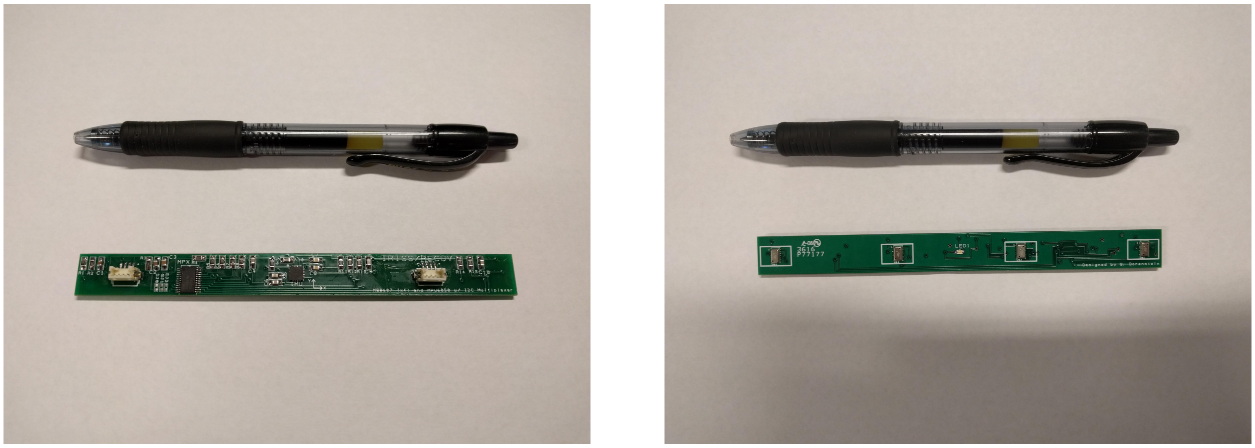


(a) Front



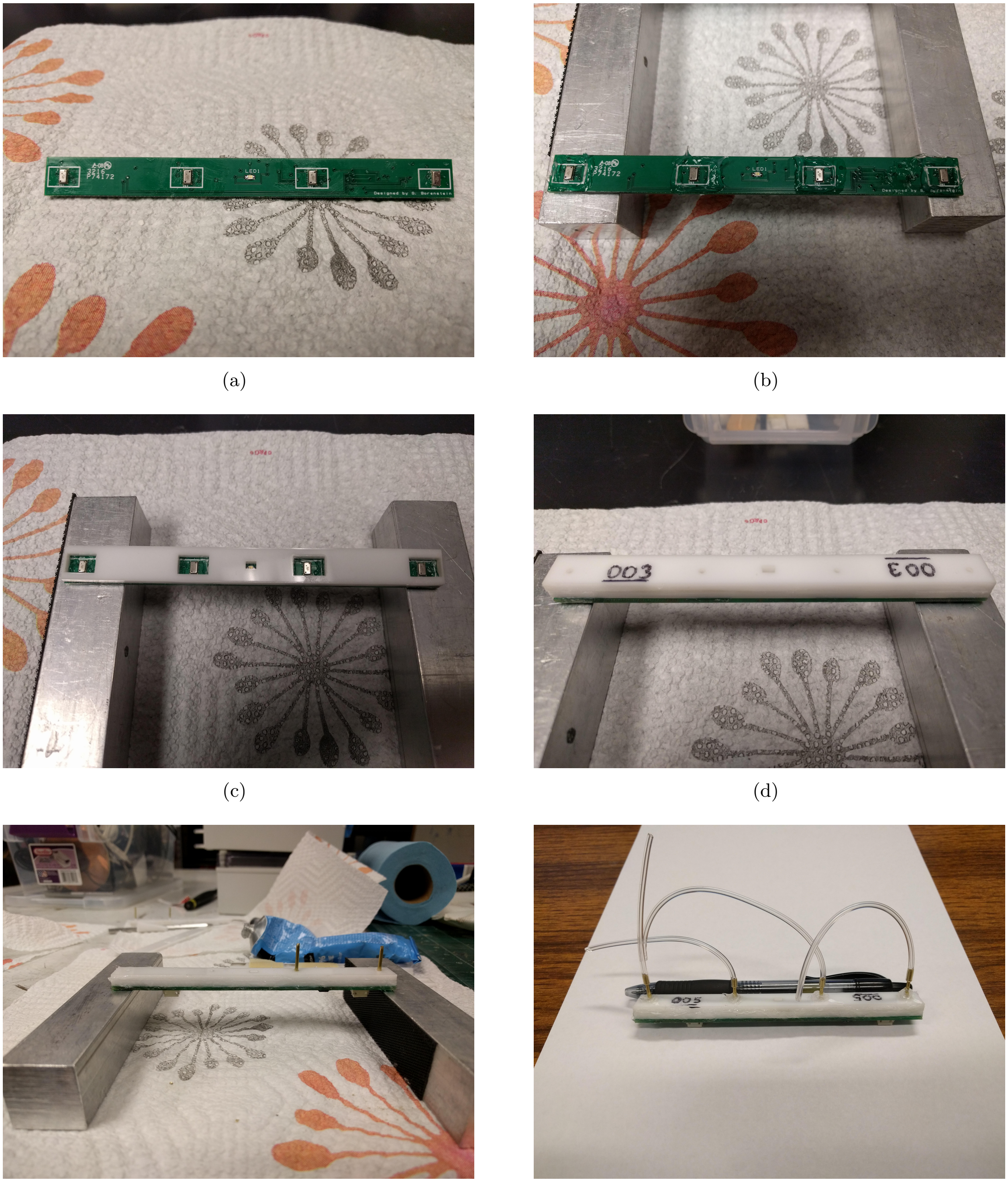
(b) Back

Figure 3.17: Circuit diagram for the sensor boards.



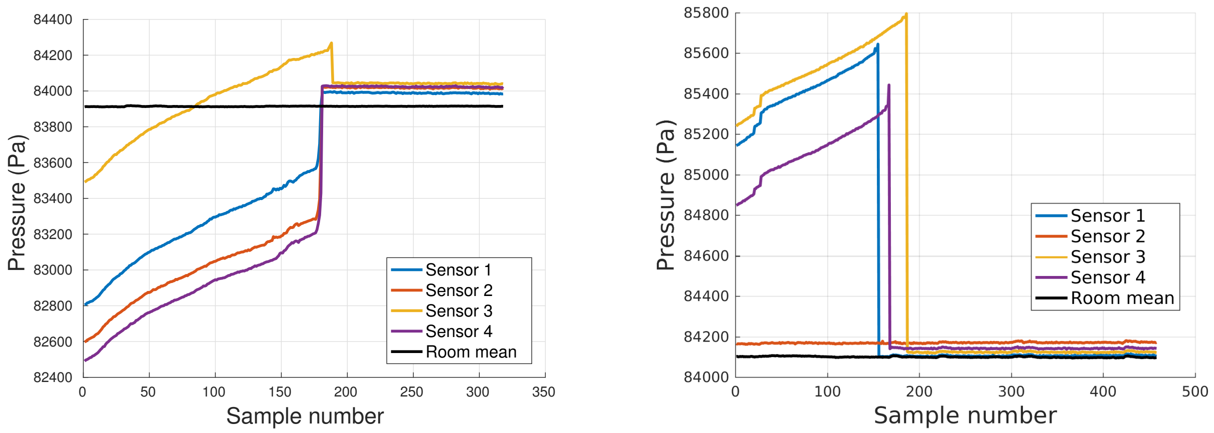
(a) Front (b) Back

Figure 3.18: Finished circuit boards with components soldered on. The black square in the center of the front side (a) is the IMU, while the four silver components on the back (b) are the PTH sensors. Boards are 132 mm long, 13 mm wide, and 2 mm thick.



(a) (b) (c) (d) (e) (f)

Figure 3.19: Process of sealing the circuit boards. Boards are cleaned (a) before applying silicone in a ring around each sensor (b). The first layer of acrylic (c) is then laid on top of the board. A second layer of acrylic (d) is sealed to the first layer with silicone. Small brass tubes (e) with an outer diameter of 1.6 mm are then glued with cyanoacrylate glue to the ports in the top acrylic layer. Tygon tubing (f) with an inner diameter of 1.2 mm is then attached to the brass tubes.



(a) Board 001

(b) Board 004

Figure 3.20: Plots showing seal testing process.

averaged together for the ambient pressure. All four sensors on board 001 start at a pressure lower than ambient, while the pressure steadily increases as time passes and the boards warm up. The instantaneous change to ambient pressure exhibited occurred when the tubing was cut open. Cutting the tubing serves as another check: if the board was not sealed properly, cutting the tubing would have no effect on the pressure. This is seen with sensor 2 of board 004: cutting the tubing made no change to the recorded pressure. Based on these results, it is deemed that board 001 was sealed properly, while sensor 2 of board 004 was not sealed properly. Five boards were assembled and sealed. All four sensors from board 001 are sealed well, but only three sensors from each of the remaining boards are suitably sealed.

3.4 Hardware Integration

Small holes were drilled in the fuselage at the static and stagnation port locations. 3/64-inch (1.2 mm) inner diameter Tygon tubing was then installed through the drill holes. This is the same size tubing the Aeroprobe MHP uses. Smaller diameter ports allow for finer resolution measurements, and the port size has minimal impact on the accuracy [50]. Port sizes as small as 0.76 mm have been used with FADS in the wind tunnel [34]. Unfortunately, the port size relative to the body does have an impact on the maximum incidence angle at which the error in the total

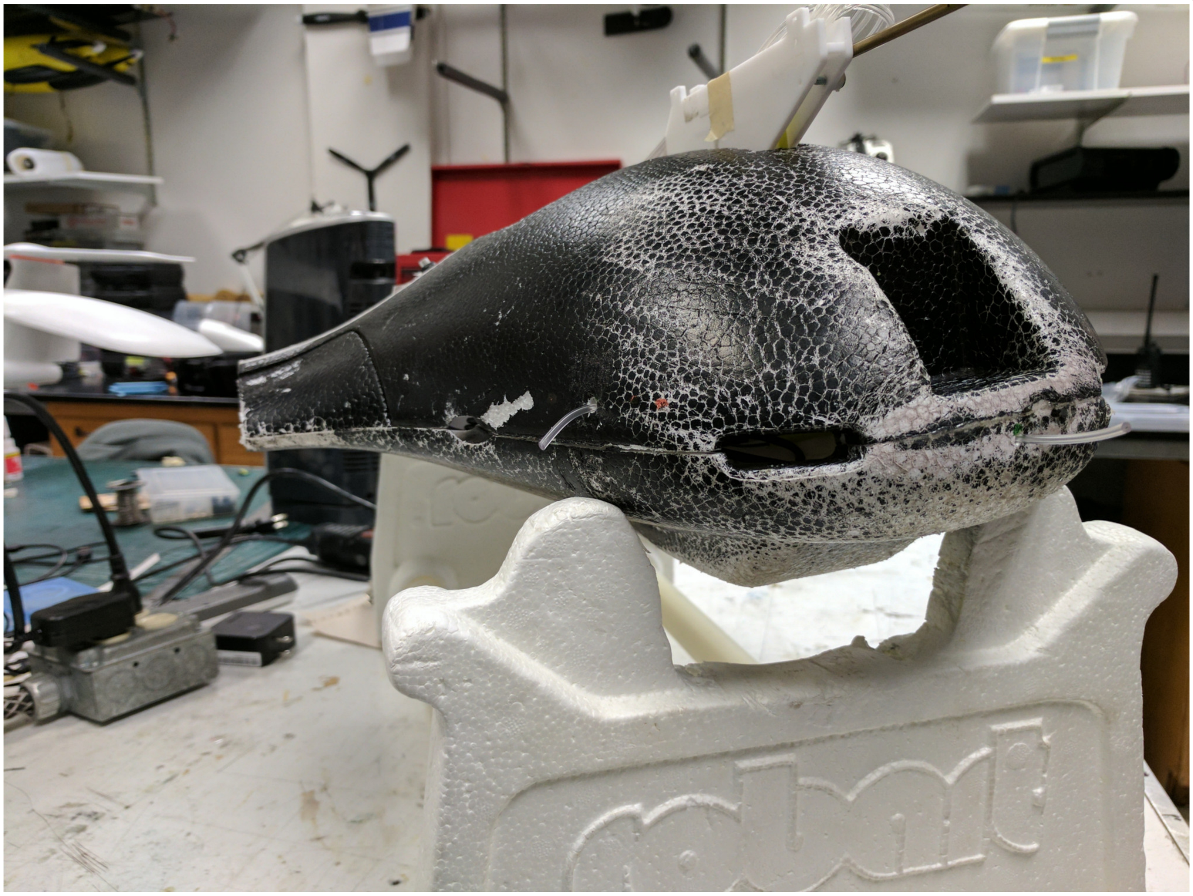


Figure 3.21: Early integration of the stagnation (center) and static (shown on left, hidden on the right) ports on the nose of the Skywalker UAS.

pressure measurement is within 1% [18]. Future work should investigate the importance of port sizing.

Installation of the tubing for the fuselage is shown in Fig. 3.21. Silicone sealant was used to fill in the gap between the tubing and drill hole. The excess tubing is then cut to be flush with the aircraft surface. A single stagnation port is shown on the front of the nose. The stagnation tube is routed to the fuselage pressure board, where it splits and feeds into two pressure sensors. One of the static ports, located aft of the stagnation port, is clearly visible, with a second static port symmetric about the centerline. Each static port has its own dedicated sensor. It is important to note that there is no real airflow in the tubing, so the humidity and temperature readings for the fuselage pressure board are ignored.

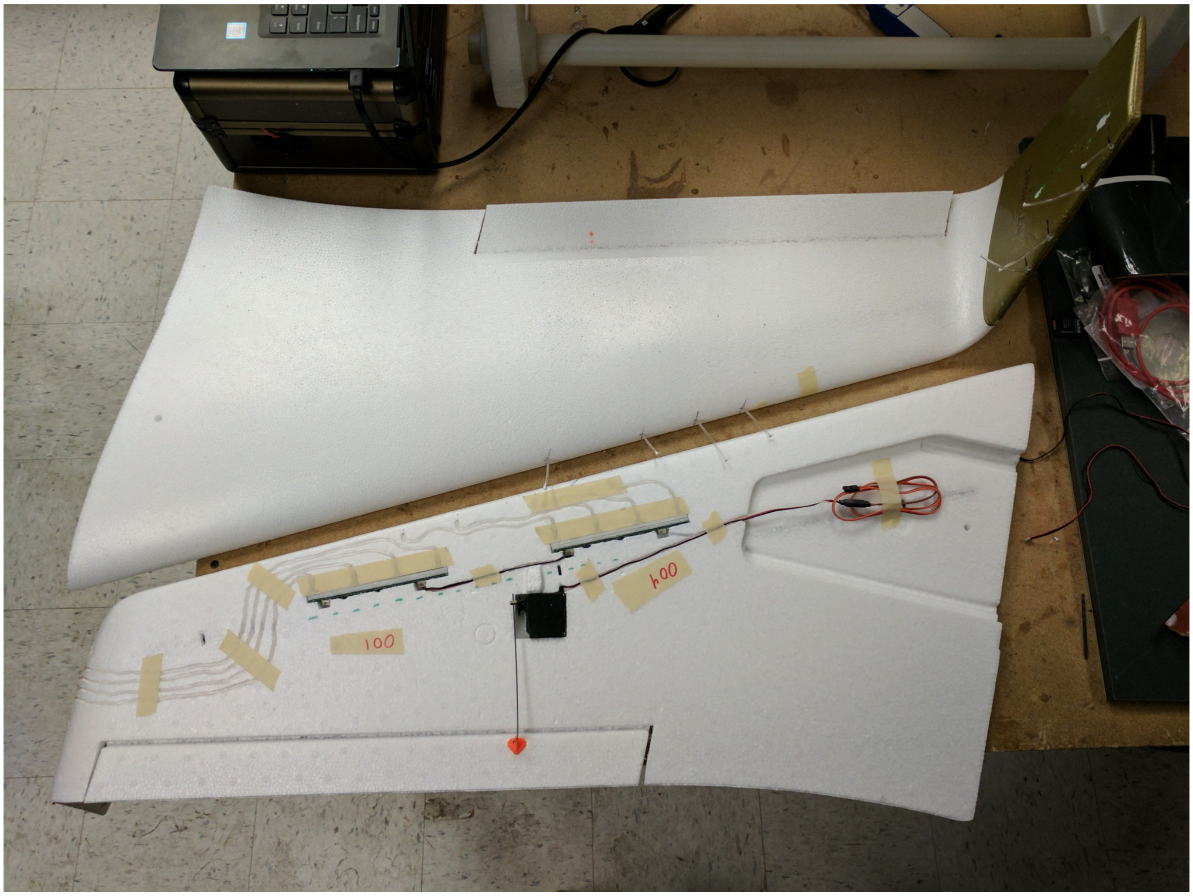


Figure 3.22: Early integration of the sealed circuit boards, tubing, and miscellaneous hardware into the wings.

On the X-8 Skywalker, there is an internal spar that runs parallel to the leading edge of the wing, between the servo and the pressure boards. This spar, and the fact that the wing is thickest in that area, necessitates the installation of the pressure boards close to the leading edge of the wing. The winglets are too small to house a pressure board, so both are installed near each other on the wing. The downside to the space constraints is that the sections of tubing for ports on the winglet are significantly longer than the tubing used for ports on the main part of the wing. Assuming incompressibility and that the change in pressure travels through the tubing at the speed of sound, the increased tubing length introduces a delay of only 1.5 ms.

The pressure board and tubing are all installed on the underside of the wing for simplicity and to maintain aerodynamic efficiency. Individual channels for the tubing are carved into the

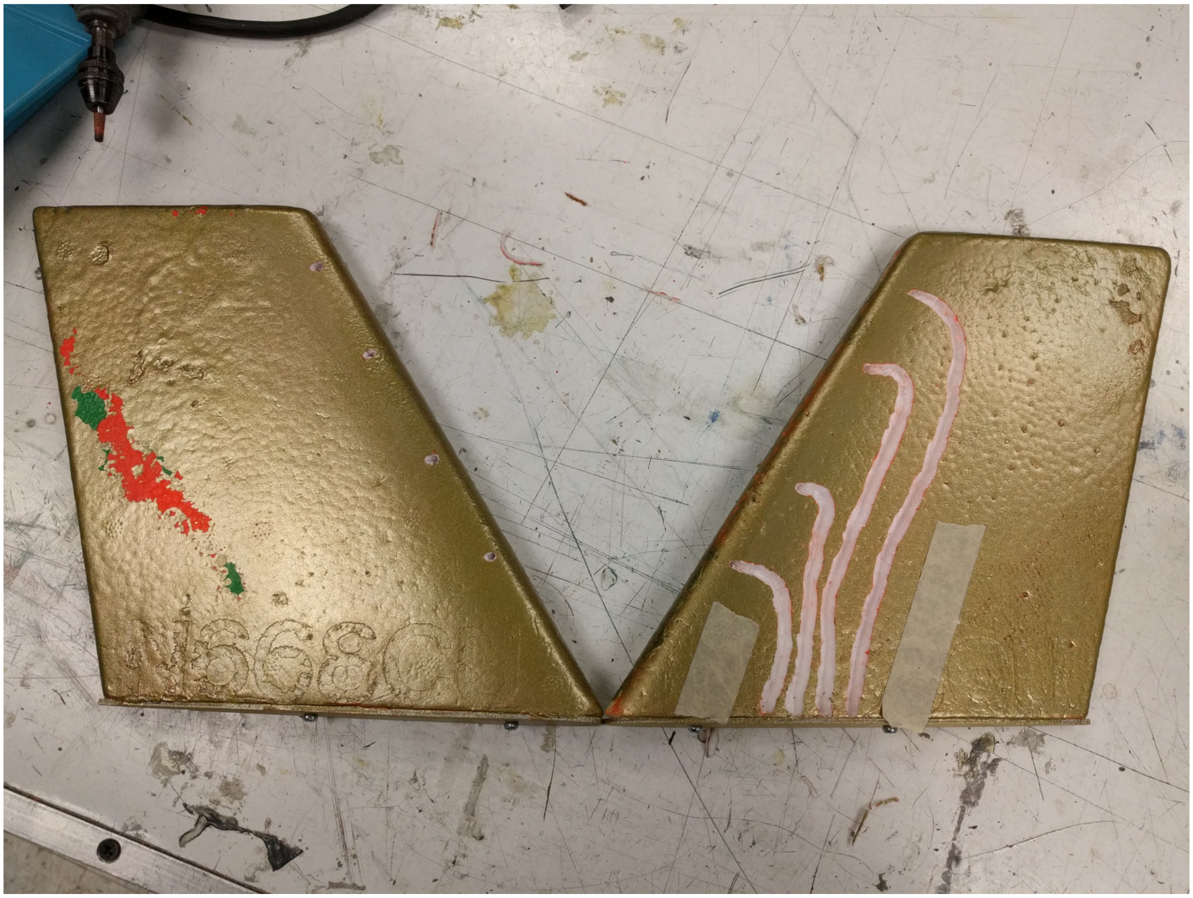


Figure 3.23: Early work on the sideplates. Note the channels created for routing tubing from the physical port locations to the sealed circuit boards installed in the wings.

wing; this way, the tubing will not affect the aerodynamics of the wing. The tubing and pressure boards are taped over with clear packing tape to maintain a smooth, flat surface on the underside of the wing. The lower wing in Fig. 3.22 shows the underside, displaying the pressure boards and tubing routing. The upper wing shows the top surface of the wing, along with the tubing installed in the chosen port locations, but prior to the tubing being cut flush with the surface. A close up view of the side plates with the channels carved in presented in Fig. 3.23. The interior face of the winglets have the pressure port, while the exterior face has the channels.

3.5 Conclusion

Computational fluid dynamics was used to determine port locations for the flush airdata system on the X-8 Skywalker UAS. The mean pressure coefficient was used to locate static and stagnation locations at a set trim condition. The gradient of the mean pressure coefficient shows the sensitivity to changes in the angle of attack and sideslip. Integrating the gradients yields a "total sensitivity", which was used to sort aircraft locations. While 16 wing locations were chosen, only 12 wing sensors were sealed properly.

Sealing the sensors was more difficult than initially anticipated, and only achieved an 80% success rate. Bringing the sensors closer to the center of the circuit boards would increase the acrylic area at the ends, increasing the chances of a proper seal. Another promising alternative is being able to 3D print a cap that goes over the circuit board, similar to the function the acrylic played. Hardware costs were kept low (∼$300 for pressure sensors, circuit boards, and associated components), but further refinement of the installation method is warranted to bring down the required man-hours and simplify the process.

Chapter 4

Skywalker UAS Flight Results

Calibration of a flush airdata system in a wind tunnel has been briefly discussed in Chapter 1, and described in Chapter 2. However, one of the goals of this dissertation is to calibrate the system during flight as an in-flight calibration could make a FADS a more accessible method of wind sensing. The current chapter will discuss the in-flight calibration. Section 4.1 will outline the flights of the Skywalker UAS, Section 4.2 will summarize the steps for preprocessing the flight data, Section 4.3 explains training of the neural networks, Section 4.4 will go over the results from the flights and compare the FADS with the reference multi-hole probe, Section 4.5 will examine the limitations of the FADS as currently implemented, and conclusions drawn from flight testing will be reviewed in Section 4.6.

4.1 Flights

The Skywalker UAS outfitted with the distributed FADS and the Aeroprobe MHP as the reference instrument was flown on March 22, 2017. Flight testing occurred at the Boulder Model Airport in Boulder, Colorado. Two flights were performed, and each lasted nearly 40 minutes. The ground tracks for both flights are shown in Fig. 4.1.

Data from the takeoff and landing portions of both flights were purposefully ignored. This was approximately one minute of flight time at both the beginning and end of each flight. At the beginning of flight 1, various maneuvers were performed and reserved for validation of the FADS. This validation portion lasted 500 seconds, and yielded 3143 data points. The remainder of flight 1

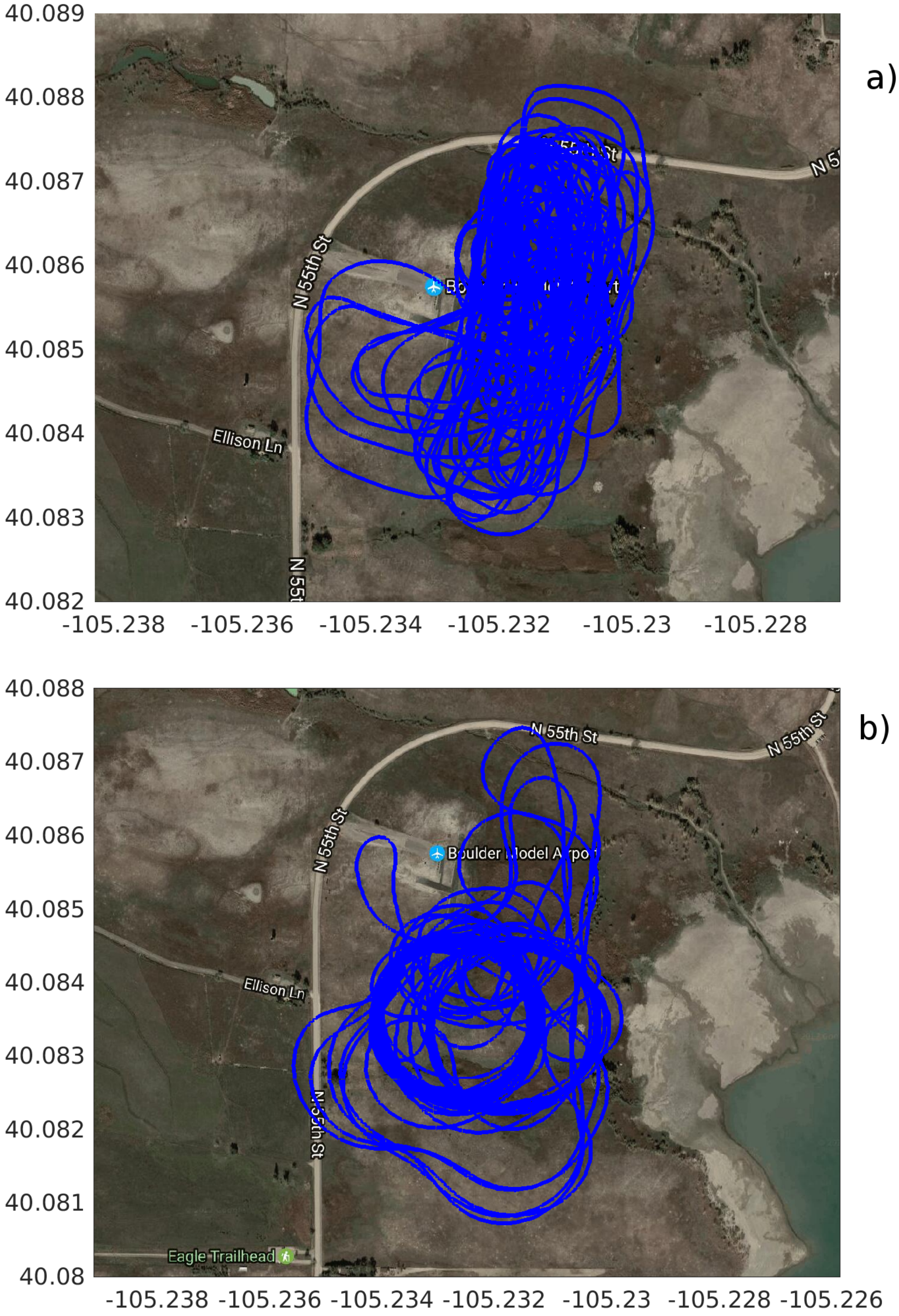


Figure 4.1: Flight 1, top, and flight 2, bottom.

(1700 seconds and 10710 data points) was used for calibration of the FADS. Flight 2 (2160 seconds and 13552 data points) was also used for calibration of the FADS.

4.1.1 Calibration Flights

During calibration, it is important to experience a variety of flight conditions. A larger calibration domain leads to a larger domain in which the FADS is more accurate. Neural networks, the regression technique of choice for the results, typically do not extrapolate well outside of the training domain ([84]; also serves as a good introduction to neural networks). To accomplish the goal of varied flight conditions, the flights were split with approximately 60% of the time under manual control, with the remaining 40% under autopilot control.

The calibration portion of flight 1 was predominately composed of box shaped patterns with the long legs oriented mostly north and south. The airspeed and altitude were varied throughout the flight. Airspeed has a large impact on the angle of attack, so changing the airspeed can lead to changes in angle of attack. Additionally, winds gusts can lead to short duration changes in airspeed, angle of attack and sidelip.

Orbits constituted the majority of flight 2. The airspeed of the orbits was typically set between 18 and 22 m/s. The set altitude of the orbits was typically 1685 m or 1710 m MSL (95 m to 120 m AGL). The direction flown around the orbits was also reversed, with about half the flight time spent travelling clockwise, and the other half counterclockwise.

All of the airspeeds, α, and β measured by the MHP during calibration are shown in Fig. 4.2. The contour lines and color scale represent the number density of data points per bin. Warmer colors represent areas with a higher density of data points. Looking at Fig. 4.2 b), it is apparent that the most common air-relative orientation is about $\alpha = 3^\circ$ and $\beta = 1^\circ$. An aircraft might have a (small) positive α during steady level flight, so this air-relative orientation makes sense. However, it's expected that β would be symmetric about 0°, which is not the case. It is possible the aircraft spent more time turning to one direction than the other, which would explain the bias towards positive values of β. The fact that β is centered about 1° is likely a combination of errors

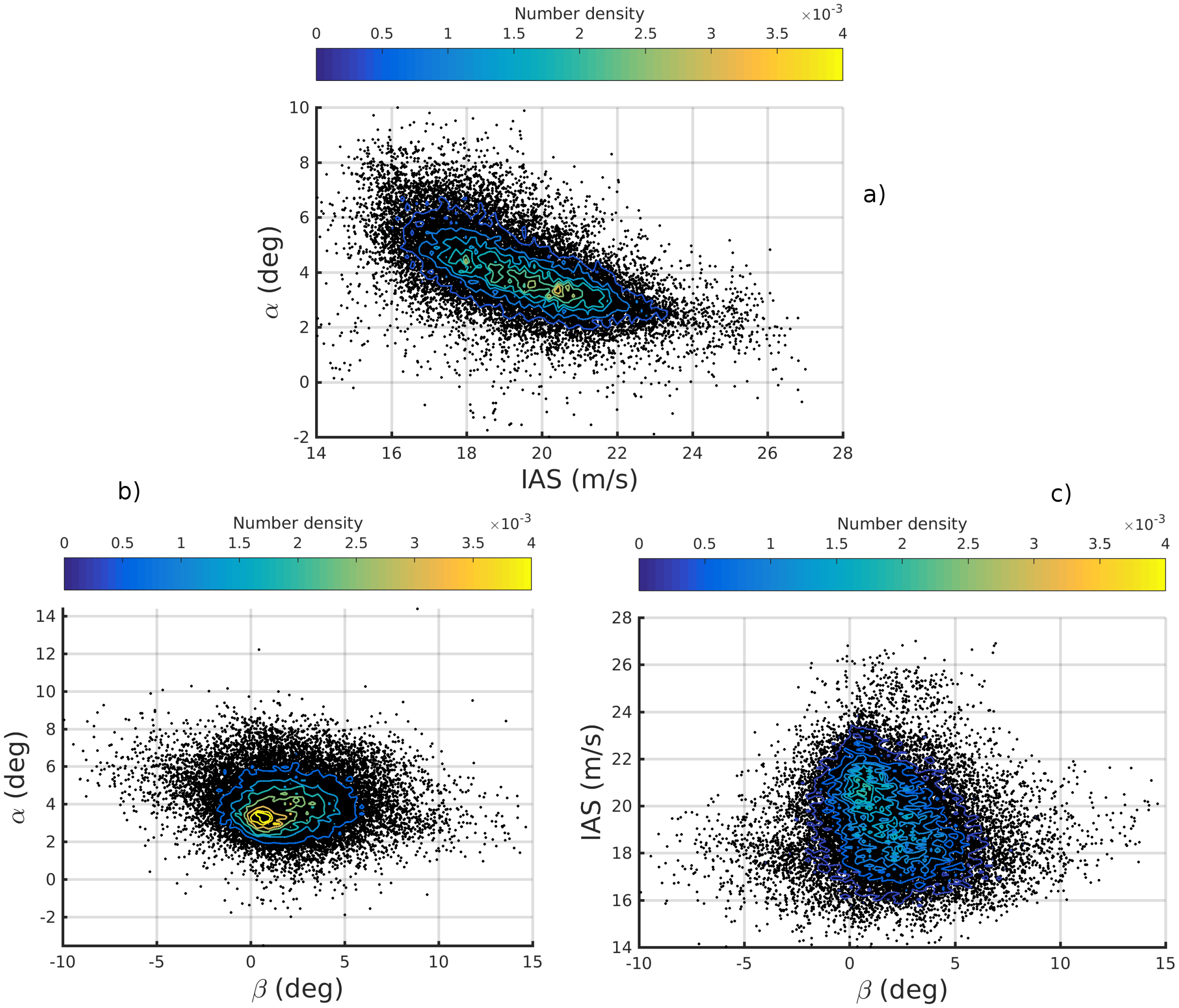


Figure 4.2: MHP data points during calibration.

in the MHP and the MHP being mounted with a small angular offset from the centerline in the horizontal direction. It is important to note that angular offsets from the body coordinate frame are unimportant for wind sensing; only offsets between the wind instrument coordinate frame and the IMU frame matter.

It is also clear that most of the air-relative orientations are within 2° to 6° for α and $-2°$ to 6° for β. With the calibration points as such, it is hypothesized that the FADS will be most accurate within those ranges. Fig. 4.2 a) and c) show the number density of data points in IAS-α space and β-IAS space, respectively.

4.1.2 Validation Flight

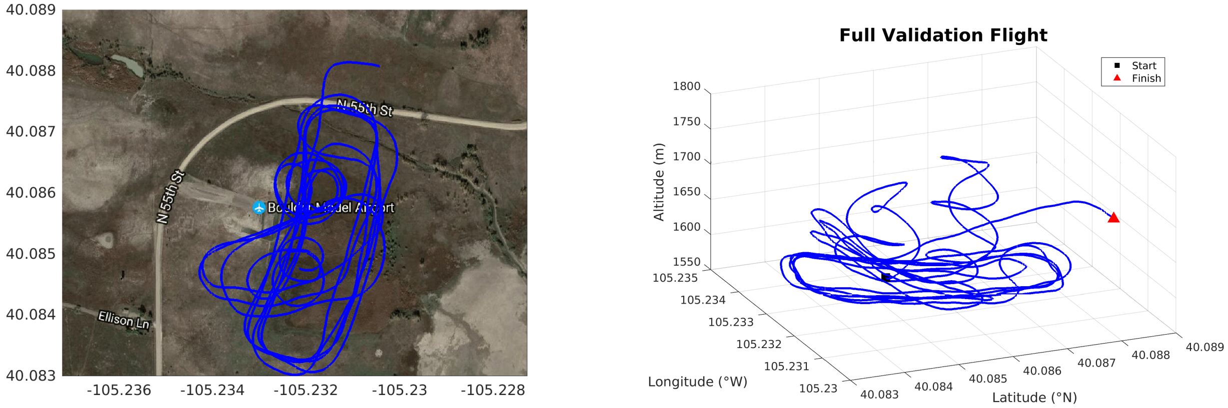
One of the focuses of both the Research and Engineering Center for Unmanned Vehicles (RECUV; [57]) and Integrated Remote and In Situ Sensing program (IRISS; [27]) is to perform targeted meteorological observations from unmanned aircraft systems. The actual flight pattern is determined by the specific mission and aircraft, but commonly involves the "follow me" mode. For "follow me", the UAS ground station is located in the Tracker[1] vehicle, and is equipped with GPS. The ground station operator specifies a GPS waypoint that has an offset from the Tracker vehicle. When the Tracker drives, the waypoint moves with it (though fixed relative to the Tracker unless the operator changes it). The aircraft is constantly chasing this moving waypoint, and the ground track of the aircraft mimics that of the Tracker. As most of the roads driven on during meteorological deployments are straight (and oriented in the cardinal directions), most of the flight time is spent performing transects, though the other common flight pattern is an ascending or descending orbit. A typical flight path is shown in Fig. B.1 in Appendix B.

The validation portion of the Skywalker flight was intended to mimic the previously discussed real world flight profiles. Validation was made up of transects (in the form of rectangles due to space constraints and no "follow me" mode for the Skywalker), ascending and descending orbits, and straight legs with varying altitude. The ground track of the Skywalker during validation is shown in Fig. 4.3 a), while b) shows the 3D trajectory.

All of the IAS, α, and β measurements from the MHP during validation is displayed in Fig. 4.4. The contour lines from the calibration points shown in Fig. 4.2 are also displayed. The scale (and contour levels) are identical, but a different colormap is used for increased contrast. The purpose is to show the concentration of validation data points in relation to the calibration data points. If the validation points are of a comparable distribution to the calibration points, then the errors should be comparable. The validation points show a somewhat similar, but slightly different, distribution compared to the calibration points. In the β-α space, α is shifted towards lower values

---

[1] The Tracker is one of several vehicles that provide support for aircraft operations. The role of the Tracker changes based on the mission and what other vehicles are in use, but is typically "electronically tethered" with the aircraft via the autopilot and ground station.



(a) Ground track (b) 3D trajectory

Figure 4.3: Validation portion of the flights.

compared to the contour lines. In both IAS-α space and β-IAS space, the data points are biased towards higher values of IAS.

Nine distinct patterns are pulled from the validation flight: Spiral 1, Spiral 2, Ellipse 1, Box 1, Box 2, Box 3, Box 4, Leg 1, and Leg 2. In addition to overall validation results, results from individual patterns will also be analyzed in an attempt determine ideal patterns to fly in order to minimize errors. The trajectories for all nine patterns are presented in Figs. B.2 through B.10 in Appendix B.

4.2 Data Preprocessing

The data logging onboard the X-8 Skywalker is handled by a Raspberry Pi 2 Model B computer. With the setup flown, the Pi writes three files: one file with the data from the autopilot (called the "messages" file), a file with only the MHP data (called "aeroprobe"), and a file containing data from the MS8607 sensors (called "pth"). Data are timestamped as it comes into the Pi, and the Pi clock is updated to the system time of a laptop upon boot.

The messages file contains data from the autopilot, including GPS, aircraft attitude, and telemetry data sent to the ground station. There are two timestamps that appear throughout the file: "time_unix_usec" and "time_boot_ms". Most fields use one or the other for timestamping.

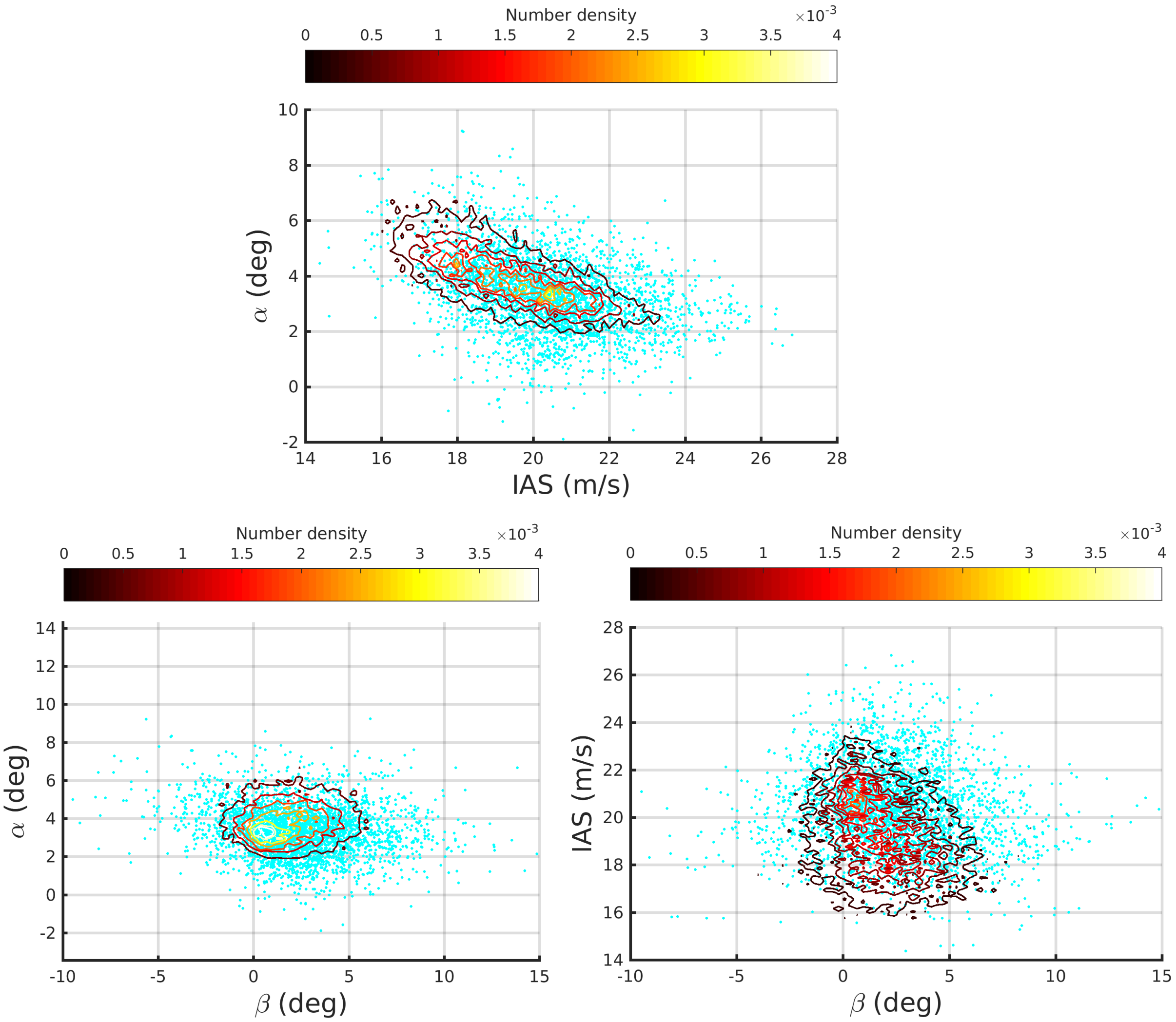


Figure 4.4: MHP data points during valibration.

These timestamps were synced together after the flights to ensure all the data from the messages file would reference a single time.

The recorded GPS altitude was used to determine when takeoff and landing occurred for both flights. As mentioned previously, the takeoff and landing portions were removed (these data are typically ignored in real RECUV/IRISS operations anyway). Data from when the aircraft was on the ground were also ignored, with the exception of roughly 30 seconds on the ground that was used to zero the PTH sensors.

Even with the data from the different files referencing the same time, timing offsets (such

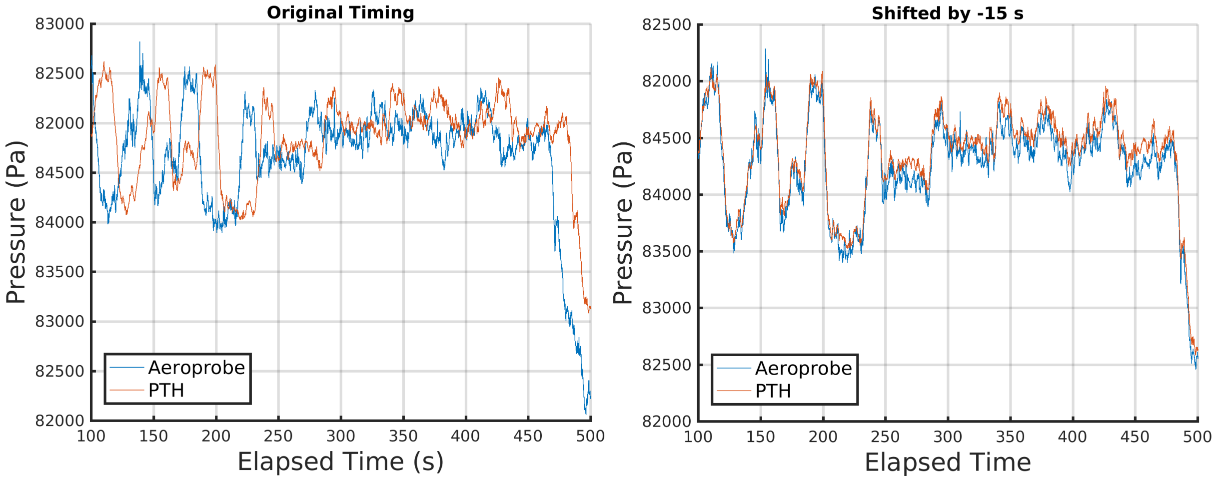


Figure 4.5: Shifting total pressure from FADS to line up with MHP.

as communication delays) still existed and needed to be accounted for. To sync the MHP and PTH data, the total pressure[2] measured by both was compared. As shown in Fig. 4.5, there was an initial offset between the data.[3] The MHP data (logged at 50 Hz) were interpolated onto the same time as the PTH data (6.2 Hz) to allow for direct comparison. The mean squared error was computed between the MHP and PTH data, and is defined as:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{X}_i - \mathrm{Y}_i)^2 \tag{4.1}$$

with total pressure at time i from the MHP represented by X_i, and total pressure from the PTH sensors in the fuselage represented by Y_i. Using a brute force approach, multiple time offsets are applied to the PTH data, and the MSE is calculated for each time offset. The time offset that produced the smallest MSE is permanently applied to the PTH data.

This approach to syncing the data only worked because the MHP and PTH sensors in the fuselage were measuring the same quantity, and the measurements were less than 20 cm apart spatially. The trends in both data sets should happen at the same time (though the actual pressure values are expected to be different). GPS data were synced to the MHP time in the same method,

---

[2] For incompressible flow, the total pressure is equivalent to the stagnation pressure

[3] The shift shown was artificially introduced for clarity to demonstrate the method. Offsets were typically on the order of tens of milliseconds to several seconds

but comparing GPS altitude to the pressure altitude from the MHP. The ground station telemetry data were synced to the MHP through airspeed, and the autopilot's pitot-static data were synced to the MHP through static pressure comparison.

The synced PTH, MHP, GPS, ground station telemetry, attitude, and pitot-static probe data are combined into a single file. All the data are linearly interpolated onto the same time as the PTH sensors, thus producing a single time vector common to all the instruments. The data are then separated into the calibration and validation sets.

Final preprocessing of the flight data includes zeroing the pressure from the PTH sensors and computation of the derived parameters (such as density and indicated airspeed). 30 seconds of PTH pressure data prior to both flights are used to zero the sensors. One of the sensors is chosen as the baseline (any sensor will work). The mean pressure over this interval for each sensor is compared to the mean pressure from the baseline sensor. This offset is then removed from the measured pressure in flight for each sensor.

After sensor offsets have been removed, numerous derived parameters from the PTH measurements can then be computed. Static pressure is the mean of both static port sensors, total pressure is the mean of both stagnation port sensors, and the dynamic pressure is the static pressure subtracted from the total pressure. Indicated airspeed is computed by the standard equation:

$$\mathrm{IAS} = \sqrt{\frac{2\mathrm{q}}{\rho_0}} \tag{4.2}$$

with ρ_0 being mean sea level density,[4] and q is the dynamic pressure, also represented by:

$$\mathrm{q} = \frac{1}{2}\rho \mathrm{V}^2 \tag{4.3}$$

Density is calculated with the Ideal Gas Law, Eq. 1.11, where R is the specific gas constant of air,[5] T is the mean temperature from the sensors in the wings, and total pressure is used for P.[6] Differential pressure for each sensor is calculated by subtracting the static pressure from the

---

[4] 1.225 kg/m^3

[5] R = 287.058

[6] In-flight differences between total and static pressure are on the order of 0.3%

sensor pressure. Finally, data from the four sensors not sealed properly were removed, resulting in differential pressures from only 12 sensors.

4.3 Neural Network Training

Neural networks have been successfully implemented with FADS previously, including flight data from an F-18 [62], a small UAS [54], a mini air vehicle [65], and a wind tunnel model [9]. Hornik, Stinchcombe, and White [22] have shown that a single hidden layer in a multi-layer feedforward network (a class of neural networks) is enough for the network to be a universal approximator, given sufficient hidden units and training data. Thus, neural networks are known to be an excellent choice for use with FADS.

When training a network through backpropagation, Matlab uses the Nguyen-Widrow initialization algorithm for determining the initial weights and biases [43, 48]. There is some randomness in this method, which means re-initializing the network can lead to different results. Additionally, the training data are randomly chosen from the calibration data, which also leads to potentially different results from networks trained on the same data. A basic ensemble of neural networks has already been described in Chapter 2. Basic ensembles are produced again, but only the single most accurate network from the ensemble is used to produce the results.

Numerous training methods are available within the MATLAB environment, including gradient descent, conjugate gradient methods, Levenberg-Marquardt, Bayesian regularization, and resilient backpropagation. The backpropagation method [64] has been available for awhile, and early work focused on improving the method by using a steepest descent method with momentum and a variable learning rate [76]. The scaled conjugate gradient method [46, 6] increases training speed by avoiding the per iteration line search, and is better suited for larger problems thanks to lower memory requirements. Other adaptations to the conjugate gradient method exist that focus on better restarting procedures [53]. Resilient backpropagation [60] uses a local adaptation of the weight update to increase training speed over standard gradient descent methods. Levenberg-Marquardt [21, 42], which employs a nonlinear least squares algorithm, is very efficient for smaller networks (up to a few hundred weights[7]). Bayesian regularization [16, 41] adds a smoothing term to improve generalizability,[8] at the expense of increased computation and memory usage.

A direct comparison between Bayesian regularization (BR), Levenberg-Marquardt (LM), and Scaled Conjugate Gradient (SCG) training functions using the flight data is presented in Fig. 4.6. Basic ensembles with 500 networks were trained utilizing each training function for different hidden layer sizes. The root mean square error (RMSE) is used to compare the performances of the different training functions:

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{\mathrm{Y}}_i - \mathrm{Y}_i\right)^2} \tag{4.4}$$

where $\hat{\mathrm{Y}}_i$ is the estimate produced by the network (or mean estimate from the ensemble), and Y_i is the true value. The reference MHP from Aeroprobe Corporation flown on the Skywalker is taken to be the truth (potential issues with this approach are discussed in Section 4.5). The ensembles are trained with the calibration data, and the RMSE values are computed with the validation data. For inputs, the networks use the 12 differential pressures from the PTH sensors, density, static pressure, and indicated airspeed, and the two outputs are α and β. Additionally, individual errors (ε_α, ε_β, and $\varepsilon_{\mathrm{IAS}}$) are simply:

$$\varepsilon_i = \hat{\mathrm{Y}}_i - \mathrm{Y}_i \tag{4.5}$$

with the notation the same as before. A positive error means the FADS estimate is higher than the measurement from the MHP.

The RMSE using the ensemble mean estimates are represented by triangles in Fig. 4.6, with BR colored blue, LM red, and SCG black. The curves of BR and LM show the methods initially underfitting the data, then reaching a minimum error, before overfitting the data. Due to the smoothing built into Bayesian regularization, this method overfits at a slower rate compared to

---

[7] Number of weights for a single hidden layer feedforward network is equal to (hidden layer size × number of inputs) + hidden layer size + (number of outputs × hidden layer size) + number of outputs

[8] A network generalizes well when it performs as well with novel inputs as it does with training inputs [16]

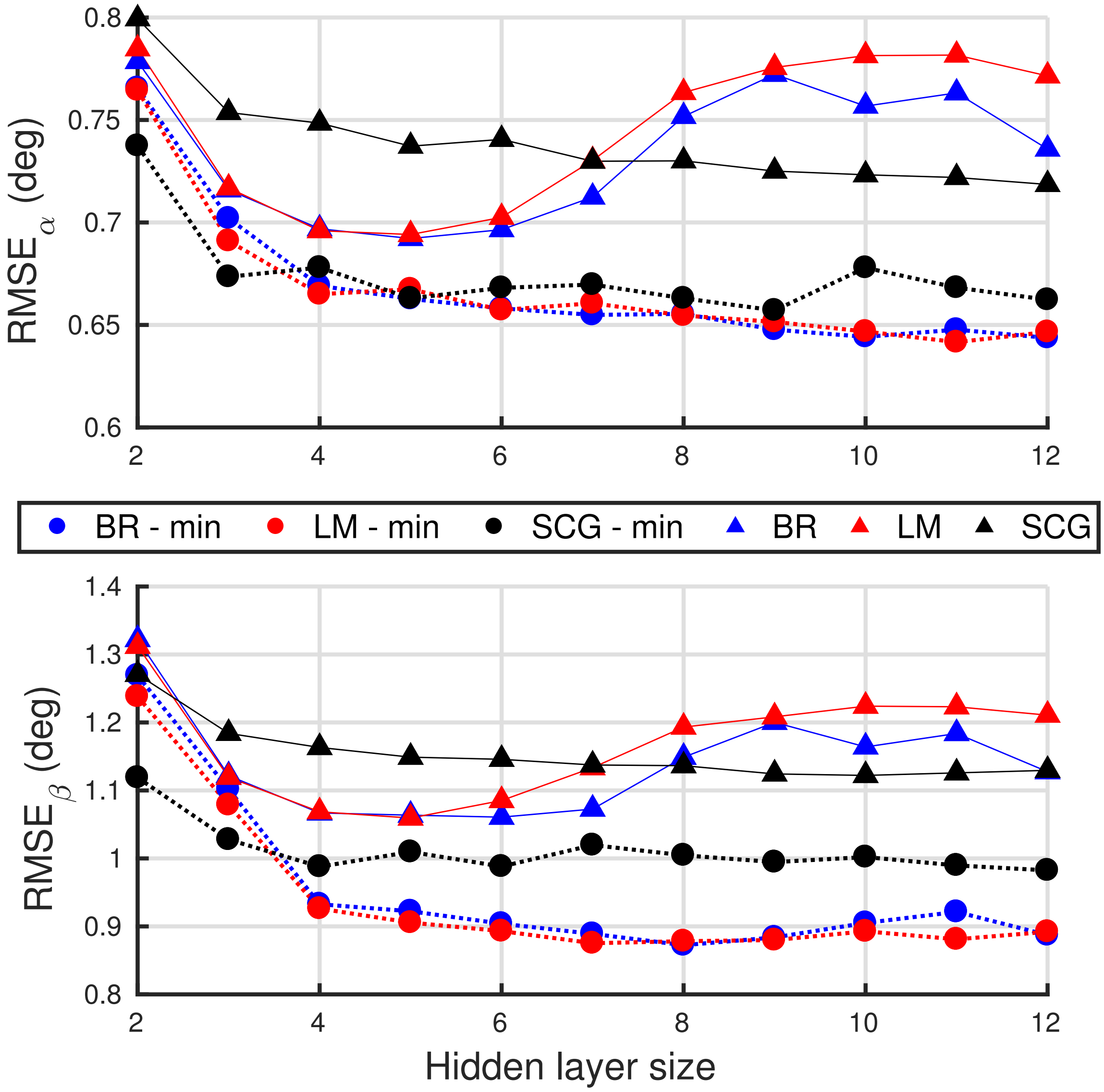


Figure 4.6: Comparison of training functions and hidden layer sizes.

LM. SCG experiences a more gradual decline in the error, and does not appear to be affected by overfitting in the same way as BR and LM, though the minimum error from the SCG ensemble is approximately 5% greater than the minimums with BR or LM. Based on these data, a hidden layer size of 5 using the BR or LM training functions would produce the lowest RMSE, assuming the user wanted a basic ensemble of networks.

The network from the ensembles that produced the minimum RMSE is also visible in Fig. 4.6, using filled circles. The best network using SCG tracks the best networks from BR and LM closely, in terms of RMSE_α, up until a hidden layer size of 9. However, in terms of RMSE_β, SCG is nearly constant, and is around 10% higher than BR and LM. The RMSE for BR and LM are nearly identical, so other attributes of the functions need to be considered when choosing one of those training methods (LM is approximately three time faster, but BR generalizes better).

The minimum RMSE_β happens with a hidden layer size of 8, but the minimum RMSE_α does not occur until a hidden layer size of 10. To determine the hidden layer size, the norm of the RMSE was looked at:

$$\|\text{RMSE}\| = \|(\text{RMSE}_\alpha, \text{RMSE}_\beta)\| = \sqrt{\text{RMSE}_\alpha^2 + \text{RMSE}_\beta^2} \tag{4.6}$$

For the Bayesian regularization and Levenberg-Marquard networks, the minimum $\|\text{RMSE}\|$ value for each hidden layer size is shown in Fig. 4.7. The minimum value for both training functions occurs with a hidden layer size of 9, and both functions produce an identical norm. The network's ability to generalize well is of higher importance than training speed; as such, Bayesian regularization is the training method of choice.

2000 new networks were trained using Bayesian regularization and a hidden layer size of 9 in an attempt to find a network with a lower $\|\text{RMSE}\|$ than reported in Fig. 4.7. Out of the new networks, the best network has $\|\text{RMSE}\| = 1.087°$, a 1.2% reduction compared to the previously plotted networks. Looking at the individual components, $\text{RMSE}_\alpha = 0.651°$ and $\text{RMSE}_\beta = 0.871°$. All subsequent results use this single network.

Regression plots showing how well the network fits the calibration data is shown in Fig. 4.8. A perfect linear fit is shown as a black line, and has a slope of 1 and y-intercept of 0. The slope and y-intercepts of the actual linear fit (in red) are shown on the y-axis. When looking at the linear fit, the network does a better job with β than with α. However, RMSE_α is lower than RMSE_β. This discrepancy is explained by the wider variability in the β estimates. The estimates of β are

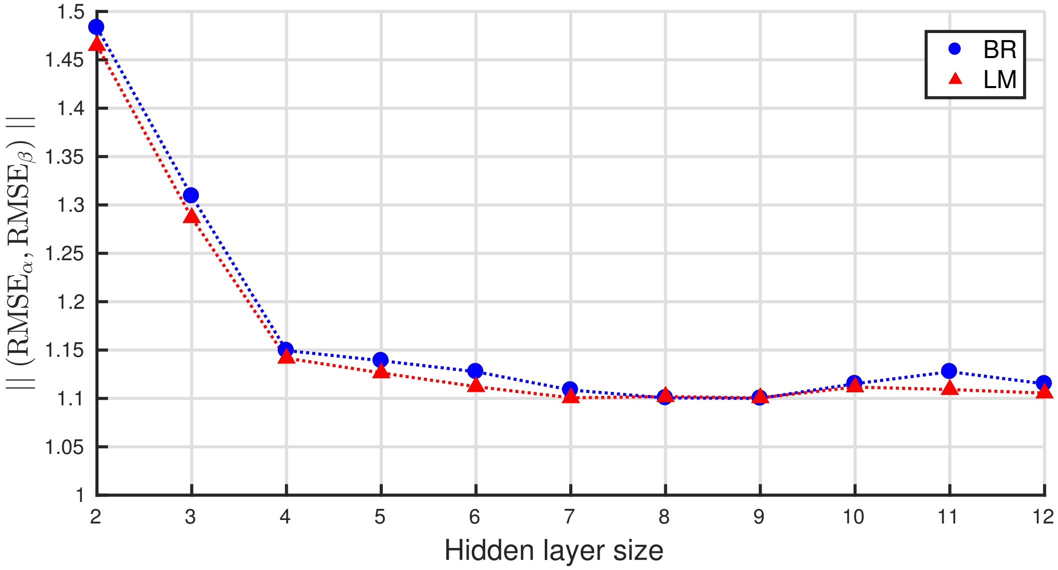
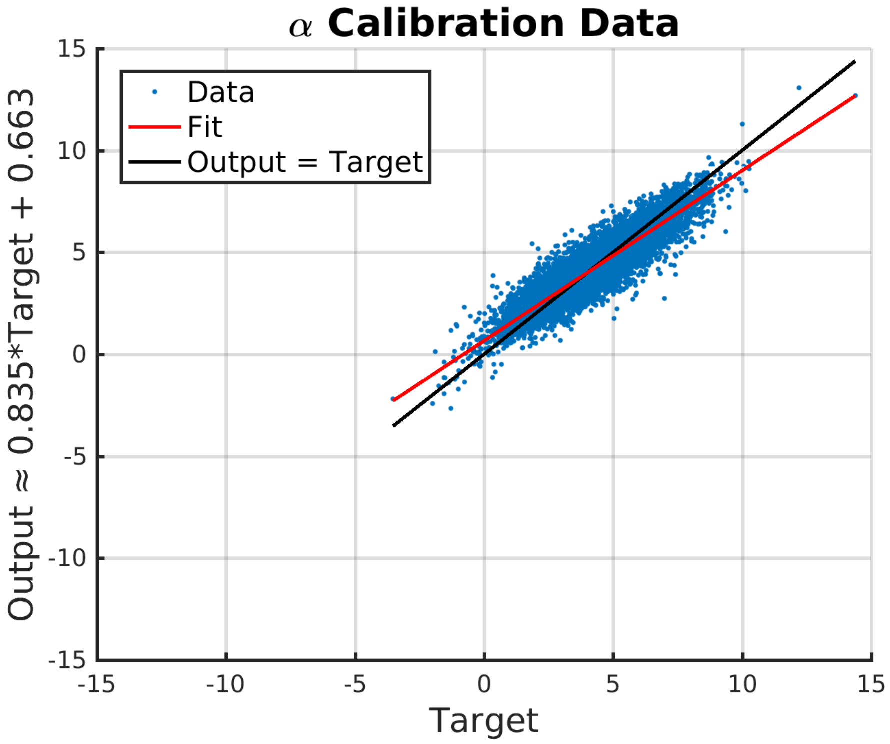


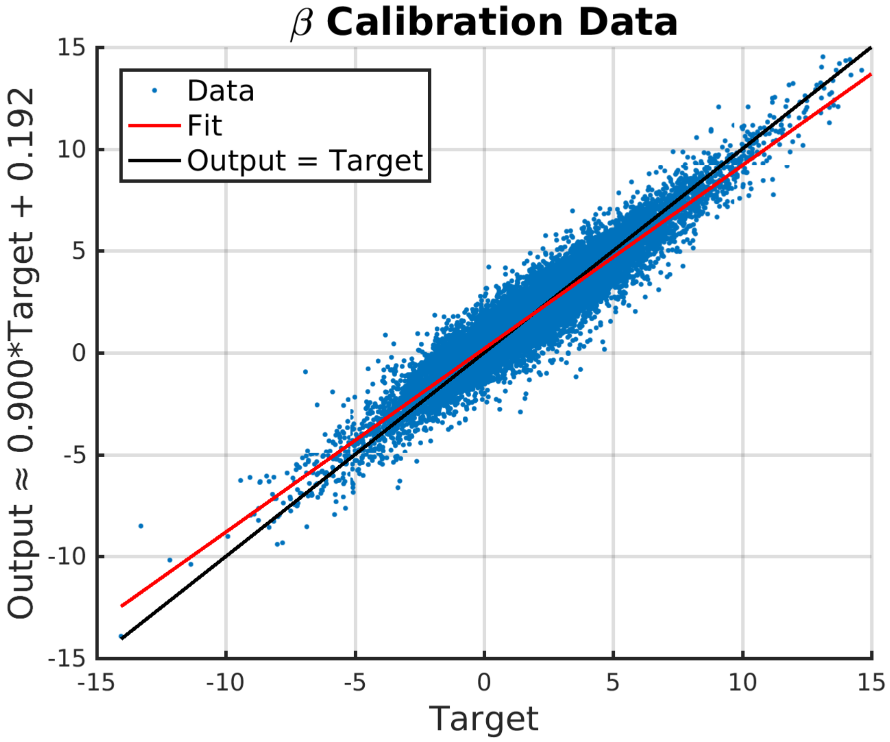
Figure 4.7: Comparing the minimum norm RMSE values for different hidden layer sizes.

centered about a closer to perfect line (slope closer to 1), but are spaced further apart from that line (higher RMSE).

The regression plots and RMSE values only describe how the network performs overall. Knowing how the network performs under different flight conditions (e.g. low α, positive instead of negative β, etc) can inform the user of flight patterns that minimize the error. Maps of the error using the calibration data are shown in Fig. 4.9, with IAS errors (computed directly from dynamic pressure, not the neural network) in a), and α and β in b) and c). The calibration data are grouped into bins 0.3° in the α direction by 0.75° in the β direction. The data points that fall into each bin are then used to calculate RMSE_α, RMSE_β, and $\mathrm{RMSE}_{\mathrm{IAS}}$. Focusing on just the network performance, it is clear that lower errors occur in the general $\alpha = 2° : 4°$ and $\beta = 0° : 2°$ area. This is the same vicinity as the densest location of calibration data points. Overall, the network performs better with respect to α in the $\alpha = 2° : 5°$ and $\beta = -2° : 8°$ region, while $\alpha = 1° : 5°$ and $\beta = -1° : 4°$ is better with respect to β performance.



(a)



(b)

Figure 4.8: Calibration data regression.

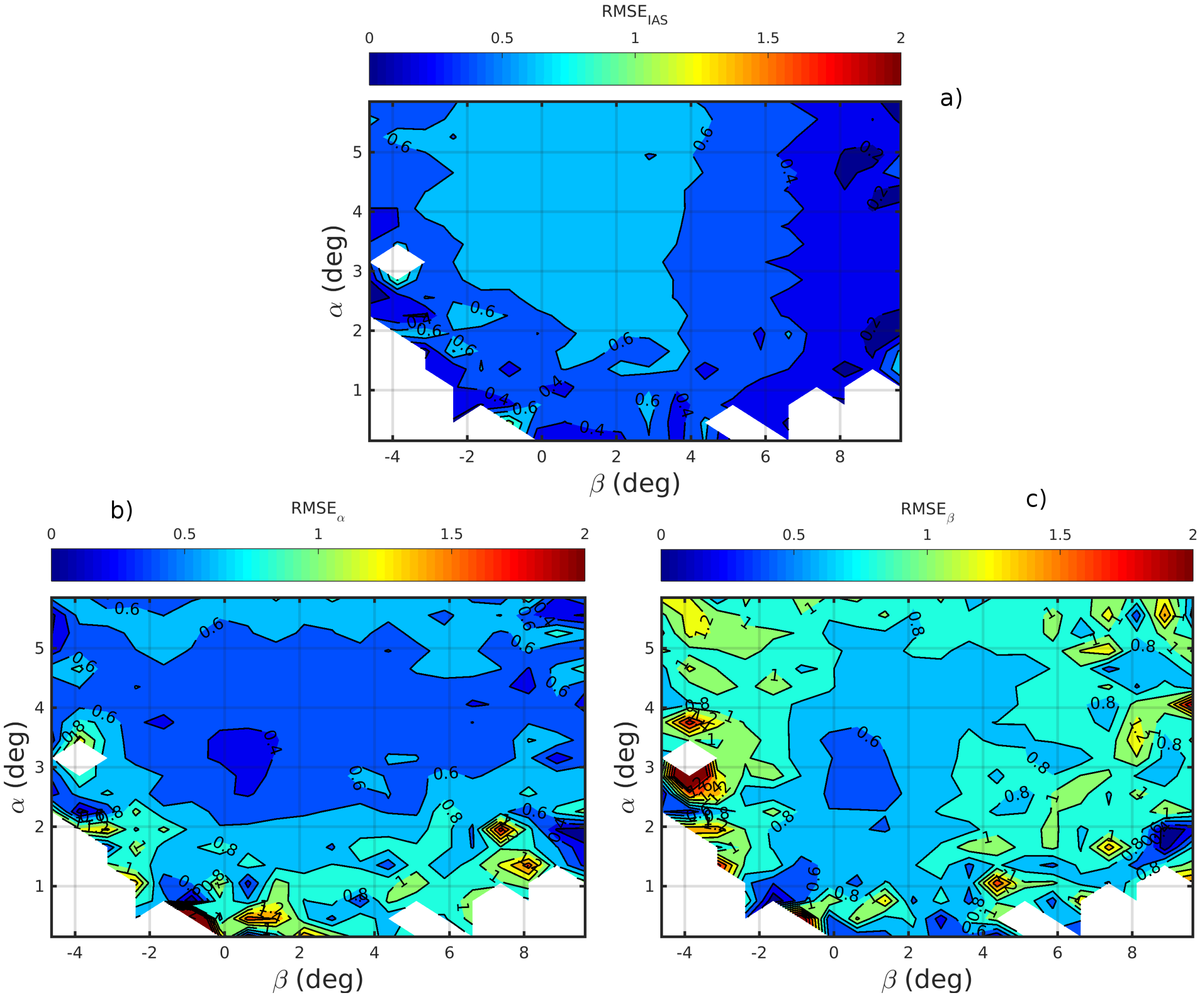


Figure 4.9: Calibration data RMSE values.

4.4 Results

Comparing the map of RMSE values produced by the network for the validation data, shown in Fig. 4.10, to the RMSE map for calibration data, Fig. 4.9, shows validation results are on par with the calibration results. The same higher accuracy regions for the calibration data hold true for the validation data. While this result is expected, it still serves as a useful check. However, what was unexpected was the noticeably lower RMSE values for IAS with the validation data. This discrepancy will be explained later in the current section.

In addition to the 2D error maps, it is helpful to see how the validation errors are affected by the flight parameters. Errors in IAS, α, and β are plotted against the true IAS in Fig. 4.11. The

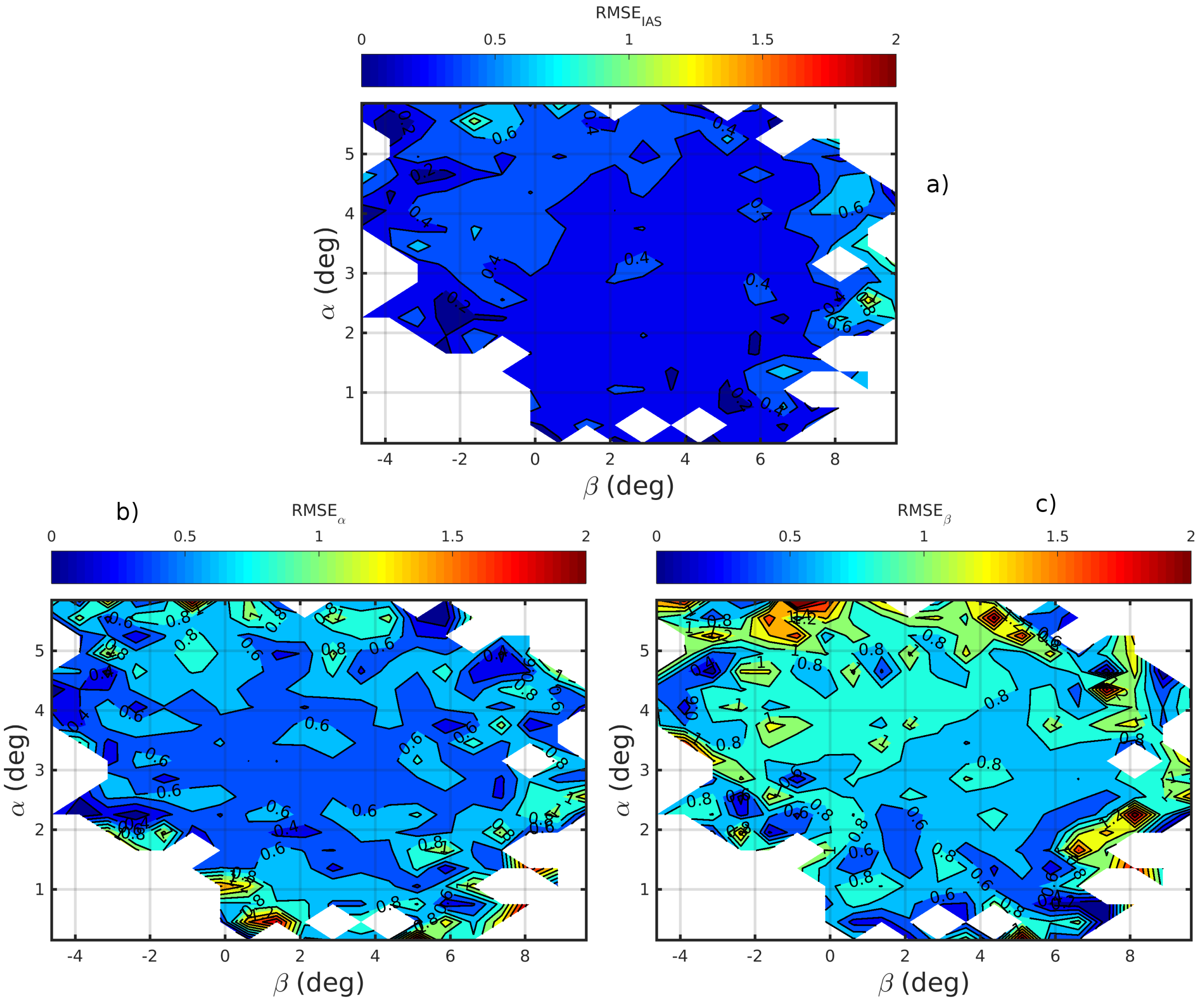


Figure 4.10: Validation data RMSE values.

errors in IAS and β become more positive as the true airspeed increases, but errors in α follow the opposite trend. Assuming the linear fit passes through the center of the errors (which it appears to do a good job of), this fit can serve as a proxy for the mean error. Based on this, β is about twice as sensitive to changes in IAS compared to α. The magnitude of errors in IAS is expected to decrease as the true airspeed increases, yet this is not the case. As such, it is hard to draw meaningful conclusions related to IAS.

The effect of the true α on the errors is shown in Fig. 4.12. Clearly, the errors in α are influenced by the true angle of attack. This tendency was also apparent in the 2D calibration RMSE maps. Interestingly, the true α has virtually no effect at all on the errors in β. The opposite

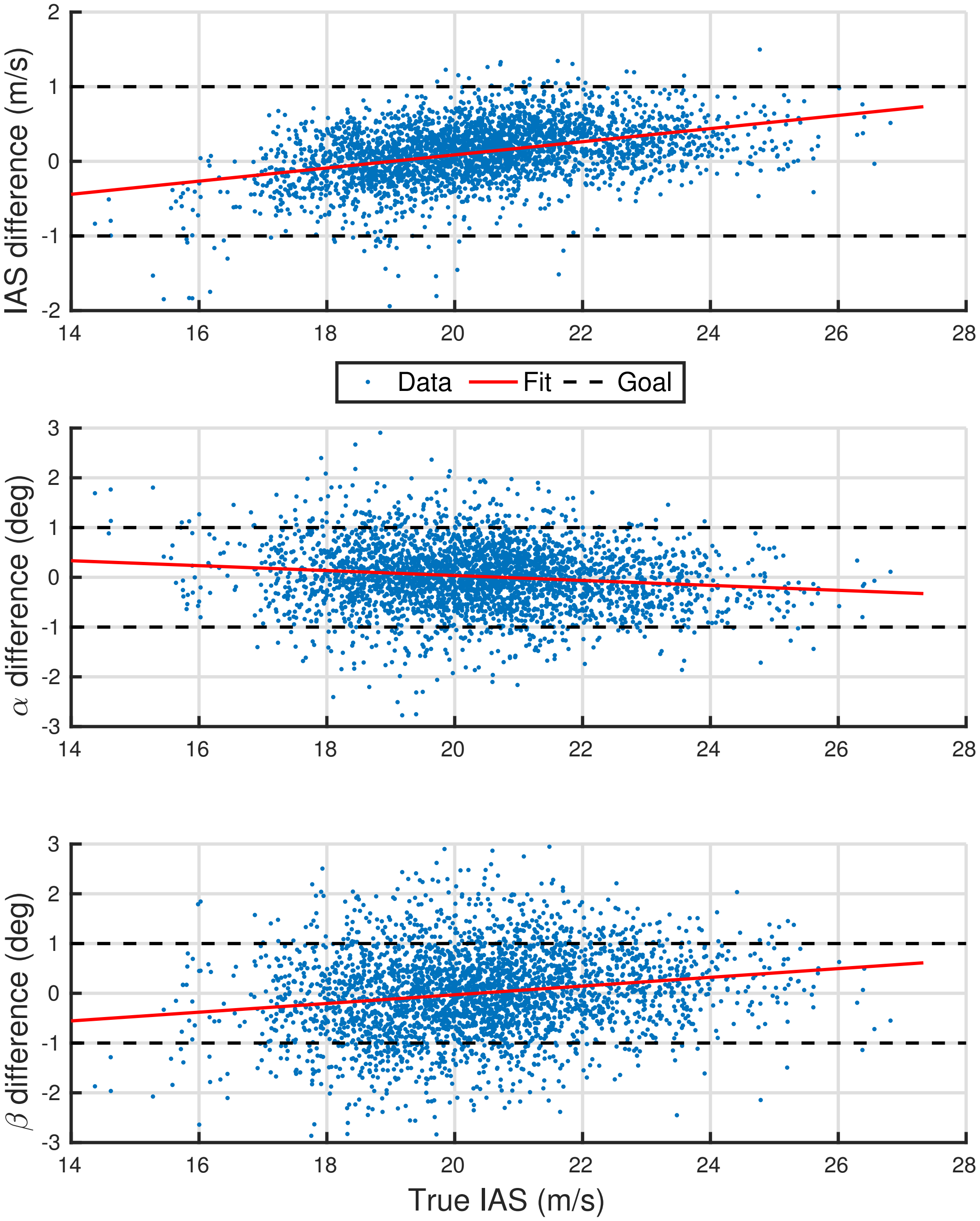


Figure 4.11: IAS effects on validation errors.

Table 4.1: FADS error sensitivity

	IAS (m/s)	α (deg)	β (deg)	ρ (kg/m^3)	$\mathrm{P_s}$ (Pa)
ε_α (deg)	-0.0495	-0.1962	0.0364	-11.445	-1.56×10^{-4}
ε_β (deg)	0.0876	0.0067	-0.0697	11.531	8.485×10^{-5}
$\varepsilon_{\mathrm{IAS}}$ (m/s)	0.0881	0.0131	-0.0573	19.297	7.783×10^{-5}

is not true; the true β does influence the errors in α, as seen in Fig. 4.13. The true β has an impact on the β errors that is twice as strong as its impact on the α errors.

In general, the true ρ and $\mathrm{P_s}$ have a minimal effect on the errors. This is displayed in Figs. B.12 and B.13 in Appendix B. The slope of the fits presented represents a first order look at the sensitivity of the errors to the true flight parameters. The sensitivities are displayed in Table 4.1. For example, the sensitivity of ε_α to changes in the true α is -0.1962 deg/deg, which is 30 times more sensitive than ε_β to true α at 0.0067 deg/deg.

The individual errors from the full validation data set are plotted in Fig. 4.14. It is immediately apparent that the error in IAS is converging to a constant bias as time passes. This explains the discrepancy discussed earlier. The validation data was collected at the start of flight 1, prior to full convergence. The second flight exhibits the same trend of the FADS measuring a lower IAS than the MHP at the start of the flight, before ending with a higher measurement. The IAS error in both flights is shown in Fig. 4.15. Zeroing the sensors post-flight instead of pre-flight may address this issue. It is unknown whether the error lies in the measurements from the FADS or the MHP, and more data is required to determine the cause of this bias in the IAS difference between the two systems.

While there is a clear bias in IAS, 97.9% of the errors in IAS are within 1 m/s. Both α and β appear to be be unbiased, though it is possible to make out some oscillations in the β error early on in the flight. In the end, 93.8% of α errors and 87.3% of the β errors are within 1° of the MHP. Viewing the histograms of the errors, Fig. 4.16, also shows the bias in IAS, but proves that overall, α and β are indeed unbiased. Unsurprisingly, the variance in the β errors is larger than the variance in the α errors.

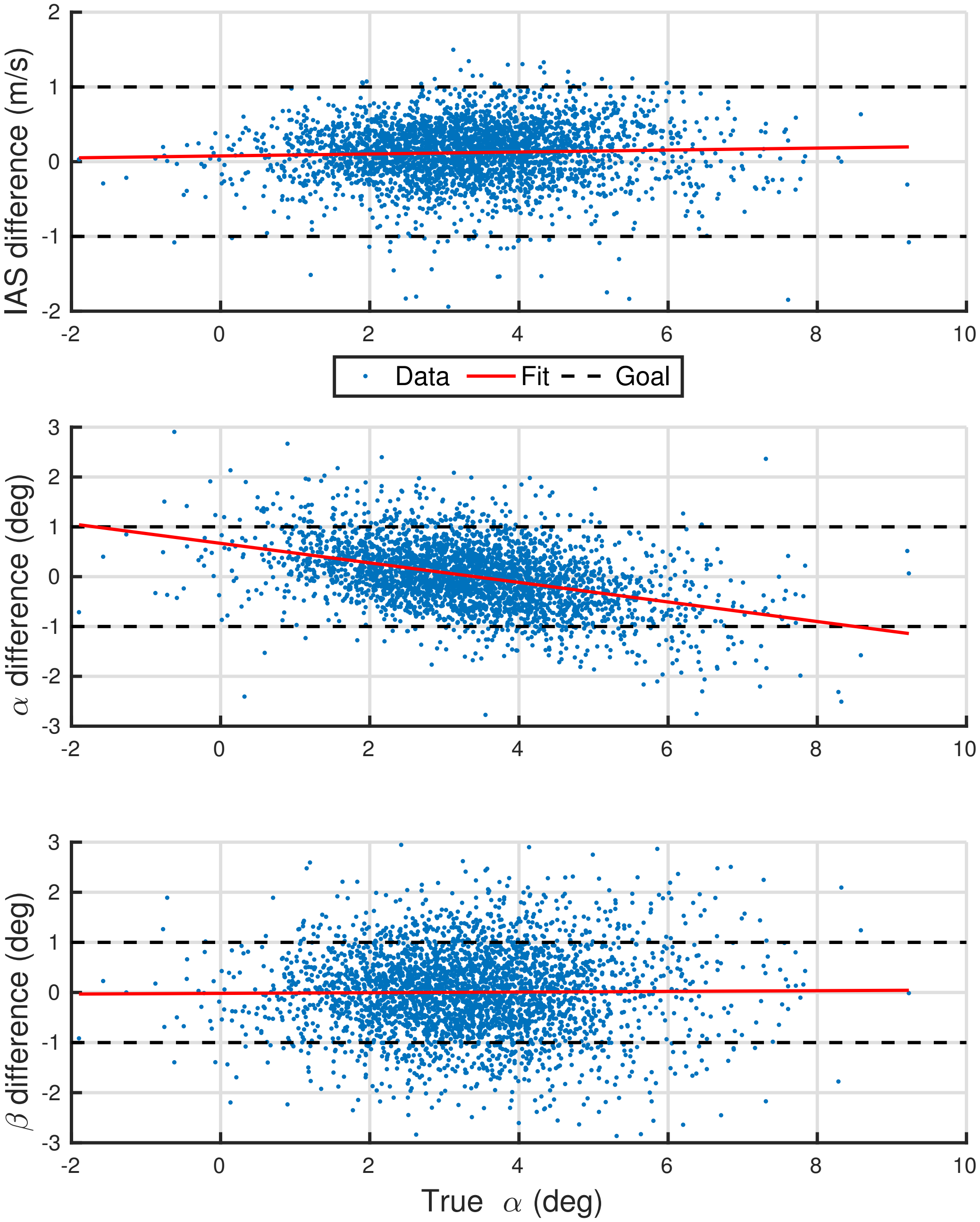


Figure 4.12: α effects on validation errors.

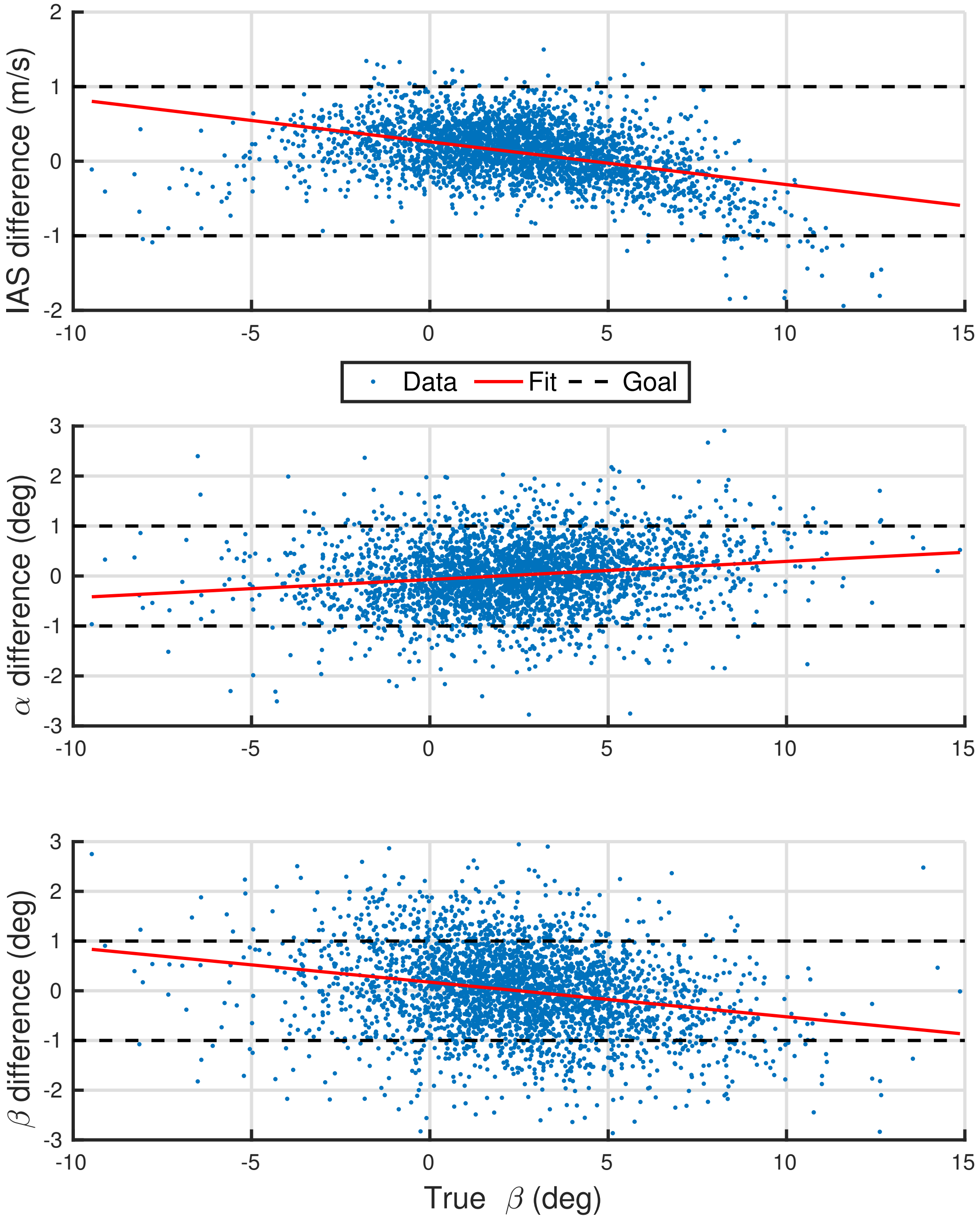


Figure 4.13: β effects on validation errors.

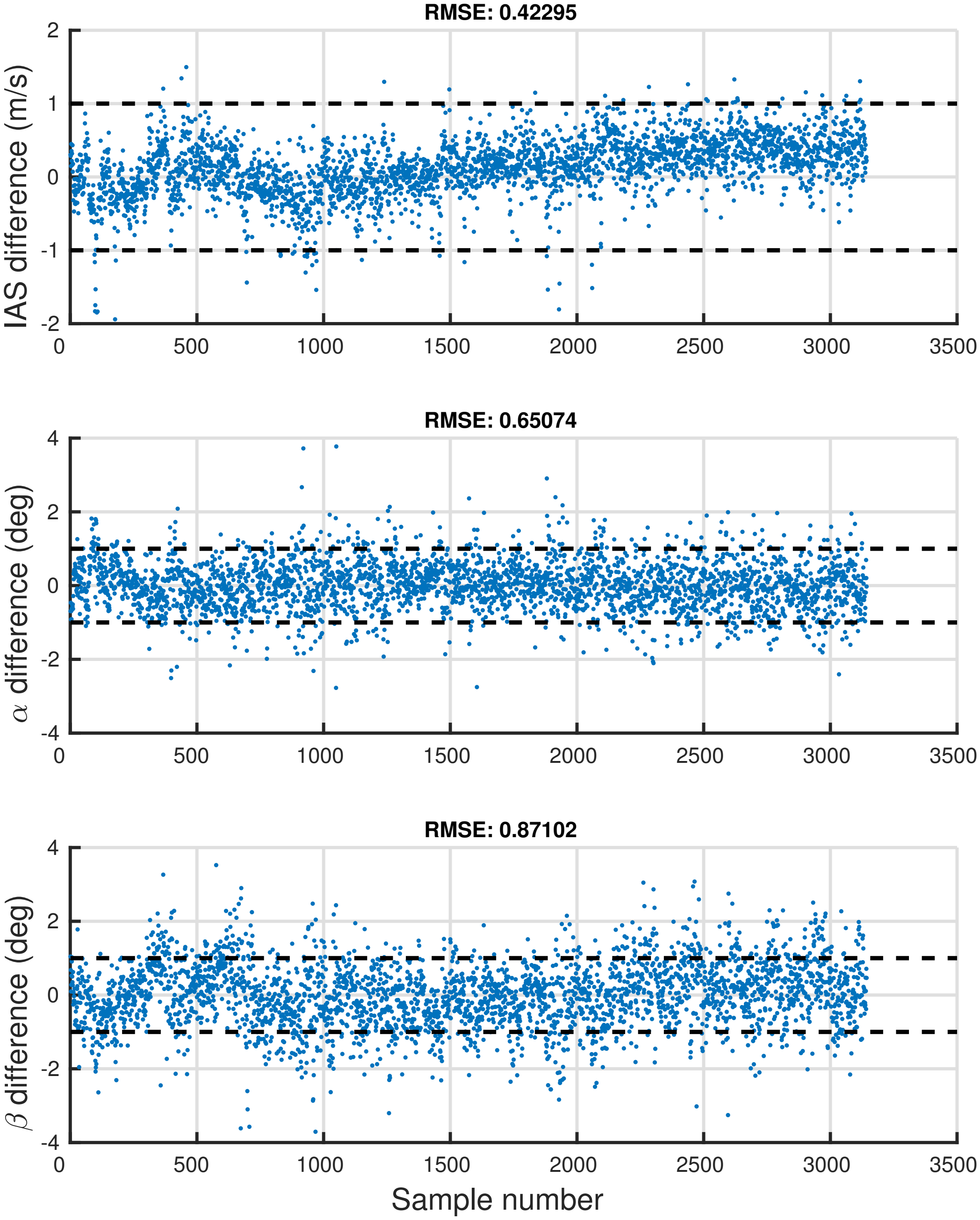


Figure 4.14: Results from full validation portion of flight.

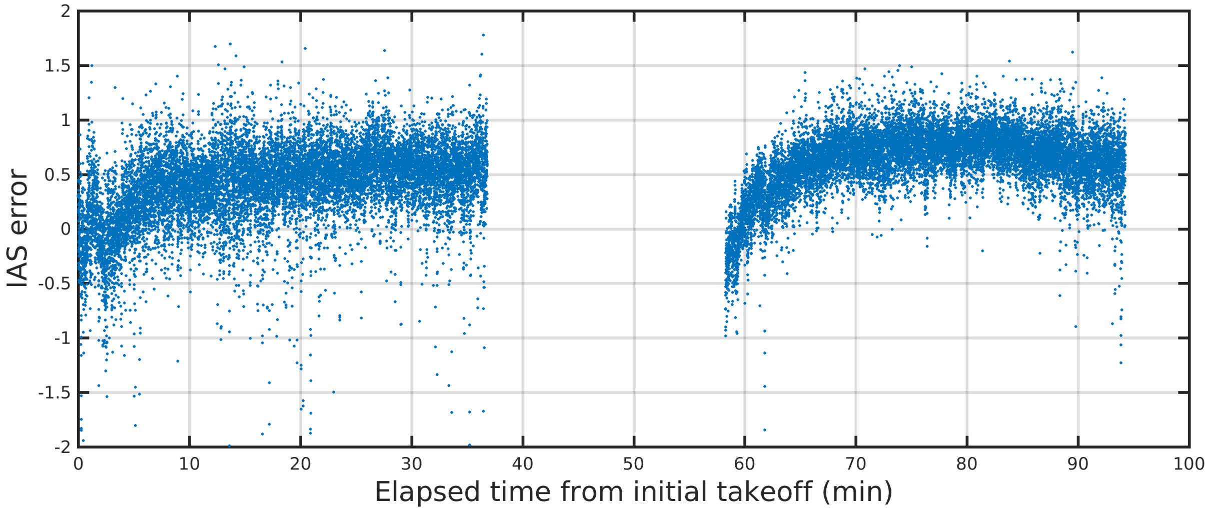


Figure 4.15: IAS errors from both flights.

Results presented so far have focused on the full validation data set. As mentioned previously, the validation portion of the flight is composed of nine distinct patterns. Looking at the patterns individually will inform the user as to what patterns are best to fly for the system as calibrated. The best pattern overall was the first box pattern flown, displayed in Fig. 4.17. This was a box pattern with left hand turns, a nearly constant altitude, lasted 55 seconds, and contains 345 measurements. It is immediately apparent that the FADS (orange) tracks the MHP values (blue) very well for

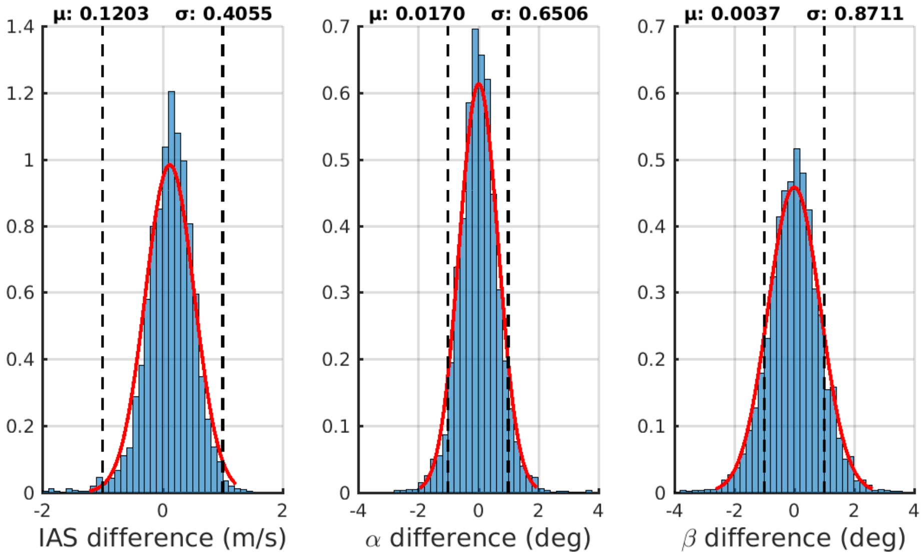


Figure 4.16: Histogram of errors in the full validation portion of flight testing.

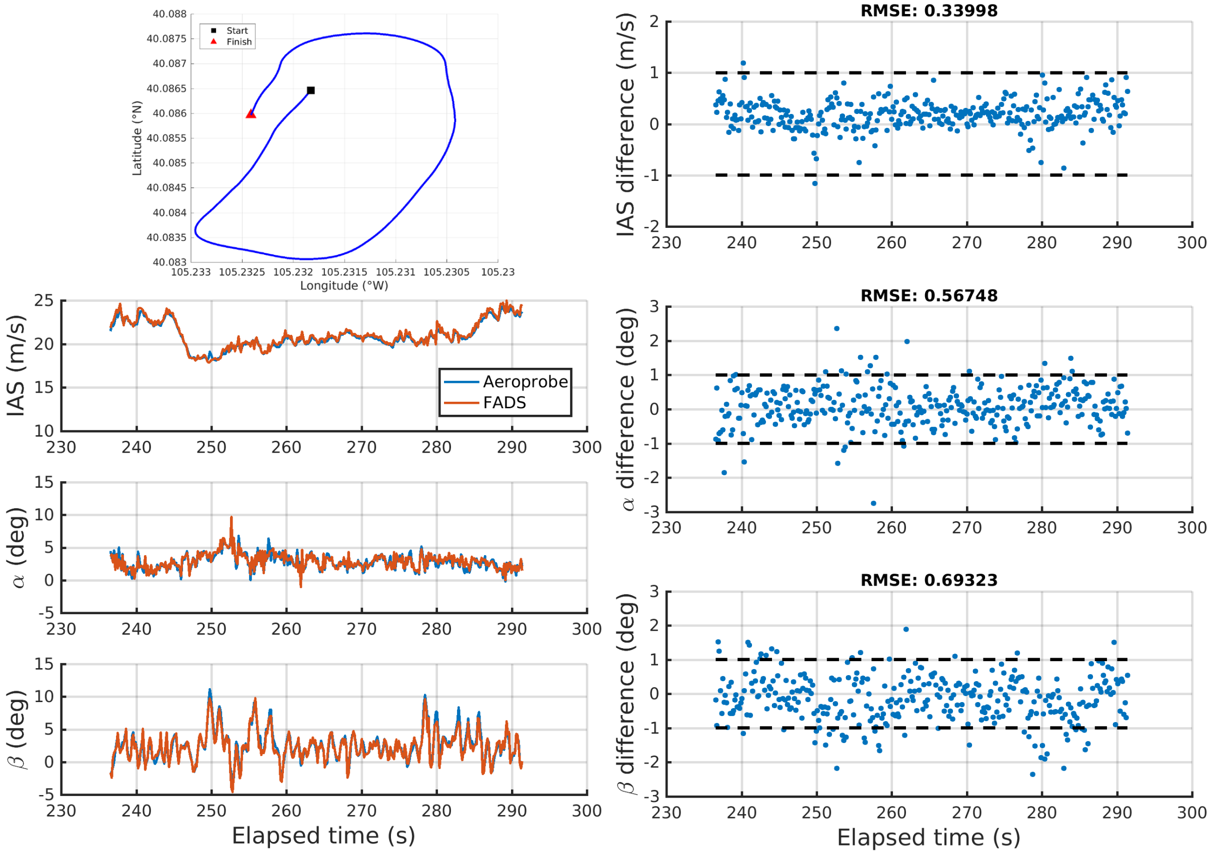


Figure 4.17: Box 1 validation pattern.

IAS, α, and β. As has been the case so far, the FADS does a better job estimating α than β, as evidenced by the RMSE values of 0.567° and 0.693°, respectively. The measurements in β are less noisy compared to α, so it is easier to see how well the FADS β estimate tracks the truth. The largest errors in β occur with the peaks at high β values. It is already known that the FADS has larger errors for high β values.

The worst pattern was the left turning ellipse, Fig. 4.18, with one end at a higher altitude than the start/finish end. This pattern lasted 45 seconds and 284 measurements were taken during this time. The FADS reports a lower IAS compared to the MHP for the majority of the pattern, and misses two of the peaks the MHP records. The FADS also misses several of the peaks in α. It is possible this is because the FADS exhibits a slower response in α than the MHP does. The variations in β are slower, and the FADS does a great job following the variations.

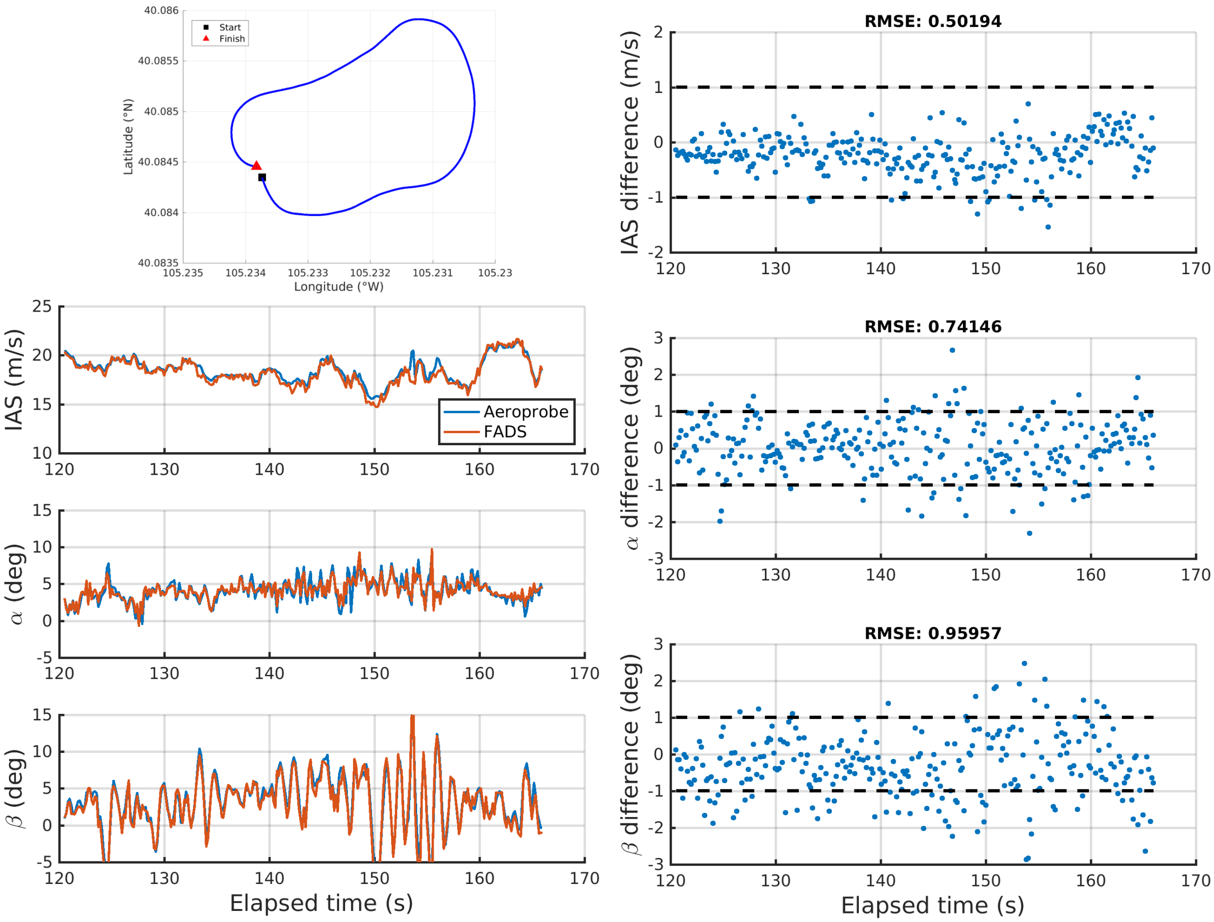


Figure 4.18: Ellipse 1 validation pattern.

The first spiral, which includes both the ascending and descending portions, is shown in Fig. 4.19. This pattern lasted 42 seconds and 260 measurements were recorded. There are two parts of this pattern where the FADS is reporting a 2 m/s slower IAS compared to the MHP. The α and β performance of the FADS during this pattern is better than the overall validation set, though the α estimate tends to be higher than the MHP measurements.

Another validation pattern that did well was the first straight leg, shown in Fig. 4.20. The altitude was constant, and the airspeed was nearly constant. This pattern was only 19 seconds long, and there were 119 measurements. There is better agreement in IAS with this pattern, likely due to the minimal variations in airspeed. The performance of the FADS for α and β is also above average. This above average performance is likely due to the slower variations in α and β during

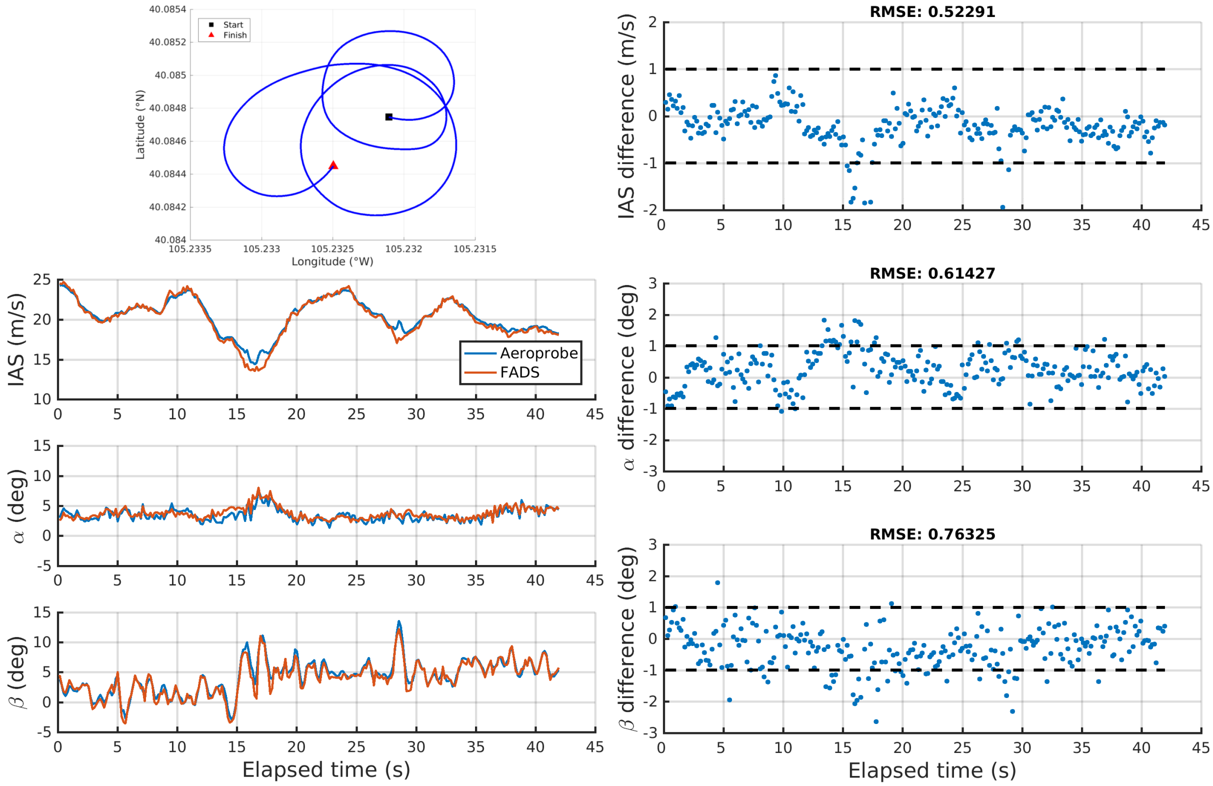


Figure 4.19: Spiral 1 validation pattern.

this pattern.

The RMSE values from each pattern are presented in Table 4.2, along with the overall RMSE values for the full validation data set. The box 1 pattern exhibits the lowest error in IAS and β, while leg 1 has the lowest α error (though box 1 has the second lowest α error). Spiral 1 had the worst agreement in IAS, ellipse 1 had the largest error in α, and spiral 2 was the worst for β. Overall, the FADS performed the best during the box 1 and leg 1 patterns. Therefore, it is recommended that, based on the current calibration of the system, boxes and straight legs are flown in order to minimize the error.

Section 1.3 discussed potential errors when transforming the relative wind to the inertial wind. Applying this analysis to the errors recorded in flight gives an estimate of the inertial wind errors due to the errors in the FADS. Accounting only for errors in the FADS, Eq. 1.6 simplifies to:

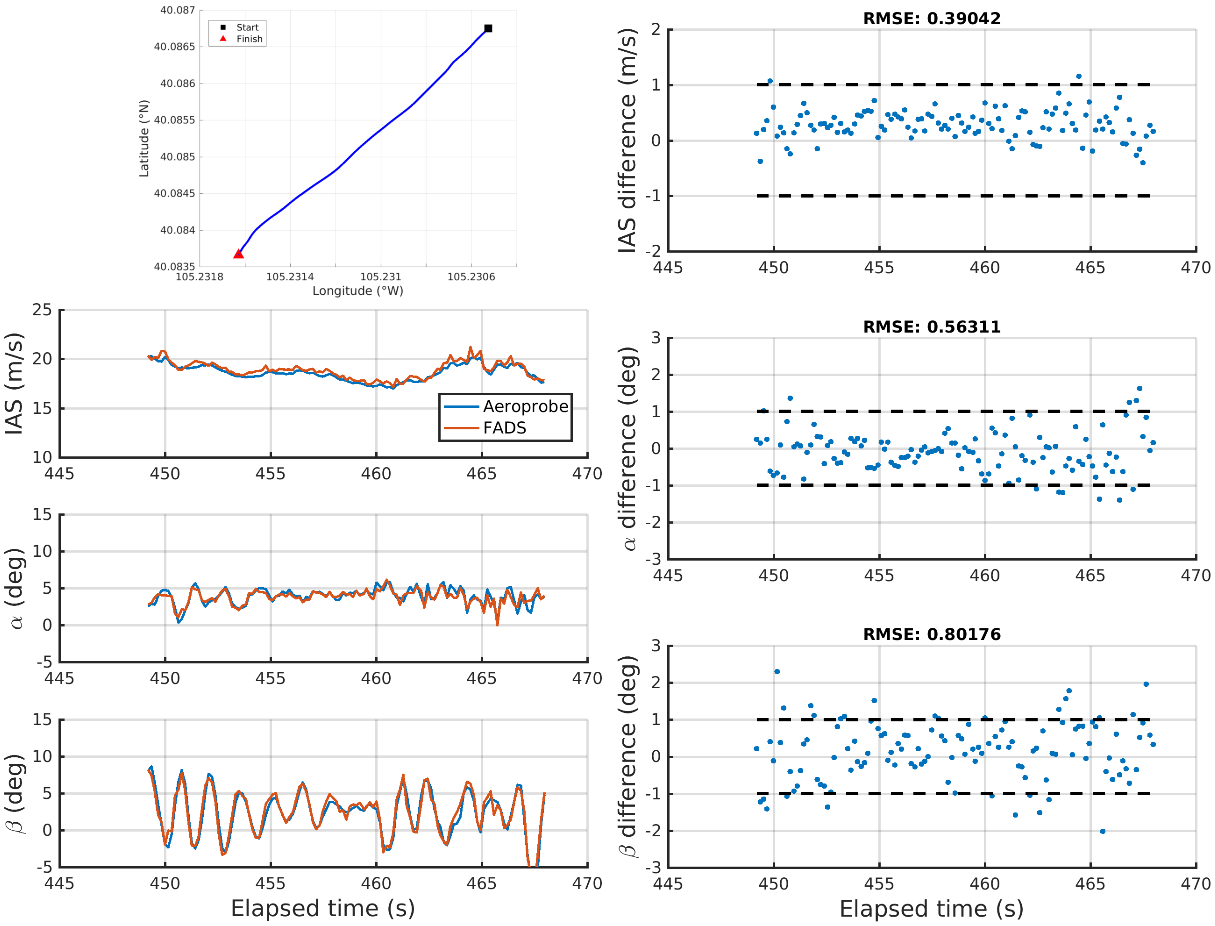


Figure 4.20: Leg 1 validation pattern.

Table 4.2: RMSE of validation patterns

	IAS (m/s)	α (deg)	β (deg)
Spiral 1	0.5229	0.6143	0.7633
Spiral 2	0.3678	0.6281	0.9663
Ellipse 1	0.5019	0.7415	0.9596
Box 1	0.3400	0.5675	0.6932
Box 2	0.4314	0.6775	0.9270
Box 3	0.4359	0.6323	0.8637
Box 4	0.4720	0.6753	0.9172
Leg 1	0.3904	0.5631	0.8018
Leg 2	0.4456	0.6965	0.7714
Full validation	0.4230	0.6507	0.8710

$$\sigma_w \approx \sqrt{\left(\frac{\partial W}{\partial V_a}\sigma_{V_a}\right)^2 + \left(\frac{\partial W}{\partial \alpha}\sigma_\alpha\right)^2 + \left(\frac{\partial W}{\partial \beta}\sigma_\beta\right)^2} \tag{4.7}$$

From Fig. 1.5, the following sensitivities are computed (assuming a flight speed of 20 m/s):

$$\frac{\partial W}{\partial V_a} = [-1, 0, \approx 0] \tag{4.8}$$

$$\frac{\partial W}{\partial \alpha} = [\approx 0, 0, -0.35] \tag{4.9}$$

$$\frac{\partial W}{\partial \beta} = [0, 0.35, 0] \tag{4.10}$$

with the sensitivities shown in the x-, y-, and z-directions for the inertial wind. Combining these sensitivities with the standard deviations in the FADS errors shown in Fig. 4.16, the standard deviations in the inertial wind are: $\sigma_{W_x} = 0.41$ m/s, $\sigma_{W_y} = 0.30$ m/s, and $\sigma_{W_z} = 0.23$ m/s. The standard deviation in the inertial wind magnitude, σ_W, is 0.56 m/s. Therefore, the 3σ error bound[9] in the inertial wind magnitude from the flush airdata system is approximately ±1.7 m/s.

4.5 System Limitations

The results show the flush airdata system performs comparably to the reference multi-hole probe, yet the FADS, as implemented, still has some limitations. The largest limitation currently is access to a high-quality reference instrument for truth data that can be used with sUAS during calibration. The MHP has its own measurements errors inherent to the system. Additionally, flow effects from the Skywalker itself will affect the MHP measurements. While accounting for these errors will lead to improvements in the accuracy of the FADS, modeling these errors is beyond the scope of this dissertation. Plus, some of the error corrections for the MHP can be made post-flight, which only requires the network to be retrained, as opposed to needing new flight data.

Several other limitations of the FADS, as implemented, exist. The MHP records data at 50 Hz, while the FADS only records at 6.2 Hz. The MHP is able to measure features on a shorter time

---

[9] Errors in FADS are normally distributed, so the 3σ error bound represents the interval in which 99.7% of the errors lie

scale than the FADS in able to respond to, leading to errors. While neural networks have proved their capability, they offer no robustness to sensor failure as sensor inputs cannot be excluded from the network. One possibility is training new networks without inputs from certain sensors, but this becomes infeasible quickly. The number of new networks required to account for $\mathrm{n_{faulty}}$ sensors out of $\mathrm{n_{total}}$ sensors is:

$$\mathrm{net_{required}} = \frac{\mathrm{n_{total}}!}{(\mathrm{n_{total}} - \mathrm{n_{faulty}})!\mathrm{n_{faulty}}!} \tag{4.11}$$

Accounting for a single sensor failure requires 12 new networks, while two sensor failures requires 66 new networks. This has been mitigated by grouping sensors together, and ignoring entire groups if one sensor is faulty [63], preprocessing the inputs with an autoassociative network [9], or taking a least squares approach [79].

One final limitation of note is related to the airframe of the Skywalker UAS. The aircraft is made of foam, and the spar in the wings does not connect to the winglets. This led to noticeable flexing at the winglet joint compared to the rest of the wing. The flexing likely contributes to the larger errors in β compared to errors in α.

4.6 Conclusion

Two flights with the X-8 Skywalker and the flush airdata system were made on March 22, 2017. Most of the flight data was used for calibration, but a portion was reserved for validation of the FADS. The FADS was calibrated against a 5-hole probe from Aeroprobe Corporation. Airspeed from the FADS was calculated using the dynamic pressure measured between static and stagnation ports installed on the fuselage. The angle of attack and sideslip were estimated with a neural network using pressure measurements from 12 distributed sensors.

The best training method and hidden layer size was determined by comparing the RMSE values. It was determined that, with the collected flight data, the Bayesian regularization training method with nine hidden units performed the best. It was also concluded that the flight dataset is not large enough to warrant the faster, though less accurate, conjugate gradient training methods. Maps in α-β space were created showing where the FADS was most accurate. These maps can be used to help inform the user of flight patterns that produce the best results, which were shown to be boxes and straight legs. 97.9% of the IAS errors were within 1 m/s of the reference MHP, while 93.8% of the α errors and 87.3% of the β errors were within 1° of the MHP. Overall, the errors in α and β were unbiased, while a small bias was present in IAS. Further improvements to the FADS can be achieved by addressing the current limitations. Even with the system limitations, calibrating a FADS in flight with an sUAS is a viable option for wind sensing. Performance was comparable to the reference MHP, yet with significantly reduced hardware costs.

Chapter 5

Summary and Conclusion

5.1 Summary

Being able to perform accurate wind sensing is of great importance to many disciplines. Small unmanned aircraft systems offer numerous advantages over many conventional methods. sUAS are able to fly where it would be unwise, unsafe, or otherwise inappropriate to use manned aircraft and are able to gather targeted *in situ* measurements, which allows for data gathering of other parameters of interest. One of the more accurate options for relative wind measurement is the multi-hole probe. However, these probes are both expensive and may be exposed due to the need for access to the freestream flow. This exposure can increase the risk of damage to the probe during routine operations. Additionally, these multi-hole probes can cost several times that of the UAS. Flush airdata systems are an attractive alternative to multi-hole probes as they remove external components from the system. This mitigates the risk of damage to the airdata system. This is one of the primary motivators for implementing flush airdata systems on UAS.

Prior work used wind tunnel testing of the Eagle Owl UAS to determine the efficacy of the distributed flush airdata system approach to wind sensing. For simplicity, these tests were only concerned with measuring the angle of attack and sideslip. Cost functions used results from computational fluid dynamics simulations to determine port locations. Using the wind tunnel data, nonlinear least squares was compared to multi-layer feedforward neural networks. The neural networks produced more accurate results, with 100% of the angle of attack errors and 99.97% of the angle of sideslip errors within 1° of the truth.

Lessons learned from the Eagle Owl testing were used to adapt the flush airdata system for use in flight on the X-8 Skywalker UAS. Simulations with computational fluid dynamics software were again used to determine the port locations. However, no cost functions were used. Instead, the gradients of the pressure coefficient was used to calculate a “total sensitivity”, with which points on the airframe were ordered. Eight locations close to the leading edge of the wings were chosen due to their high sensitivity to changes in the angle of attack; eight locations on the winglets were chosen for their sensitivity to sideslip. Simulation results from a specified trim condition were utilized for the determination of static and stagnation ports on the fuselage. Circuit boards with the sealed pressure sensors were installed in both wings, with pneumatic tubing leading from the port location to the sensors.

Two flights of the X-8 Skywalker were completed in March of 2017. The majority of the flight time was calibration, but a portion was reserved for validation of the flush airdata system. A multi-hole probe served as the reference instrument against which the flush airdata system was calibrated and validated. The dynamic pressure from the Skywalker’s static and stagnation ports was used to calculate the airspeed. Multi-layer feedforward neural networks were again employed to estimate the angle of attack and sideslip. Multiple neural network training methods and hidden layer sizes were compared in an effort to determine the most accurate method and size. Focusing on the entire validation portion of the flight, the flush airdata system had 97.9% of the airspeed errors within 1 m/s of the reference multi-hole probe, while 93.8% of the angle of attack errors and 87.3% of the sideslip errors were within 1°. Root-mean-square errors of the flush airdata system were 0.42 m/s, 0.65°, and 0.87°, respectively. While there was an overall bias of 0.12 m/s in airspeed, the estimates of the angle of attack and sideslip were unbiased.

5.2 Future Work

It was shown that the outlined approach to implementing a distributed flush airdata system was successful. It was previously discussed that some of the errors in the sideslip estimates likely stem from the flexing of the wings. This points to the importance of understanding the relationship between the airframe geometry and construction to potential errors in the flush airdata system. As mentioned previously, the reference multi-hole probe was accepted as the truth, even though there are numerous sources of error in those measurements. In particular, errors in the probe measurements due to local flow effects from the airframe should be investigated.

Multi-layer feedforward networks are only one subset of neural networks. Other types of networks, such as radial basis networks [51] or cascade-correlation networks [14], are worth comparing to the feedforward networks used in this dissertation. Even sticking with feedforward networks, improvements can be made. The Eagle Owl testing used a basic ensemble of feedforward networks, while the Skywalker testing only used the single best network from a large group. More advanced ensemble methods exist that are at least as good as the best estimator in the group [52]. While a single hidden layer is all that is required for the network to perform as a universal approximator [22], using multiple hidden layers ("deep learning") may lead to an increase in accuracy. Even without changing the neural network approach, an increase in accuracy might be achieved by modifying the inputs to the networks, such as by including control surface deflections or even the throttle input. Further, while feedforward networks are a subset of neural networks, neural networks are only a small subset of machine learning in general. Many different algorithms exist, but one of the more promising methods is Gaussian processes [55]. The big advantage of Gaussian processes is that they are able to produce a covariance to go along with the estimate, something neural networks are not designed to do.

In addition to focusing on increasing the accuracy of the system, future work should improve upon the hardware installation method. While the procedure presented certainly worked, it was cumbersome and would be significantly more difficult on a non-foam airframe. A promising technology that has seen lots of investment recently is 3D printing. Time can be saved by 3D printing sections of the wings with the pressure ports already installed. Designing the sections so that the pressure sensors can be directly integrated into the 3D printed parts would also vastly decrease the complexity of the installation.

5.3 Conclusion

A new approach to designing a flush airdata system for small UAS was presented. Computational fluid dynamics was used to determine the locations of the pressure ports. Early work calibrated the system using a wind tunnel, while later work calibrated the system in flight using a multi-hole probe as the reference instrument. Airspeed measurements were made from measuring the dynamic pressure, while angle of attack and sideslip estimates were computed using feedforward neural networks.

A goal of being accurate to within 1 m/s and 1° was set. While not all of the errors were within the set bounds, the majority were, with 97.9%, 93.8%, and 87.3% of the airspeed, angle of attack, and angle of sideslip estimates meeting the goal. The airspeed measurements from the flush airdata system were biased compared to the multi-hole probe, and while some hypotheses for this bias were offered, not enough data is available to conclusively prove them. On the other hand, errors in the angle of attack and sideslip were unbiased when viewing the entire validation flight. Viewing the errors in α-β space showed that the system was most accurate in the region with the highest density of training points. Therefore, flights should be designed so that the aircraft operates in the higher accuracy area most often; alternatively, if a different higher accuracy area is desired, the system can be recalibrated. While there is room for further improvement, this method for aircraft-relative wind sensing has certainly proven successful.

The distributed flush airdata system will allow more aircraft and more teams to engage in wind sensing missions. A method of determining port locations that includes the entire airframe was developed. This method only needs to be applied once per airframe, which benefits a fleet of aircraft. However, while this approach will work on any arbitrary airframe, not every airframe is well suited for installing a flush airdata system. This is due to the fact that the accuracy of the flush airdata system is highly dependent on the aircraft geometry. As implemented, a multi-hole probe is required for calibration, though a different reference instrument can certainly be substituted. The multi-hole probe increases the cost of the system, but only one is required to calibrate an entire fleet of aircraft, thus reducing the overall cost.

Bibliography

[1] Aeroprobe. Aeroprobe Micro Air Data System User Manual. Number Revision D. Aeroprobe Corporation, 2nd edition, 2016.

[2] John D. Anderson. Introduction to Flight. McGraw Hill, 7th edition, 2012.

[3] Ben B. Balsley, Dale a. Lawrence, Ronald F. Woodman, and David C. Fritts. Fine-Scale Characteristics of Temperature, Wind, and Turbulence in the Lower Atmosphere (0-1,300 m) Over the South Peruvian Coast. Boundary-Layer Meteorology, 147(1):165–178, 2013.

[4] S. Callegari, M. Zagnoni, A. Golfarelli, M. Tartagni, A. Talamelli, P. Proli, and A. Rossetti. Experiments on aircraft flight parameter detection by on-skin sensors. Sensors and Actuators, A: Physical, 130-131:155–165, 2006.

[5] John P. Cary and Earl R. Keener. Flight Evaluation of the X-15 Ball-Nose Flow-Direction Sensor as an Air-Data System. NASA Technical Note D-2923, National Aeronautics and Space Administration, 1965.

[6] C. Charalambous. Conjugate gradient algorithm for efficient training of artificial neural networks. IEE Proceedings G Circuits, Devices and Systems, 139(3):301, 1992.

[7] Brent Cobleigh, Stephen Whitmore, Edward Haering, Jr., Jerry Borrer, and V. Roback. Flush airdata sensing (FADS) system calibration procedures and results for blunt forebodies. In 9th International Space Planes and Hypersonic Systems and Technologies Conference, number November, pages 1–27, 1999.

[8] J L Crassidis and J L Junkins. Optimal Estimation of Dynamic Systems. CRC Press, 2nd edition, 2011.

[9] W J Crowther and P J Lamont. A neural network approach to the calibration of a flush air data system. Technical Report November, 2000.

[10] Jack Elston, Brian Argrow, and Maciej Stachura. Covariance Analysis of Sensors for Wind Field Estimation by Small Unmanned Aircraft. In AIAA Guidance, Navigation, and Control Conference, number January, pages 1–9, 2015.

[11] Jack Elston, Brian Argrow, Maciej Stachura, Doug Weibel, Dale Lawrence, and David Pope. Overview of small fixed-wing unmanned aircraft for meteorological sampling. Journal of Atmospheric and Oceanic Technology, 32:97–115, 2015.

[12] Jack S. Elston, Jason Roadman, Maciej Stachura, Brian Argrow, Adam Houston, and Eric Frew. The tempest unmanned aircraft system for in situ observations of tornadic supercells: Design and vortex2 flight results. Journal of Field Robotics, 28(4):461–483, 2011.

[13] M. Lothon et al. The BLLAST field experiment: Boundary-Layer late afternoon and sunset turbulence. Atmospheric Chemistry and Physics, 14(20):10931–10960, 2014.

[14] Scott E Fahlman and Christian Lebiere. The Cascade-Correlation Learning Architecture. Advances in neural information processing systems, pages 524–532, 1990.

[15] Haiping Fei, Rong Zhu, Zhaoying Zhou, and Jindong Wang. Aircraft flight parameter detection based on a neural network using multiple hot-film flow speed sensors. Smart Materials and Structures, 16(4):1239–1245, 2007.

[16] F Dan Foresee and Martin T Hagan. Gauss-Newton approximation to Bayesian regularization. Proceedings of the 1997 International Joint Conference on Neural Networks, pages 1930–1935, 1997.

[17] Eric Frew. Aircraft Equations of Motion in Wind Axis Coordinate System. (Personal Communication).

[18] William Gracey. Wind-tunnel investigation of a number of total-pressure tubes at high angles of attack - subsonic, transonic, and supersonic speeds. Technical Report 1303, National Advisory Committee for Aeronautics, Langley Aeronautical Laboratory, 1957.

[19] Edward A. Haering. Airdata Measurement and Calibration. NASA Technical Memorandum 104316, NASA, 1995.

[20] M T Hagan, H B Demuth, M H Beale, and Orlando De Jesús. Neural Network Design. 2nd edition, 2014.

[21] Martin T. Hagan and Mohammad B. Menhaj. Training Feedforward Networks with the Marquardt Algorithm. IEEE Transactions on Neural Networks, 5(6):989–993, 1994.

[22] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. Neural Networks, 2(5):359–366, 1989.

[23] Adam Houston, Roger Laurence, Tevis Nichols, Sean Waugh, Brian Argrow, and Conrad Ziegler. Intercomparison of Unmanned Aircraftborne and Mobile Mesonet Atmospheric Sensors. Journal of Atmospheric and Oceanic Technology, 33:1569–1582, 2016.

[24] Adam L Houston, Brian Argrow, Jack Elston, Jamie Lahowetz, and Eric W Frew. The collaborative colorado nebraska unmanned aircraft system experiment. Bulletin of the American Meteorological Society, 93(1):39–54, 2012.

[25] Roger J. Laurence III, Brian M. Argrow, and Eric W. Frew. Wind tunnel results for a distributed flush airdata system. Journal of Atmospheric and Oceanic Technology, 34(7):1519–1528, 2017.

[26] Texas Instruments. TCA9548A Low-Voltage 8-Channel I 2 C Switch With Reset, 2016.

[27] Integrated Remote and In Situ Sensing. Seeing the World in 4D, 2017. Accessed 2017-06-18, http://www.colorado.edu/iriss/.

[28] InvenSense. MPU-6000 and MPU-6050 Product Specification, 2013.

[29] Espen Johansen, Othon Rediniotis, and Greg Jones. The Compressible Calibration of Miniature Multi-Hole Probes. *Transactions of the ASME*, 123(March 2001):128–138, 2001.

[30] John A. Kalogiros and Qing Wang. Aerodynamic effects on wind turbulence measurements with research aircraft. *Journal of Atmospheric and Oceanic Technology*, 19:1567–1576, 2002.

[31] John A. Kalogiros and Qing Wang. Calibration of a radome-differential GPS system on a twin otter research aircraft for turbulence measurements. *Journal of Atmospheric and Oceanic Technology*, 19:159–171, 2002.

[32] G. Kocer, M. Mansour, N. Chokani, R.S. Abhari, and M. Muller. Full-Scale Wind Turbine Near-Wake Measurements Using an Instrumented Uninhabited Aerial Vehicle. *Journal of Solar Energy Engineering*, 133(4):041011, 2011.

[33] T. J. Larson, S. A. Whitmore, L. J. Ehernberger, J. B. Johnson, and P. M. Siemers III. Qualitative Evaluation of a Flush Air Data System at Transonic Speeds and High Angles of Attack. NASA Technical Paper 2716, 1987.

[34] Terry J Larson, Stuart G Flechner, and Paul M Siemers III. Wind Tunnel Investigation of an All Flush Orifice Air Data System for a Large Subsonic Aircraft. NASA Technical Paper 1642, NASA Dryden Flight Research Facility, Edwards, CA, 1980.

[35] Terry J Larson and Paul M Siemers III. Subsonic Tests of an All-Flush-Pressure-Orifice Air Data System. NASA Technical Paper 1871, 1981.

[36] Terry J. Larson and Paul M. Siemers III. Use of Nose Cap and Fuselage Pressure Orifices for Determination of Air Data for Space Shuttle Orbiter Below Supersonic Speeds. Technical Report NASA Technical Paper 1643, 1980.

[37] Roger J. Laurence, Brian M. Argrow, and Eric W. Frew. Development of wind sensing from small uas with distributed pressure sensors. In *8th AIAA Atmospheric and Space Environments Conference*, pages AIAA 2016–4199, Washington, DC, 2016. AIAA Aviation. [http://dx.doi.org/10.2514/6.2016-4199].

[38] Roger J. Laurence, Jack S. Elston, and Brian M. Argrow. A low-cost system for wind field estimation through sensor networks and aircraft design. In *AIAA Infotech @ Aerospace*, pages AIAA 2015–1425, Kissimmee, FL, 2015. AIAA SciTech. [http://dx.doi.org/10.2514/6.2015-1425].

[39] F A P Lie and D Gebre-Egziabher. Synthetic Air Data System. *Journal of Aircraft*, 50(4):1234–1249, 2013.

[40] Chen Lu, Rongbing Li, Jianye Liu, and Tingwan Lei. Calculation Method for Air Data Based on Information from Inertial Navigation System and Wind Field *. In *IEEE Chinese Guidance, Navigation and Control*, pages 1845–1850, 2016.

[41] David J. C. MacKay. Bayesian Interpolation. *Neural Computation*, 4(3):415–447, 1992.

[42] Donald W. Marquardt. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

[43] MathWorks. initnw, 2017. https://www.mathworks.com/help/nnet/ref/initnw.html, accessed 2017-06-19.

[44] Stephanie Mayer and Gautier Hattenberger. A 'no-flow-sensor'wind estimation algorithm for unmanned aerial systems. Journal of Micro Air Vehicles, 4(1):15–29, 2012.

[45] B. Mohammadi and O. Pironneau. Analysis of the K-Epsilon Turbulence Model. Wiley, 1994.

[46] Martin Fodslette Moller. A scaled conjugate gradient algorithm for fast supervised learning. Neural Networks, 6(4):525–533, 1993.

[47] National Oceanic and Atmospheric Administration, National Aeronautics and Space Administration, and United States Air Force. U.S. Standard Atmosphere, 1976. Washington, D.C., 1976.

[48] D. Nguyen and B. Widrow. Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. IJCNN Int. Joint Conf. Neural Networks, 13:C21, 1990.

[49] Tevis Nichols, Brian Argrow, and Derek Kingston. Error Sensitivity Analysis of Small UAS Wind-Sensing Systems. In AIAA Information Systems-AIAA Infotech @ Aerospace, pages AIAA 2017–0647, Grapevine, TX, 2017. AIAA SciTech Forum.

[50] Pavel Paces, Karel Draxler, Vitezslav Hanzal, Tomas Censky, and Ondrej Vasko. A combined angle of attack and angle of sideslip smart probe with twin differential sensor modules and doubled output signal. Proceedings of IEEE Sensors, pages 284–289, 2010.

[51] J Park and I W Sandberg. Universal Approximation Using Radial-Basis-Function Networks. Neural Computation, 3:246–257, 1991.

[52] Michael P. Perrone and Leon N. Cooper. When Networks Disagree: Ensemble Methods for Hybrid Neural Networks. Technical report, 1992.

[53] M. J D Powell. Restart procedures for the conjugate gradient method. Mathematical Programming, 12(1):241–254, 1977.

[54] John F. Quindlen and Jack W. Langelaan. Flush air data sensing for soaring-capable uavs. In 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, pages AIAA 2013–1153, Grapevine, TX, 2013. Aerospace Sciences Meetings. [Available online at http://dx.doi.org/10.2514/6.2013-1153].

[55] C. E. Rasmussen. Gaussian processes for machine learning. 2006.

[56] O. K. Rediniotis and R. Vijayagopal. Miniature multihole pressure probes and their neural-network-based calibration. AIAA Journal, 37(6):666–674, 1999.

[57] Research and Engineering Center for Unmanned Vehicles. RECUV Highlights, 2017. Accessed 2017-06-18, http://www.colorado.edu/recuv/.

[58] Joachim Reuder, Marius O. Jonassen, and Haraldur Ólafsson. The Small Unmanned Meteorological Observer SUMO: Recent developments and applications of a micro-UAS for atmospheric boundary layer research. Acta Geophysica, 60(5):1454–1473, 2012.

[59] Matthew B Rhudy, Yu Gu, Jason N. Gross, and Haiyang Chao. Onboard Wind Velocity Estimation Comparison for Unmanned Aircraft Systems. IEEE Transactions on Aerospace and Electronic Systems, 53(1):55–66, 2017.

[60] Martin Riedmiller and Heinrich Braun. A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In IEEE International Conference on Neural Networks - Conference Proceedings, volume January, pages 586–591, 1993.

[61] Jason Roadman, Jack Elston, Brian Argrow, and Eric Frew. Mission performance of the tempest unmanned aircraft system in supercell storms. Journal of Aircraft, 49(6):1821–1830, 2012.

[62] Thomas J Rohloff. Air Data Sensing from Surface Pressure Measurements Using a Neural Network Method. AIAA Journal, 36(11), 1998.

[63] Thomas J. Rohloff, Stephen A. Whitmore, and Ivan Catton. Fault-Tolerant Neural Network Algorithm for Flush Air Data Sensing. Journal of Aircraft, 36(3):541–549, 1999.

[64] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. Nature, 323(6088):533–536, 1986.

[65] I. Samy, I. Postlethwaite, D. Gu, and J. Green. Neural-Network-Based Flush Air Data Sensing System Demonstrated on a Mini Air Vehicle. Journal of Aircraft, 47(1):18–31, 2010.

[66] Siemens. STAR-CCM+, 2017. http://mdx.plm.automation.siemens.com/star-ccm-plus, accessed 2017-03-31.

[67] P.M. Siemers, H Wolf, and P F Flanagan. Shuttle Entry Air Data System Concepts Applied to Space Shuttle Orbiter Flight Pressure Data to Determine Air Data - STS 1-4. In AIAA 21st Aerospace Sciences Meeting, pages 1–12, Reno, NV, 1983.

[68] TE Sensor Solutions. MS8607-02BA01, 2015. available online at http://www.meas-spec.com/downloads/MS8607-02BA01.pdf.

[69] TE Sensor Solutions. MS5611-01BA03 Barometric Pressure Sensor, with stainless steel cap, 2015.

[70] D. Telionis, Y. Yang, and O. Rediniotis. Recent developments in multi-hole probe (mhp) technology. In 20th International Congress of Mechanical Engineering, Gramado, RS, Brazil, 2009. ABCM.

[71] Michael Tjernstrom and Carl Friehe. Analysis of a Radome Air-Motion System on a Twin-Jet Aircraft for Boundary-Layer Research. Journal of Atmospheric and Oceanic Technology, 8(19-40), 1991.

[72] Aline Van den Kroonenberg, Tim Martin, Marco Buschmann, Jens Bange, and Peter Vorsmann. Measuring the Wind Vector Using the Autonomous Mini Aerial Vehicle M2AV. Journal of Atmospheric and Oceanic Technology, 25:1969–1982, 2008.

[73] Vectornav. VN-200 GPS/INS Specifications, 2017. http://www.vectornav.com/products/vn-200/specifications, accessed 2017-06-26.

[74] VectorNav Technologies. VN-200 User Manual, 2015.

[75] Rajesh Vijayagopal, Mahesh Pathak, and Othon Rediniotis. Miniature multi-hole pressure probes - Their neural network calibration and frequency response enhancement. In 36th AIAA Aerospace Sciences Meeting and Exhibit, pages 1–15, 1998.

[76] T. P. Vogl, J. K. Mangis, A. K. Rigler, W. T. Zink, and D. L. Alkon. Accelerating the convergence of the back-propagation method. Biological Cybernetics, 59(4-5):257–263, 1988.

[77] Christian W. Wenger and William J. Devenport. Seven-hole pressure probe calibration method utilizing look-up error tables. AIAA Journal, 37(6):675–679, 1999.

[78] Stephen Whitmore, Timothy Moes, and Terry Larson. Preliminary results from a subsonic high-angle-of-attack flush airdata sensing (HI-FADS) system - Design, calibration, algorithm development, and flight test evaluation. In 28th Aerospace Sciences Meeting, pages 1–12, 1990.

[79] Stephen Whitmore, Timothy Moes, and Cornelius Leondes. Failure detection and fault management techniques for flush airdata sensing systems. 30th Aerospace Sciences Meeting and Exhibit, 1992.

[80] Stephen A Whitmore. Development of a pneumatic high-angle-of-attack flush airdata sensing (hi-fads) system. NASA Technical Memorandum 104241, NASA Dryden Flight Research Facility, Edwards, CA, 1991.

[81] Stephen A Whitmore, Brent R Cobleigh, and Edward A Haering. Design and Calibration of the X-33 Flush Airdata Sensing (FADS) System. In 36th AIAA Aerospace Sciences Meeting and Exhibit, pages 1–36, Reno, NV, 1997.

[82] N. Wildmann, M. Hofsäß, F. Weimer, A. Joos, and J. Bange. MASC a small Remotely Piloted Aircraft (RPA) for wind energy research. Advances in Science and Research, 11(1):55–61, 2014.

[83] N. Wildmann, S. Ravi, and J. Bange. Towards higher accuracy and better frequency response with standard multi-hole probes in turbulence measurement with remotely piloted aircraft (RPA). Atmospheric Measurement Techniques, 7(4):1027–1041, 2014.

[84] Barry J Wythoff. Backpropagation neural networks: A tutorial. Chemometrics and Intelligent Laboratory System, 18:115–155, 1993.

Appendix A

Definitions

A list of some of the commonly used terms, abbreviations, and acronyms

α: angle of attack; defined as pitching up relative to the oncoming airflow

β: angle of sideslip; defined as yawing to the right relative to the oncoming airflow

ρ: density

aircraft state estimate: includes estimate of aircraft inertial velocity, orientation relative to the Earth, and rates of change in the orientation.

FADS: flush airdata system

IAS: indicated airspeed

inertial wind: wind experienced by an object stationary relative to the Earth

MHP: multi-hole probe; specifically, the Aeroprobe 5-hole probe in the context of results presented in this dissertation

MSE: mean squared error

P_s: static pressure

P_t: stagnation pressure (for incompressible flow, this equals total pressure)

q: dynamic pressure

RECUV: Research and Engineering Center for Unmanned Vehicles; one focus is performing targeted meteorological observations with unmanned aircraft systems

relative wind: wind experienced by the aircraft

RMSE: root mean square error

TAS: true airspeed

UAS: unmanned aircraft system

Appendix B

Flight Results

Images relevant to the Skywalker flights

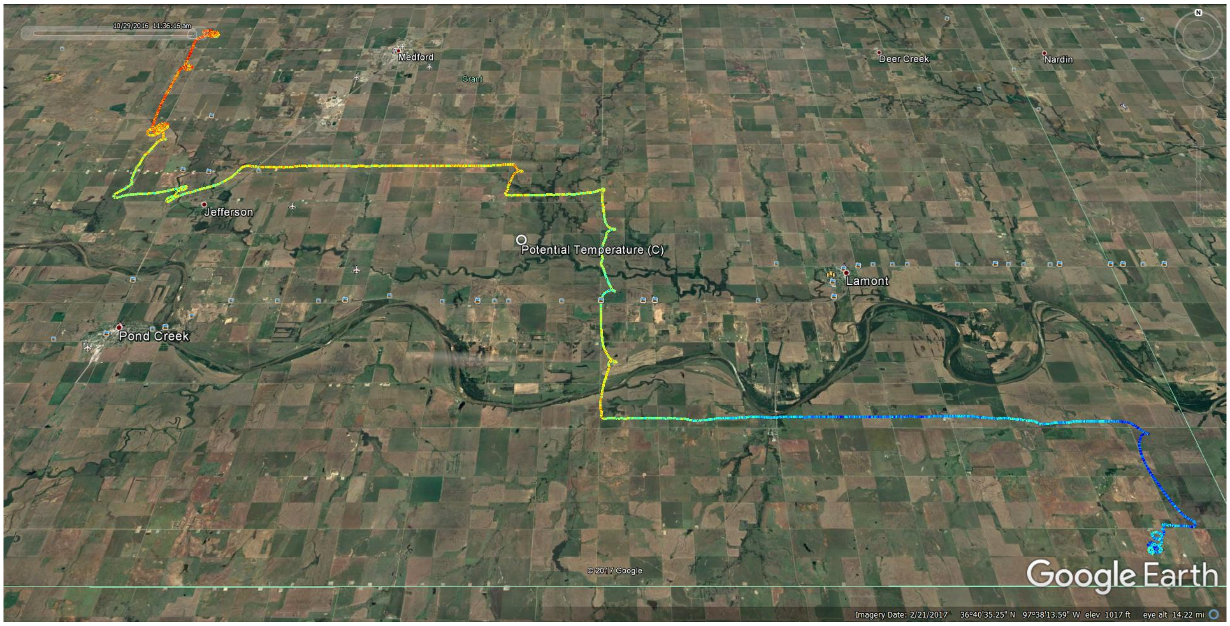


Figure B.1: Flight from October 2016 showing typical ground track.

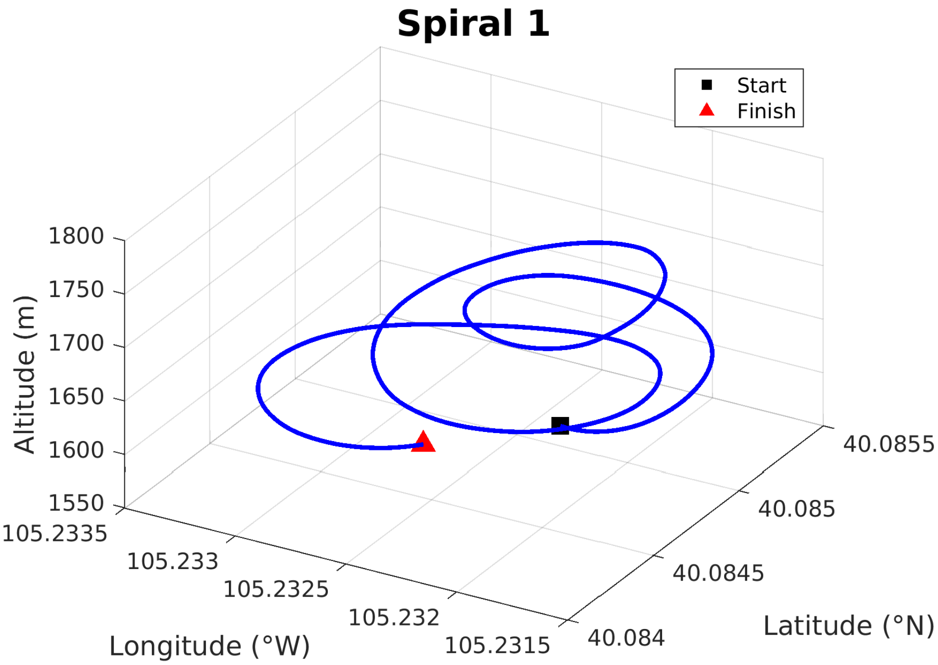


Figure B.2: Spiral 1

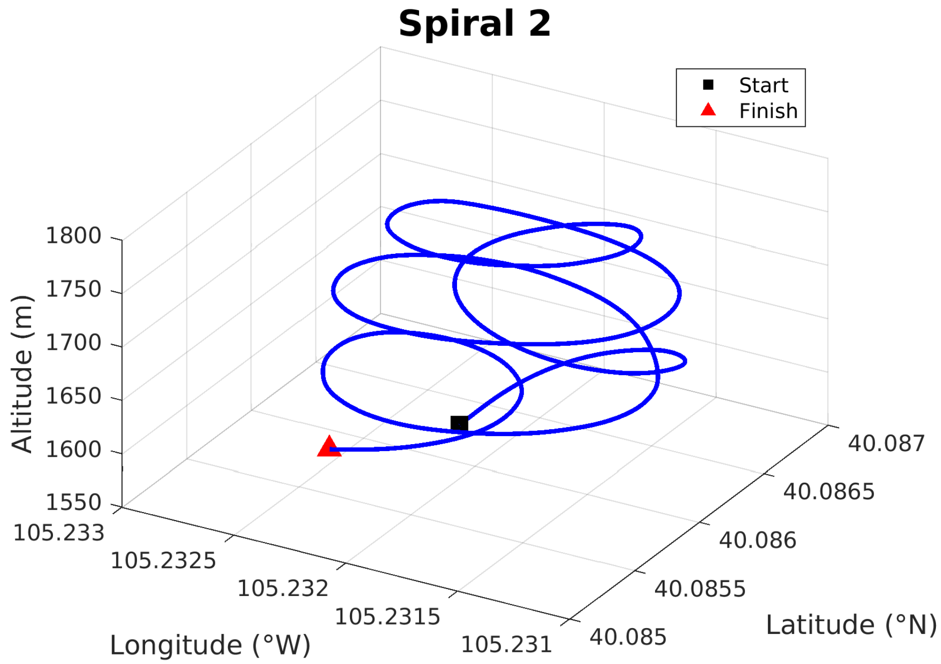


Figure B.3: Spiral 2

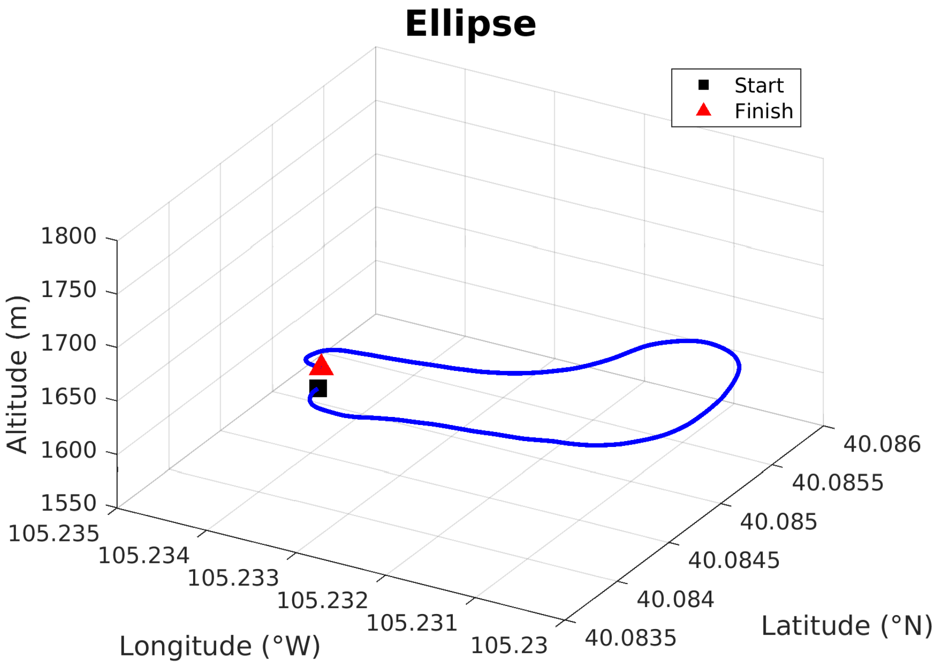


Figure B.4: Ellipse 1

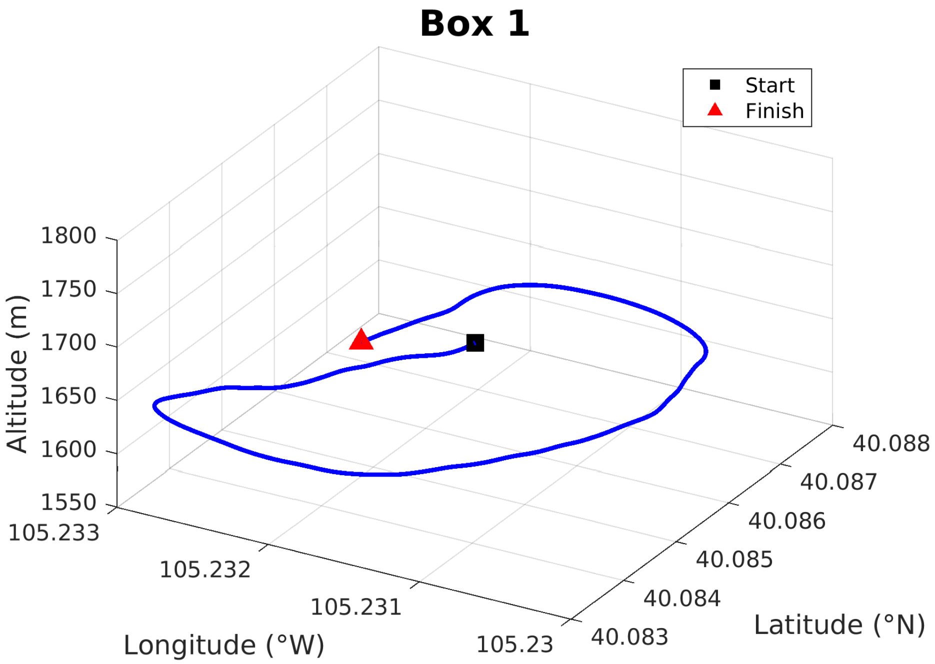


Figure B.5: Box 1

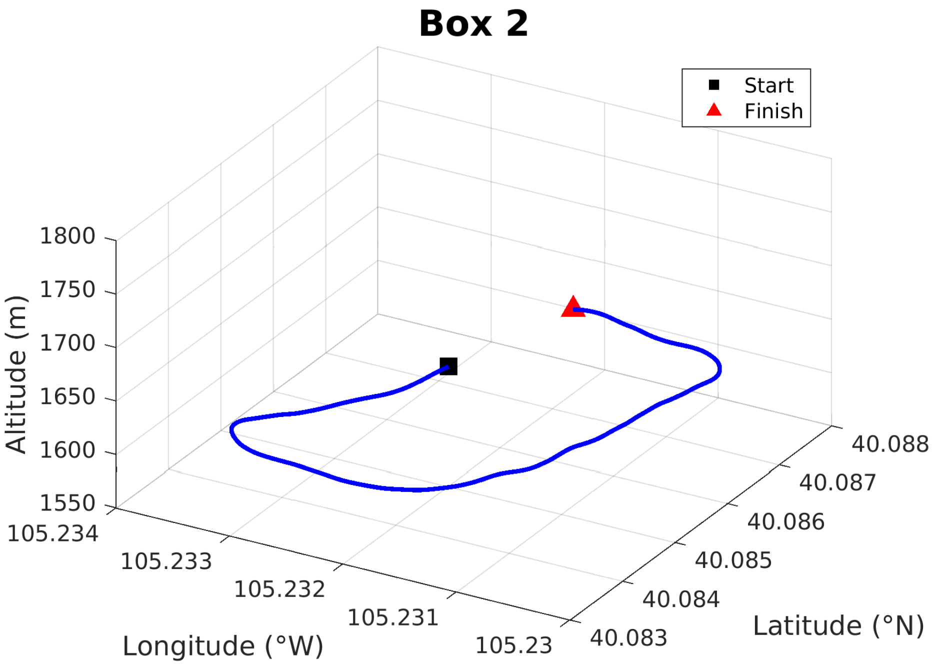


Figure B.6: Box 2

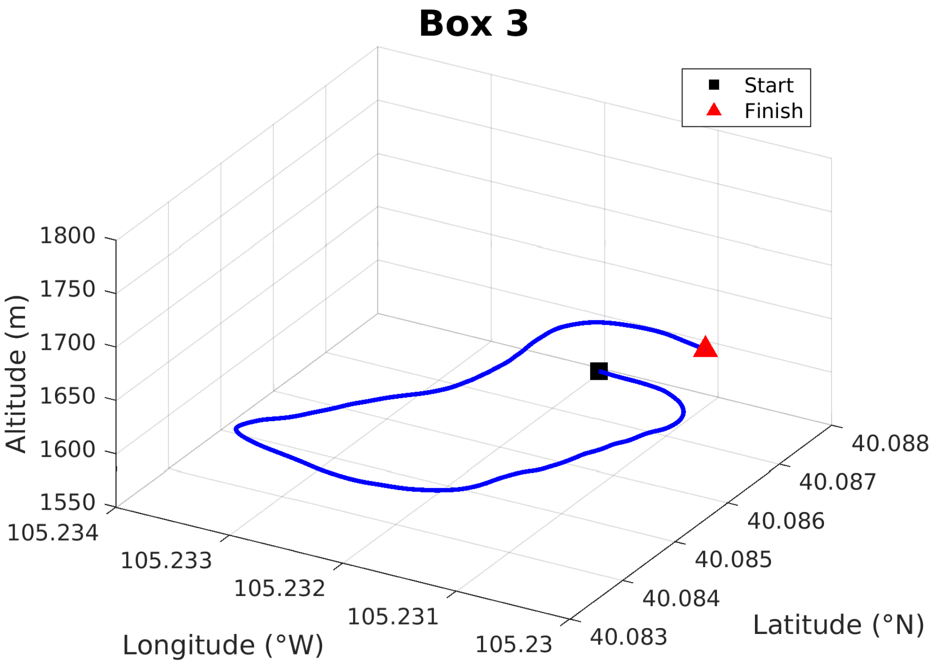


Figure B.7: Box 3

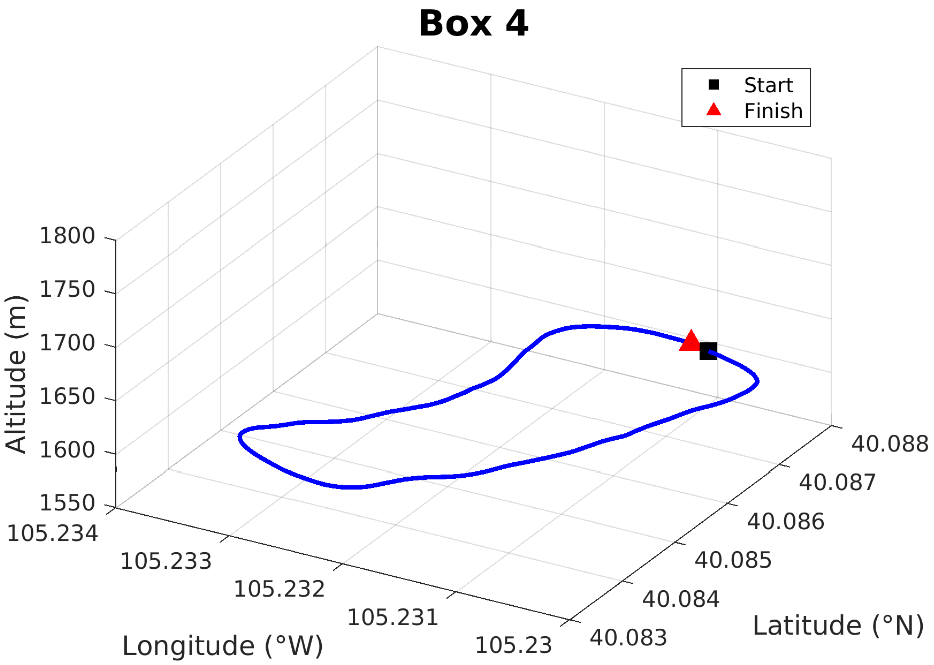


Figure B.8: Box 4

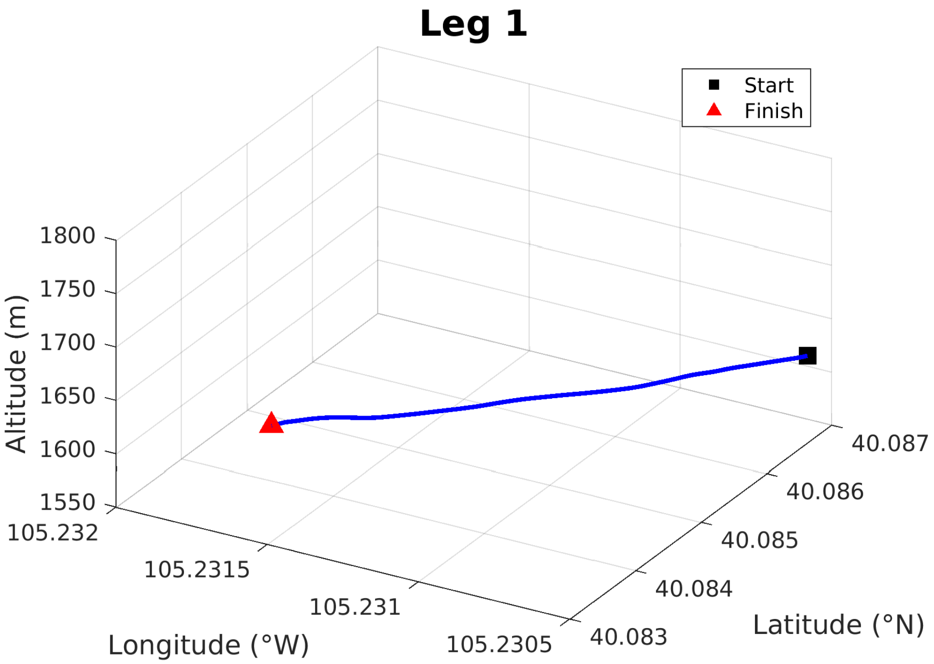


Figure B.9: Leg 1

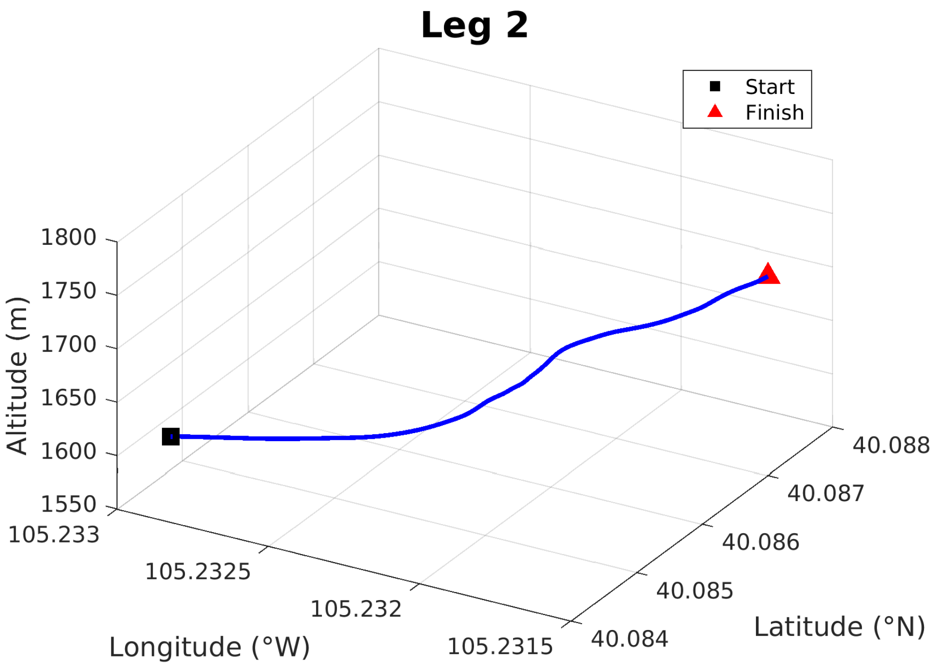


Figure B.10: Leg 2

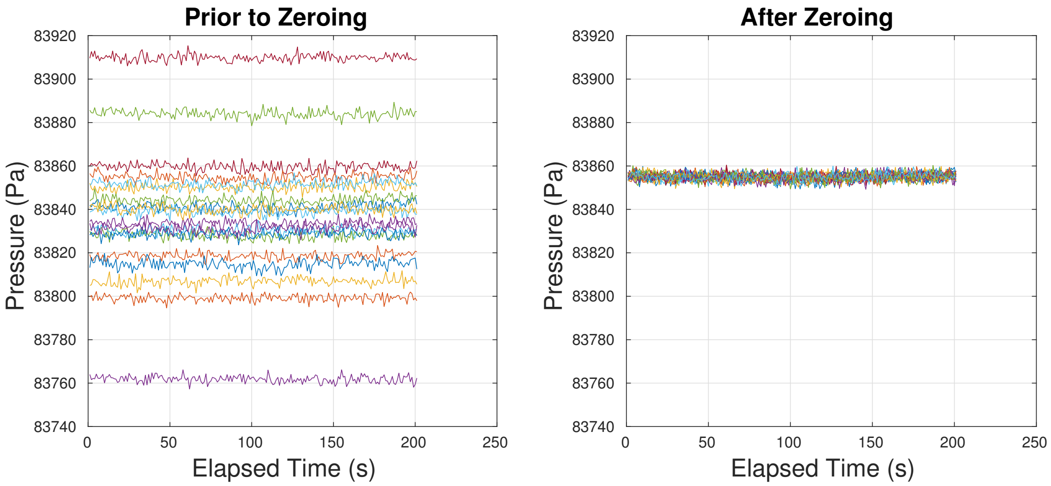


Figure B.11: Pre-flight zeroing of pressure data.

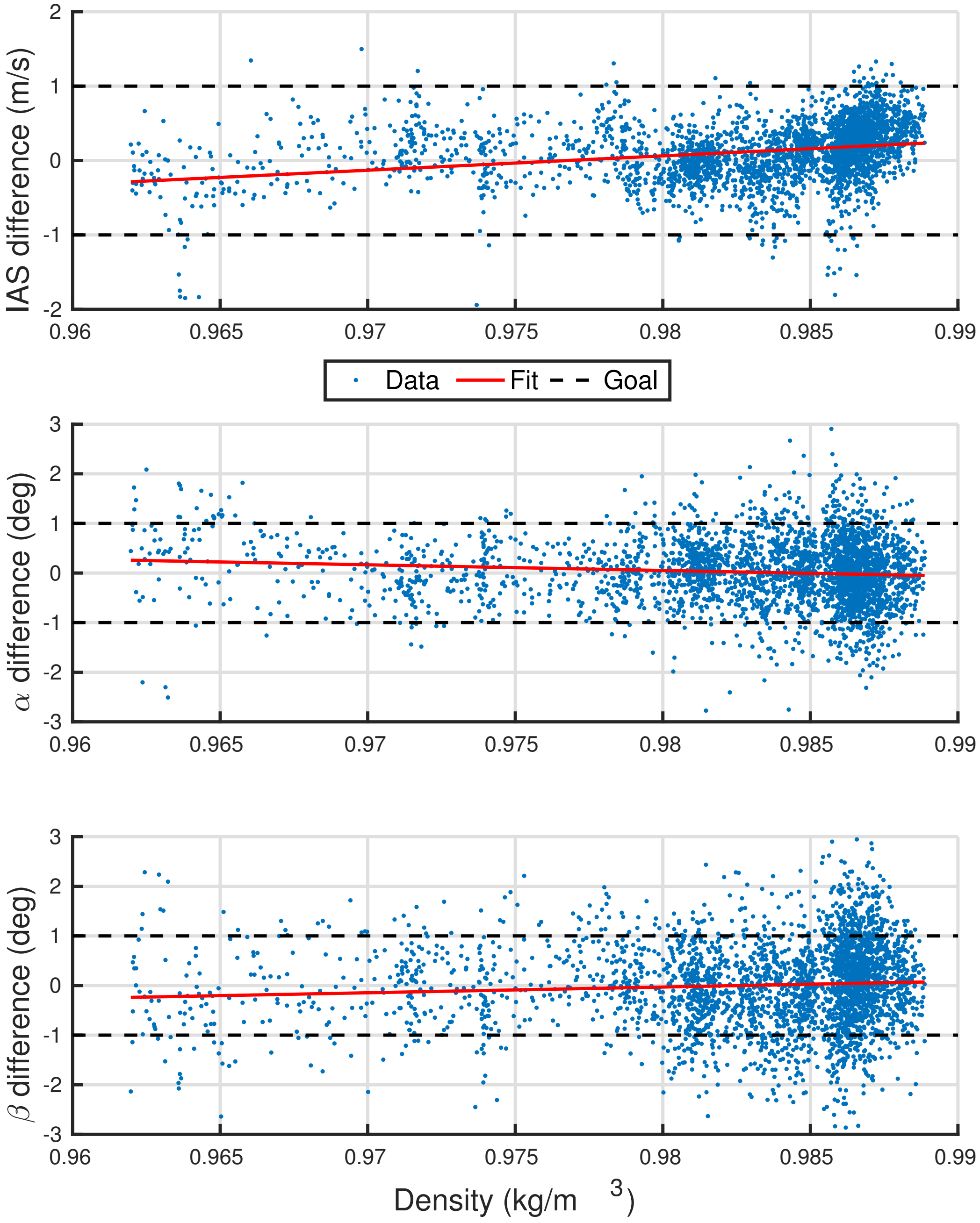


Figure B.12: ρ effects on validation errors.

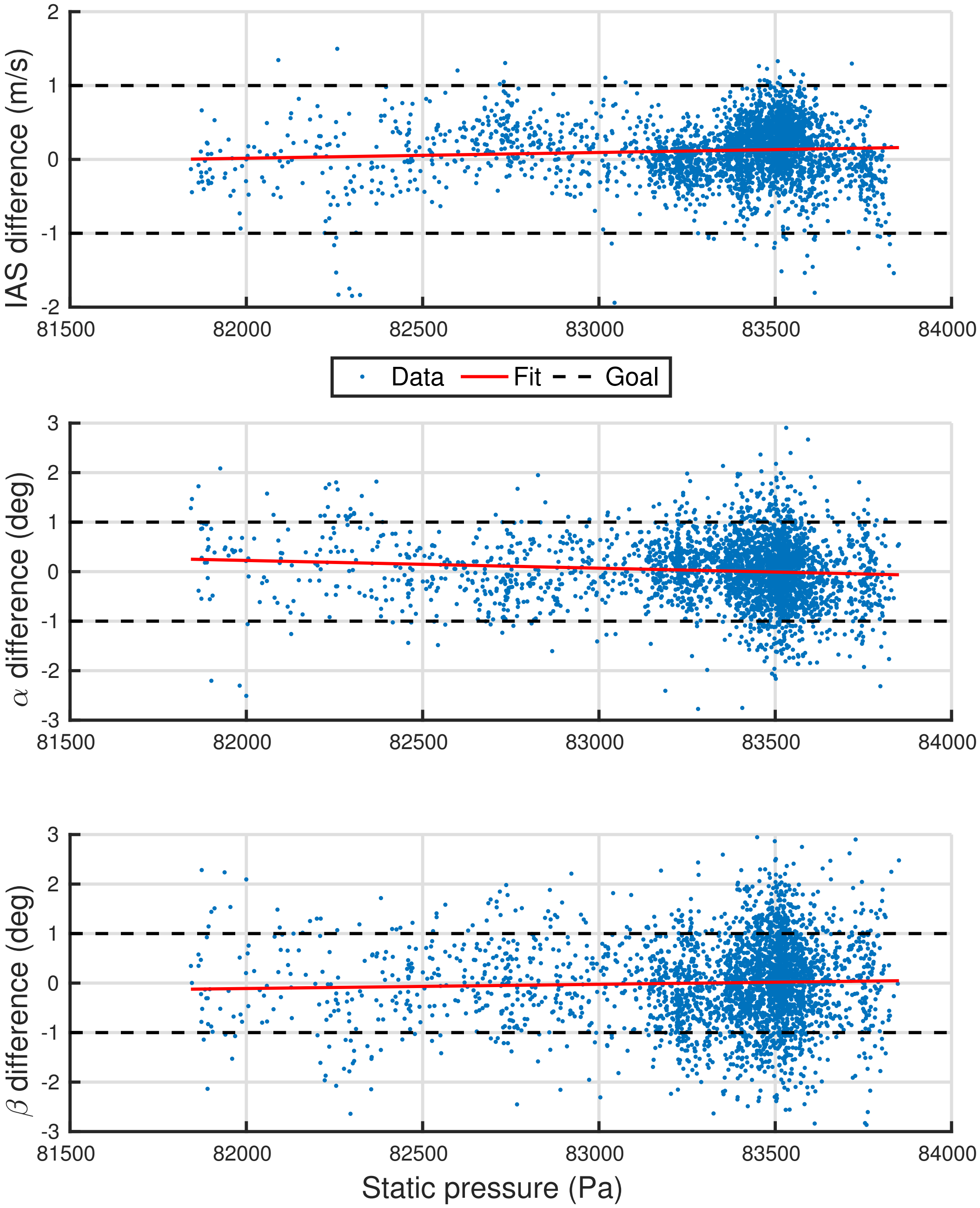


Figure B.13: P_s effects on validation errors.