TÉCNICO LISBOA

Synthetic Air Data System: An alternative to conventional sensors

Francisco Xavier Rodrigues Correia

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisor: Prof. António Manuel dos Santos Pascoal

Examination Committee

Chairperson: Prof. José Fernando Alves da Silva
Supervisor: Prof. António Manuel dos Santos Pascoal
Member of the Committee: Prof. Fernando Manuel Ferreira Lobo Pereira

October 2020

Synthetic Air Data System: An alternative to conventional sensors

Francisco Xavier Rodrigues Correia

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisor(s): Prof. António Manuel dos Santos Pascoal
Eng. Tiago João Parra Fidalgo Ramos Marques

Examination Committee

Chairperson: Prof. José Fernando Alves da Silva
Supervisor: Prof. António Manuel dos Santos Pascoal
Member of the Committee: Prof. Fernando Manuel Ferreira Lobo Pereira

October 2020

"*The art and science of asking questions is the source of all knowledge.*"

Thomas Berger (1924 - 2014)

Acknowledgments

Participating in this project was a remarkable experience, and the challenge it provided is something I am glad to have accepted, and for sure will always remember.

First and foremost, the development of this thesis would not have been possible without the constant support and dedication of engineer Tiago Marques, who other than trusting me with this challenge, helped me through all the steps of the thesis development. I also want to express my gratitude to professor António Pascoal, who accepted to embark on this project. He helped and guided me in a series of endless situations.

To my parents and siblings, a very special thank you for all the constant support, care and dedication you gave me. It would not have been possible without you.

To my longtime friends from São Miguel, thank you for being there for me and for all the good time and enjoyable moments we had.

I also want to express a special gratitude to my friends Pedro Santos, Nuno Ferreira and Fernando Roque, that made possible a smooth and amusing progress in Instituto Superior Técnico. To all my countless friends from Residência Egas Moniz, thank you for helping me grow up, and for all the late night talks, dinners and uncountable memorable moments we had together.

Thank you all for these 5 unforgettable years.

Resumo

Um *Air Data System* é constituído por vários sensores convencionais que providenciam informação relativa ao ar circundante da aeronave. Um exemplo disto é o caso da velocidade da aeronave em relação ao ar, que é altamente dependente do sensor tubo de pitot. A este tipo de sensores, estão normalmente associados problemas de obstrução do tubo por gelo, o que leva a que o sensor forneça medições erradas, e a solução para isto é normalmente a instalação de pelo menos 3 unidades deste sensor, de modo a obter um sistema fisicamente redundante. O aumento do número destes sensores leva a um aumento na complexidade e aumento no peso do sistema total, o que no caso dos veículos aéreos não tripulados pode não ser muito desejável, ou até mesmo impossível. A presente tese estuda a possibilidade de utilizar dois modelos aproximados da cinemática do avião, aplicada num algorítmo de Mínimos Quadrados, de forma a estimar a velocidade do ar e do vento. Tirando proveito de manobras aéreas, e utilizando medições de um *Global Positioning System* (GPS) e de um *Attitude and Heading Referrence System* (AHRS), foi possível testar um sistema capaz de fornecer uma redundância "virtual" aos dados de velocidade do ar, adquiridos pelo sistema padrão. As condições para o sistema proposto foram estudadas e validades, e as limitações do algorítmo, quando submetido a condições inesperadas, foram analisadas. A tese conclui propondo melhorias, ao utilizar modelos mais precisos da cinemática, ou sugerindo alternativas para otimizar o esforço computacional do sistema.

Palavras-chave: *Air Data System*, Tubo de pitot, Mínimos Quadrados, Velocidade da aeronave em relação ao ar, Velocidade do vento.

Abstract

A standard Air Data System is comprised of several conventional sensors that, together, provide key information regarding the air surrounding an aircraft. An example of this is the case of the aircraft's airspeed, which is highly dependent on a pitot tube sensor. A sensor of this type can be obstructed due to icing, leading to wrong measurements of the airspeed, and a solution is to install at least three sensor units, in order to obtain a physically redundant system. An increase in the number of sensors, brings about an increase in the complexity and weight of the system, which in the case of Unmanned Aerial Vehicles may not be desirable or even impossible to achieve. The present thesis studies the possibility of using two approximate kinematic models of an aircraft, applied in a Least Mean Square algorithm, to estimate the wind field and aircraft's airspeed. By exploring the aircraft's maneuvers, and relying on measurements from a GPS and an AHRS, a system was designed that acts as a "virtual" redundant system, to help in the estimation of airspeed. The conditions underlying the design of the proposed system were studied and validated, and the limitations of the algorithm for airspeed estimation, when the aircraft is subjected to unexpected situations, were analyzed. The thesis concludes by proposing improvements to the method derived, using more accurate models of the aircraft's kinematic and exploiting alternatives to optimize the computational effort required.

Keywords: Air Data System, Pitot tube, Least Mean Squares, Airspeed estimation, Wind estimation.

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

α	Angle of attack.
β	Angle of sideslip.
χ_c	Crab angle.
χ_c	Ground course angle.
γ	Flight path angle.
γ_a	Air-mass-referenced flight-path angle.
$\overrightarrow{GS}$	Ground speed vector.
$\overrightarrow{TAS}$	Airspeed vector.
$\overrightarrow{W}$	Wind speed vector.
ψ	Heading angle.
ρ	Air density.
ρ_0	Air density at standard sea level.

Roman symbols

A	Speed of sound.
M	Mach number.
P_S	Static pressure.
P_T	Total pressure.
Q_C	Impact pressure.

Subscripts

i, j, k	Computational indexes.
x, y, z	Cartesian components.

Superscripts

$\overrightarrow{(.)}$ Vector.

T Transpose.

Glossary

ADC Air Data Computer.
ANNs Artificial Neural Networks.
CAT Calibrated Air Temperature.
DCM Direction Cosine Matrix.
ECEF Earth-Centered Earth-Fixed.
EGM Earth Gravitational Model.
EKF Extended Kalman Filter.
GNSS Global Navigation Satellite System.
GPS Global Positioning System.
IMU Inertial Measurement Unit.
LMS Least Mean Square.
MSE Mean Square Error.
NED North-East-Down.
SADS Synthetic Air Data System.
SAT Static Air Temperature.
TAS True Airspeed.
TAT Total Air Temperature.
WGS World Geodetic System.
WMM World Magnetic Model.

Chapter 1

Introduction

1.1 Motivation

Since the early days of aviation, one of the key factors to guarantee safety and good flight performance for a given aircraft has been the ability to control it. For this purpose, a great deal of research has been done on the different systems required for aircraft control. Among these, one of paramount importance is that in charge of acquiring, in real time, measurements of important physical variables. The latter include airspeed, V_a, and the orientation of the aircraft with respect to the airflow, quantified by its angle of attack, α, and sideslip, β.

The standard method of measuring these variables is through a number of physical sensor suites, the most common being a differential sensor pressure attached to a pitot tube and/or wind vanes. Atmospheric conditions can be critical to the well-functioning of these sensors that can, for example, be obstructed due to icing, resulting in faulty measurements. An attempt to overcome this problem is the installation of multiple sensors, as a means to produce a physically redundant system.

Whether the aircraft is operated by a human pilot or an automatic control system, an accurate estimate of the air data triplets V_a, α, and β is essential to the efficient and safe operation of the aircraft. Even if the sensor suite has a high level of redundancy, all sensors are subjected to the same environmental conditions, which means that failure on any individual pitot tube is also likely to occur on others, under certain atmospheric conditions. On February 6, 1996, Birgenair Flight 301 [1] crashed shortly after take off from Puerto Plata in the Dominican Republic for Germany due to a blocked pitot tube, taking all 189 human lives aboard the aircraft. Also, on June 1, 2009, on a flight from Rio de Janeiro to Paris, an Airbus A-330 [2] aircraft crashed during a thunderstorm over the Atlantic Ocean, due to the obstruction of the pitot tubes, killing all 228 passengers and crew members, making the Air France flight 447 the deadliest accident in the history of Air France.

Airspeed information is mandatory to operate any aircraft: flying too slowly can cause the plane to stall and flying too fast can cause a structural break up. Both previous accidents were caused by the malfunction of pitot tubes, resulting in incorrect readings and consequently causing the planes to stall.

It is interesting to note that the pitot tube, which is universally used on aircrafts for measuring airspeed

was invented over 250 years ago by the French mathematician and scientist Henri Pitot with the intention of measuring the flow of water in rivers and canals. This invention turned out to be extremely important in modern aviation.

Still, the inconvenience of depending on physical sensors susceptible to malfunction during certain weather conditions, among other reasons, led specialists to explore alternatives to obtain air data key information. This drove the current study of the so called Synthetic Air Data System (SADS).

1.2 Objectives

The main objective of this work is to investigate and build a synthetic air data system, capable of estimating the airspeed of an aircraft.

The challenge proposed by TEKEVER is to think unconventionally and build a SADS, with the smallest amount of sensors at hand. It is described by two fundamental requirements: the elimination of both the pitot tube sensor and the aircraft dynamic model.

The underlying goal is to investigate whether or not, both requirements can be accomplished, and define its conditions. Besides the two requirements just mentioned, the drones set to be used in testing don't possess wind vanes, for measuring angle of attack and/or sideslip, leaving at hand little more than data from a GNSS sensor, that provides information concerning geo-spatial positioning, like position, speed, bearing, track, etc; an AHRS, that provides information of attitude information, including roll, pitch and yaw, and heading information in a 3D space.

In sum there are three conditions to build the SADS,

- Elimination of pitot sensors;
- Elimination of the aircraft dynamic model;
- Elimination of wind vanes sensors.

The need to estimate the airspeed without any direct measurement and with the least possible information, consequently leads to the knowledge of the wind velocity, which without any further information ends up being one of the parameters of estimation as well.

The missions specified by the company TEKEVER mainly consist of endurance missions, in which the aircraft spends most of its time in a loiter maneuver, at the same height. Following these requirements, the work of C.Hurter et al. [3] was explored to fit the current requirements and, apart from the approximations made by the authors, it was also investigated whether or not, their work could be extended to encompass a theoretical kinematic model suggested by William Premerlani [4].

1.3 Thesis contribution

The main contributions of this thesis address and contribute to the solution of the following problems:

- Given a yaw maneuver, for a steady flight level, provide real-time estimation of the airspeed and wind components, eliminating both the use a dynamic model and a pitot tube sensor;
- After successfully estimating the wind components, provide a time limit to use the last wind estimates in a situation in which, the continuous estimation of the wind is not possible;
- Provide a ”virtual robustness” framework to back up the standard air data system.

1.4 Thesis outline

The outline of this thesis was selected in such a way as to reflect the several steps of the work, and is therefore organized in six Chapters.

Chapter 1 contextualizes the motivation and the main objectives that led to the work here provided. Also, the main contributions are succinctly described.

Chapter 2 presents the State of the Art on the subject of Synthetic Air Data System, and all relevant past work, regarding such topic.

Chapter 3 concerns all relevant background information, necessary to the comprehension of the work as a whole. It starts with the important definition of standard Air Data Systems and the processing of the information inherent to them. Following, there is an insight on reference frames and rotation matrices, and on vectorial relation between the ground speed, airspeed, and wind speed. Also, the Least Squares method is detailed, as well as the condition number of a matrix. The chapter wraps up with two approximations to the modelation of the kinematic behaviour of the aircraft.

Chapter 4 starts off with the validation of the wind velocity remaining approximately constant in a time period that is quantified, which is essential to the assumptions taken later in the chapter. This is followed by the solution found for the problem, which involves incorporating each of the two models, described in Chapter 3, in a Least Squares problem, and the restrictions inherent to the problem. Validation is the last topic, alongside a comparison between the two models used, using simulated data.

Chapter 5 refers to the experimental results obtained with real flight data. All results are discussed and the differences obtained with the two models used are emphasized.

All the analysis using simulated and real flight data, in Chapters 4 and 5, respectively, was done using the MATLAB® software. In the end, Chapter 6 summarises the conclusions drawn throughout the developed work, the main achievements, and possible improvements for future work.

Chapter 2

State of the Art

In the aerospace industry, there has always been room for technological improvement. With the appearance of increasingly powerful computers, the need and opportunity to develop more efficient computational air data systems has attracted the attention of worldwide researchers. A good example of this is the work of Nazaroff [5] in 1972. A requirement, when applying linear control theory techniques to practical problems, is the knowledge of the system's state vector which can be obtained using an observer to perform state reconstruction. In his work, Nazaroff [5] proposed a systematic method, capable of locating the observer eigenvalues with respect to a quadratic error criteria. On a different perspective, in 1976, Duffy [6] studied a filtering architecture with a view to reducing the computational effort required to estimate system states and biases. The work provided a decoupled solution for filter and smoothing algorithms applied to a maneuvering re-entry vehicle.

As consequence of the increasing development of avionics sectors, the interest in the concept of Synthetic Air Data System has been increasing in the last years, especially in the drone industry, where the number of sensors is a stringent limitation. A Synthetic Air Data System is not defined as a unique method, but rather by a set of multiple approaches with the intention to finding an alternative to specific classic methods. Depending on various criteria, the problem can be simply defined as that of estimating single or multiple variables of interest without the explicit use of classical sensors. In our case, the main focus is on estimating the airspeed of the aircraft, as a way to increase the necessary redundancy to the limited pitot tubes available.

Zeis [7] is often acknowledged as one of the pioneers of the SADS concept. His work is rooted in previous theoretical work [8–11] on the estimation of the angle of attack α, that borrows from the results in Duane Freeman [9], where α is estimated using as measured data acceleration inputs and surface positions during a given maneuver. Using standard inertial measurement units, Zeis [7] assumed the availability of airspeed measurements along with accurate knowledge of the lift coefficient C_L to estimate the angle of attack and found the estimator to be highly feasible. However, his approach still requires a pitot tube and the accuracy of the system was highly dependent on the availability of accurate stability derivatives. Also, due to the computational performance available at the time, a significant limitation was the lack of a more powerful regression tool that would allow for the incorporation of flight regimes not

modeled by his estimator.

Wise [12] exploited the use of an Extended Kalman Filter framework to estimate the angle of attack and sideslip, considering decoupled longitudinal and lateral models. He required information data from dynamic pressure and used an inertial measurement unit to obtain measurements of Euler angles, body rates, and accelerations. As a result, the estimation accuracy was highly dependent of the aerodynamic database accuracy.

Austin Murch [13] developed an angle of attack command augmentation system for the NASA AirSTAR flight-test infrastructure, that proved to be successful in estimating the angle of attack in real time, relying on a pitot-tube combined with static ports. The approach taken is an algebraic process and the stochastic nature of the sensor was not taken into account.

Using information from GPS velocity, airspeed estimated from engine throttle, and the pitot tube based airspeed, Hansen [14] explored the possibility of detecting faults on pitot tubes. The available information was used as a way to achieve redundancy and the statistical properties of residuals were investigated. Two detectors were desingned, and both were able to identify the fault within an acceptable time. Later, Hansen [15] expanded the work described above to include wind estimation, and analysis of faults.

Fravolini et al. [16] focused on airspeed estimation and fault monitoring, and estimated the airspeed based on the dynamic equation of the angle of attack. The authors proposed two methodologies: the first assumed the aerodynamic coefficients and geometric and inertial parameters to be known, while the second used a Least Square optimization to identify the parameters.

With the continuous evolution of technology, Artificial Neural Networks (ANNs) have been exploited and have shown to be able to estimate the airspeed with quite good precision. Srikanth Gururajan et al. [17] explored the possibility of accommodating ANNs in a sensor failure detection scheme and evaluated two networks: the Extended Minimal Resource Allocating Network and a Multilayer Feedforward NN. Both were validated with flight data and had similar performance as on-line estimators in what concerns the standard deviation of the airspeed tracking error. However, the downside of ANNs is that they require a reasonable amount of reliable flight data to validate the model, besides the fact that they can only be used for the specific aircraft under the flight conditions that they used as training sets.

In 2013, F. Lie and Demoz Egziabher. [18] developed a method that enabled the estimation of the triplet V_a, α and β. Their method suggested using a cascaded Extended Kalman filter architecture, in which information from an Inertial Measurement Unit (IMU) and a Global Navigation Satellite System were fused, taking also into account a dynamic model of the aircraft. The first filter uses information from the IMU and GNSS to estimate the aircraft's attitude and absolute velocity, which are then used as measurement updates for the second filter and fused with the dynamic model to produce the estimates of airspeed, angle of attack, and sideslip. The result was an algorithm capable of generating estimates close to the conventional sensors and with high insensitivity to modeling errors.

Just as in the case of F. Lie et al. [18], nowadays, the most common methods of SADS exploit the use of high fidelity aircraft dynamic models. However, to obtain the dynamic model is a complex and expensive task, and the parameters of the aircraft change over time due to the propulsion system

and the natural variation of the airframe parameter. With a view to overcoming these difficulties, new methods have been explored to estimate the airspeed without requiring an explicit aircraft model.

Rudy et al. [19] studied a model-free method that excludes the use of pitot tubes. The approach relies on the use of a nonlinear Kalman filter that estimates the airspeed fusing measurements from an IMU, GNSS, and wind vanes. This approach proved to yield good results and has the high advantage of being independent of the aircraft. It should be noted, however, that the approach of Rudy et al. [19] still requires wind vanes to get information of the air flow.

One way of looking into the problem of estimating the airspeed indirectly is by estimating the wind speed. If we redirect our attention to the wind and succeed in estimating its components with reasonably accuracy, while having access to the components of the inertial velocity, the true airspeed becomes a consequent solution of the problem.

The wind is a stochastic process that varies in space and time. Wind estimation has been addressed by many authors. For example, Patcher [20] studied the problem of estimating the heading of a Micro Air Vehicle flying in an unknown wind by fusing data from GPS and air data measurements. He was successful in estimating the horizontal components of wind speed and vehicle heading, but did not address wind gradients or wind acceleration. Rodrigues et al. [21] used an optical flow sensor in the longitudinal and lateral directions to estimate the aircraft's crab-angle, which he combined with measurements of GPS and airspeed to compute the wind velocity. Nicola de Divitiis [22] described a method to estimate the wind velocity from measurements of a small ducted fan vehicle using an approach based on adjoint differential equations that describe the wind-velocity evolution.

Along similar lines, Langelaan [23] researched how to estimate three-dimensional wind components. Using the GPS velocity and measurements of acceleration, combined with the vehicle state information, he was able to compute the wind directly. Also, he concluded that the error in wind velocity and acceleration estimates depended on the vehicle airspeed and on the noise associated with the sensor.

Lee et. al. [24] used a Square Root-Unscented Kalman filter to estimate 3D time-varying wind with turbulence and investigated the convergence limitations of the estimations.

Rhudy el al. [25] exploited a few promising methods to estimate the wind vector, but did not provide an observability analysis and evidence of downward wind component estimation. Later, a nonlinear observability analysis was done to back up the previous methods, in which canonical flight maneuvers were examined by researchers at the Department of Aerospace Engineering and Mechanics, University of Minnesota, Twin Cities [26].

In 2014, C. Hurter et al. [3] attempted to extract wind parameters out of the analysis of the trajectory of a given aircraft. Using a 2D approach and neglecting the vertical wind component, they presented a solution that, with enough measurements, could estimate the North and East components of the wind making use of a least mean square algorithm. Using radar measurements of several aircraft and a cluster of data, they were able to successfully estimate the required wind components. This, however, required a large amount of input data depending on the flight conditions, besides the fact that the aircraft had to fly at an approximately constant airspeed during the data acquisition process.

In summary, advances of science and technology have brought significant opportunities for research

in the area of air data, and nowadays the most common tool to process air data still remains the use of a dynamic model combined with observations of several sensors fused in a Kalman filter. Several approaches offer solutions to this problem, but the biggest restriction is in the non-elimination of the dynamic model and/or a pitot tube, which was found to be successfully implemented when studying specific maneuvers at a given flight level.

Chapter 3

Theoretical Background

This chapter aims to cover the fundamental knowledge behind the work developed. It starts with an analysis of an Air Data System (ADS), its components and purpose, followed by a basic overview of coordinates reference systems. There is a briefly explanation of an important algorithm, the Least Mean Squares, and last but not least there is a short insight of the theoretical kinematic models used in this work.

3.1 Air Data System

An Air Data System is a key item for the well functioning of all avionic systems during flight, and consists of aerodynamic and thermodynamic sensors and associated electronics [27]. This system is in charge of collecting and processing air data and then present it to the pilot through flight displays or forwarding it to other sub-systems, such as autopilots, weapon system fire control computations, and control of cabin-air pressurization systems [27]. Data such as pressure altitude, vertical speed, calibrated airspeed, true airspeed, Mach number, angle of attack, static air temperature and air density ratio are examples of data processed by this system.

The acquisition of air data consists of measuring three fundamental parameters:

- Total pressure P_T;
- Static pressure P_S;
- Total air temperature TAT.

Total pressure P_T is the sum of the local atmospheric pressure P_S and the impact pressure Q_C, and can be measured by an absolute pressure sensor connected to a pitot tube facing the moving airstream [28]. Equation 3.1 represents the algebraic relation just described. Figure 3.1 illustrates a pitot tube, the required sensor to measure total pressure.

$$P_T = Q_C + P_S \tag{3.1}$$

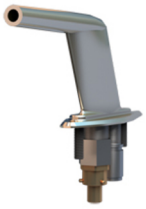


Figure 3.1: Pitot tube (Adapted from [29]).

Static pressure P_S is the absolute pressure of motionless air surrounding the aircraft, i.e, is the barometric pressure at the altitude where the aircraft is traveling and is independent of any pressure disturbances caused by the motion of the aircraft [30]. Static pressure P_S can be measured with a pitot-static tube or a flush-mounted port on the fuselage, also known as a static port. The static ports can be visualized in figure 3.2.

Figure 3.2: Static ports (Adapted from [31]).

Impact pressure Q_C is the pressure produced by a moving stream against a surface that brings part of the moving stream to rest [30]. The acquisition of this data is done through the solution of equation 3.1.

Total Air Temperature TAT gives a measure of the Static Air Temperature SAT plus the kinetic rise in temperature K_r, due to air being brought to rest relative to the temperature sensing probe [27]. This relation is described by

$$TAT = SAT + K_r \tag{3.2}$$

Total Air Temperature is measured from a temperature probe (figure 3.3) on the body of the aircraft and can sometimes be combined in a sensor, along with pressure gathering [30].

Figure 3.3: Total air temperature sensor (Adapted from [29]).

3.1.1 Air Data Computer

The measurements of pressure and temperature are accomplished by transducers that convert the sensed parameters into mechanical motions or electrical signals [28]. The Air Data Computer (ADC) is now introduced as the core of the system, a purely electronic equipment that receives the signals of the transducers and is responsible for processing and redirecting them to all necessary sub-systems [32].

A basic representation of a complete air data system is illustrated in Figure 3.4.

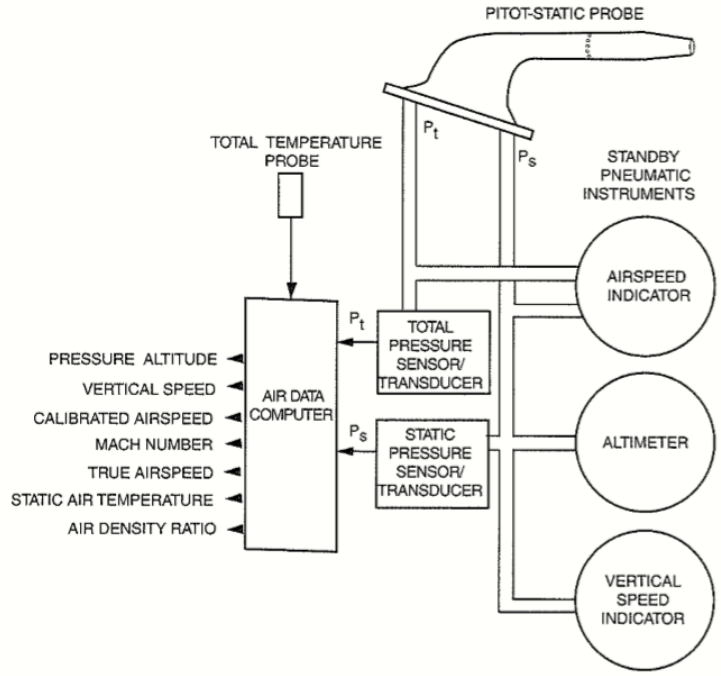


Figure 3.4: Air Data System (Adapted from [33]).

The system consists of three sensors (total temperature probe; total pressure sensor(pitot tube); static pressure sensor); three standby pneumatic instruments: airspeed indicator; altimeter; vertical speed indicator; and finally, the air data computer.

The air data computer can be represented as a block with 7 main outputs, briefly described next:

1. **Pressure altitude,** P_S: By definition, the pressure altitude is the height of the aircraft above sea level derived from the measurement of static pressure P_S assuming a standard atmosphere, and is the standard parameter used for vertical navigation in controlled airspace [27].

2. **Vertical Speed,** V_V: The vertical speed of altitude rate is, as the name suggests, the rate of altitude that the aircraft is being subjected to and can be derived differentiating the static pressure P_S.

3. **Calibrated Airspeed, CAS**: This is the speed that, under standard sea level conditions, would result in the same impact pressure as the one measured on the aircraft and can be derived directly from the impact pressure Q_C.

4. **Mach number, M**: The Mach number is mathematically defined as the ratio of the true airspeed versus the speed of sound, that is, $M = \frac{TAS}{A}$. It and can be obtained from the ratio of the impact pressure versus the static pressure: $M = \frac{Q_C}{P_S}$.

5. **True airspeed, TAS**: Defined as the speed of the aircraft relative to the air; it is usually computed from the Mach number relationship through $M = \frac{V}{A}$.

6. **Static Air Temperature, SAT**: Also known as Outside Air Temperature (OAT), the SAT is the actual temperature of the air outside the aircraft, without the kinetic effect of the aircraft's movement. The correction factor for the kinetic heating effect can be derived by using the recovery ratio of the probe together with the Mach number, which can then be used to derive the SAT using the relation described by equation 3.2.

7. **Air density ratio,** ρ/ρ_0: This parameter can then be computed from the measurements of static pressure P_S and temperature TAT. It should be noted that ρ_0 is the air density at standard sea level conditions.

An important note is that the manner in each of the parameters are obtained in the air data computer were not taken into account too deeply given the fact this is not significant for the work developed.

3.2 Reference frames, rotations

When working with aircraft systems, it is important to fully understand the motion of an aircraft relative to the earth. The choice of a frame of reference can be arbitrarily, and the use of more than one option will yield identical results. Still, it should be noted that some coordinate systems are more intuitive and suitable to work with than others, thus it is convenient to use common reference frames. From this perspective, we consider in the present work two main reference frames: the first one is fixed to the Earth and is an inertial reference system; and the second one is fixed to the aircraft and referred to as the body reference system.

Besides some of the reasons previously referred, the use of these two different reference frames is especially important when dealing with some of the following situations [34]:

- Motion is easily described in a body-fixed reference frame, but Newton's equations of motion are derived relative to a fixed, inertial reference frame.

- Aerodynamic forces and torques act on the aircraft body and are easily described in a body reference frame.
- On-board sensors like accelerometers and rate gyros measure information with respect to the body frame. GPS, on the other hand, measures position, ground speed and course angle, with respect to the inertial frame.
- Waypoints, flight trajectories and map information are specified in an inertial frame.

When working with more than one reference system, it becomes necessary to work our way around and transform all necessary parameters into a single one. One coordinate system can be transformed into another through two operations: rotation and translation.

3.2.1 Coordinate systems

The North-East-Down reference system

Since the Earth is moving and rotating, it is not so obvious to define an inertial frame attached to it. For this purpose, the starting point is to choose a common reference frame attached to the earth, the Earth-centered, Earth-fixed frame (ECEF). The ECEF has its origin at the center of the Earth, the x-axis points to the intersection between the Equator and the Prime meridian (latitude 0º, longitude 0º), the Y-axis points to 0º latitude and 90º longitude and the Z-axis to the North Pole (latitude 90º along the axis of rotation of the Earth). To use this coordinate system, the shape of the surface of the Earth should be described using the widely adopted WGS-84.

The World Geodetic System (WGS) is a standard that includes the definition of the coordinate system's fundamental and derived constants, the ellipsoidal Earth Gravitational Model (EGM), a description of the associated World Magnetic Model (WMM), and a current list of local datum transformations. The WGS-84 is its latest revision.

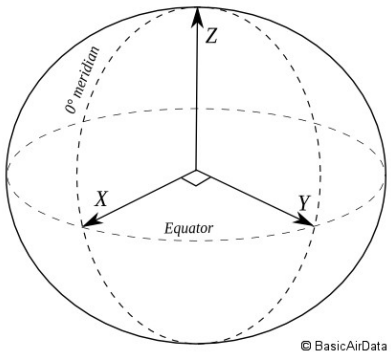


Figure 3.5: Earth-centered,Earth-fixed frame (Adapted from [35]).

From this point on, the flat earth assumption is considered, at least locally: the earth is flat, nonrotating, and an approximate inertial reference frame. The acceleration of gravity is constant and perpendicular to the surface of the earth. This assumption, however, will not hold in cases where the vehicle travels far away from its origin, in which case the system equations will ”drift away” from reality.

With ECEF at hand, it is possible to obtain a locally inertial reference system, in which it is acceptable to describe trajectories and altitude, called the North-East-Down (NED) reference system: this reference system has its origin fixed in the ECEF coordinates, the “North” axis points North in the local meridian direction, the “East” axis points East in the local parallel direction and the ”Down” axis is perpendicular to the other two and points towards the Earth.

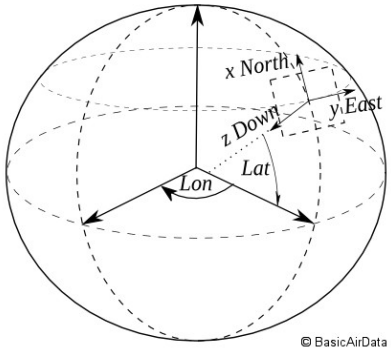


Figure 3.6: North-East-Down frame (Adapted from [35]).

Body reference system

The body frame is described with its origin at the center of gravity of the aircraft. The X-axis points toward the aircraft nose, the Y-axis along the right wing and the Z-axis points down, towards the Earth.

The body can be oriented with respect to the fixed coordinate frame, NED. This orientation can be parameterized locally by the so-called Euler angles.

The designation of each of the Euler angles and their range, in radians, is as follows (see Figure 3.7):

- Roll: $\phi \in [-\pi, \pi]$;
- Pitch: $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$;
- Yaw: $\psi \in [0, 2\pi]$.

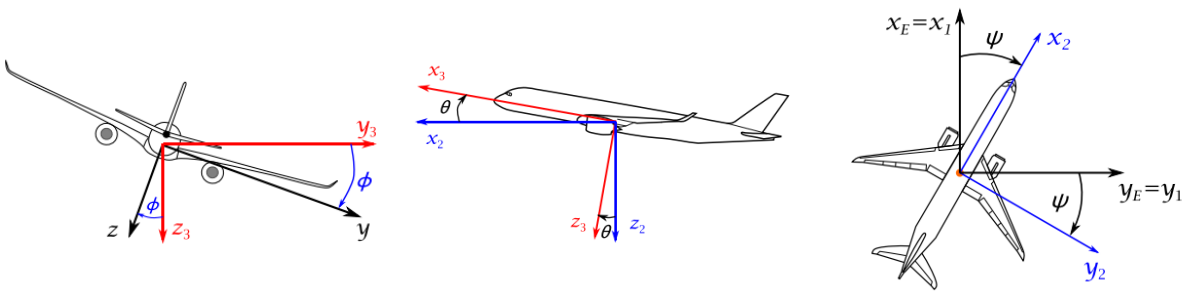


Figure 3.7: Euler angles: Roll, pitch and yaw, respectively (Adapted from [36]).

Euler angles are not the only way to describe the orientation of the aircraft in an inertial frame. A major problem for the use of this notation are the singularities that occur for $\theta = \left\{\pm\frac{\pi}{2}\right\}$. This is usually known as the *gimbal lock*, and occurs when the nose of the aircraft is pointed straight up or down. In

this situation, the transformation from one reference frame to another breaks, and the integration of the Euler angles becomes indeterminate. [37]

A common alternative is the use of quaternions. These are widely used as parameters to represent the attitude of rigid bodies and present strong advantages such as not having any singularities, and being computationally more efficient.

On the other hand, their physical meaning is not as intuitive as Euler angles and for it, Euler angles were the notation used in this work to describe the rotation of the aircraft, because during the maneuvers, anticipated singularity conditions will never be reached.

3.2.2 Coordinate system transformations

When working with different reference frames, it becomes necessary to convert vectors from one frame to another in order to properly work with all the available data. This being said, a vector in any frame can be represented in another, given the correct transformation.

The relation between two different reference frames is described in equation 3.3, where the superscript indicates the coordinate frame in which the vector is resolved. Superscript $(.)^I$ indicates the local navigation system and the superscript $(.)^B$ the body frame.

$$X^I = R_B^I X^B \tag{3.3}$$

The matrix R_B^I is called a Direction Cosine Matrix (DCM) and represents the required rotation to characterize an original vector from a certain coordinate frame into another, in this case to represent the vector X, originally expressed in the body frame, in the inertial one [38].

Overall, a coordinate transformation relates the components in one frame to the components in another. To obtain a transformation from the NED to the body frame, it is necessary to follow a sequence of rotations: a positive rotation about the z-axis by an angle ψ (Eq. 3.4); a positive rotation about the y-axis by an angle of θ (Eq. 3.5); and a positive rotation about the x-axis by an angle ϕ (Eq. 3.6).

$$R(\psi) = \begin{bmatrix} cos(\psi) & sin(\psi) & 0 \\ -sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

$$R(\theta) = \begin{bmatrix} cos(\theta) & 0 & -sin(\theta) \\ 0 & 1 & 0 \\ sin(\theta) & 0 & cos(\theta) \end{bmatrix} \tag{3.5}$$

$$R(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi) & sin(\phi) \\ 0 & -sin(\phi) & cos(\phi) \end{bmatrix} \tag{3.6}$$

The rotation matrix for moving from the NED to the body frame is given by

$$R_I^B(\phi, \theta, \psi) = R(\phi)R(\theta)R(\psi) \tag{3.7}$$

Performing the multiplication, from the right to the left,

$$R_I^B(\phi, \theta, \psi) = \begin{bmatrix} cos(\psi)cos(\theta) & cos(\theta)sin(\psi) & -sin(\theta) \\ cos(\psi)sin(\phi)sin(\theta) - cos(\phi)sin(\psi) & cos(\phi)cos(\psi) + sin(\phi)sin(\psi)sin(\theta) & cos(\theta)sin(\phi) \\ sin(\phi)sin(\psi) + cos(\phi)cos(\psi)sin(\theta) & cos(\phi)sin(\psi)sin(\theta) - cos(\psi)sin(\phi) & cos(\phi)cos(\theta) \end{bmatrix} \tag{3.8}$$

The opposite DCM, the matrix that describes the rotation from body to the inertial frame, is described by

$$R_B^I(\phi, \theta, \psi) = R(-\psi)R(-\theta)R(-\phi) \tag{3.9}$$

It is the computed as the transpose of the matrix 3.8 $\left(R_B^I = (R_I^B)^T\right)$, or, in complete notation,

$$R_B^I(\phi, \theta, \psi) = \begin{bmatrix} cos(\psi)cos(\theta) & cos(\psi)sin(\phi)sin(\theta) - cos(\phi)sin(\psi) & sin(\phi)sin(\psi) + cos(\phi)cos(\psi)sin(\theta) \\ cos(\theta)sin(\psi) & cos(\phi)cos(\psi) + sin(\phi)sin(\psi)sin(\theta) & cos(\phi)sin(\psi)sin(\theta) - cos(\psi)sin(\phi) \\ -sin(\theta) & cos(\theta)sin(\phi) & cos(\phi)cos(\theta) \end{bmatrix} \tag{3.10}$$

The order of rotations is a very important property, it is unique, and changing it yields entirely different results.

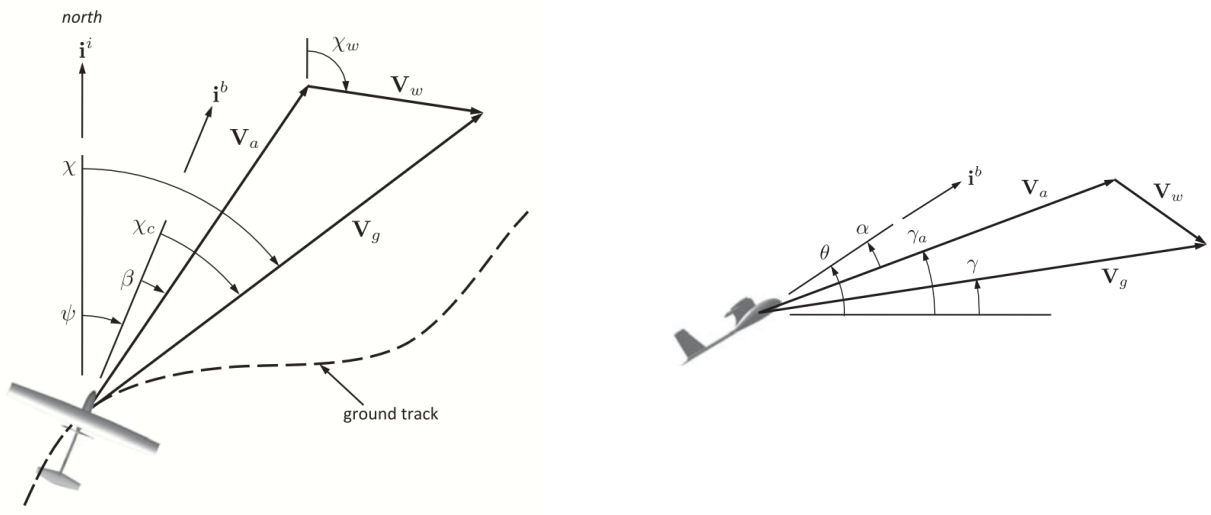
3.3 Ground speed, airspeed and wind speed

When looking at the problem from both reference systems, one can identify three different velocities: The velocity over ground $\overrightarrow{GS}$, velocity of the aircraft with respect to the inertial frame; the airspeed velocity $\overrightarrow{TAS}$, velocity of the airframe with respect to the surrounding air; and finally the wind velocity $\overrightarrow{W}$, velocity of the wind at a given moment.

Together, these three vectors yield an equation called the ”Wind triangle” equation, which simply states that the groundspeed vector is the vector sum of the airspeed vector together with the wind speed vector, that is,

$$\overrightarrow{GS} = \overrightarrow{TAS} + \overrightarrow{W} \tag{3.11}$$

The ”Wind triangle” relation can only be applied if all of its elements are properly rotated to the same reference system. A more detailed picture of this relation is given in the horizontal and vertical planes, in Figure 3.8.



(a) Projection on the horizontal plane.

(b) Projection on the vertical plane.

Figure 3.8: The wind triangle projected in horizontal and vertical plane (Adapted from [39]).

The i^i-axis points in the north direction, in the inertial frame, and i^b-axis along the x-axis of the body frame. Apart from the angles previously described, it is also possible to identify a new angle, firstly introduced here, the *crab angle* χ_c that is defined as the difference between the ground course and the heading:

$$\chi_c = \chi - \psi \tag{3.12}$$

Also, on the vertical projection, it is possible to identify the flight path angle γ, and the air-mass-referenced flight-path angle γ_a, defined when there is a down component of the wind. Each of the velocities presented on equation 3.11, can be decomposed in the inertial frame, in each of its axes components as follows:

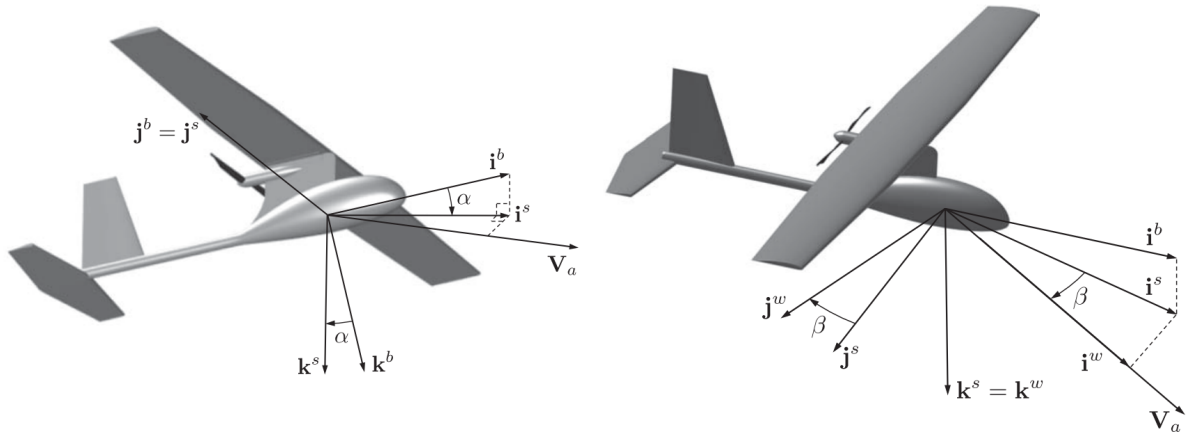
$$\bullet\ \overrightarrow{GS}^I = \begin{bmatrix} V_N \\ V_E \\ V_D \end{bmatrix} \qquad \bullet\ \overrightarrow{W}^I = \begin{bmatrix} W_N \\ W_E \\ W_D \end{bmatrix} \qquad \bullet\ \overrightarrow{TAS}^I = R_B^I(\phi,\theta,\psi) \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{3.13}$$

When working with the airspeed of the aircraft it is also common to work with the angle of attack and sideslip, depicted in Figure 3.9 and defined as:

- Angle of attack: $\alpha = tan^{-1}\left(\frac{w}{u}\right)$;
- Angle of sideslip: $\beta = sin^{-1}\left(\frac{v}{|\overrightarrow{TAS}|}\right)$.

For $TAS = \sqrt{u^2 + v^2 + w^2}$, these angles can be used to describe the components of the airspeed as represented in equation 3.14:

$$V_a^B = TAS \times \begin{bmatrix} cos(\alpha)cos(\beta) \\ sin(\beta) \\ sin(\alpha)cos(\beta) \end{bmatrix} \tag{3.14}$$



(a) Angle of attack α.

(b) Angle of sideslip β.

Figure 3.9: The wind frame (adapted from [39]).

It should be highlighted that in Figure 3.9, the i^s-axis points along the projection of the airspeed vector onto the i^b-k^b plane of the body frame, the j^s-axis is identical to the j^b-axis of the body frame, and the k^s-axis is constructed to make a right-handed coordinate system. Also, the i^w-axis points along the airspeed vector.

3.4 Least Mean Squares

The method of Least Mean Squares (LMS) is a procedure to determine the parameters that best fit a certain model to a cluster of data, using simple calculus and algebra.

Assuming we have a set of observations $x(n) = x_1, ...x_N$ and we want to predict an output $y(n) = y_1, ..., y_N$, it is safely assumed that the variable $y(n)$ can be approximated by a function that depends on the observations made: $\hat{y}(n) = f(x(n), \theta)$, where $\theta = [\theta_1, ..., \theta_k]$ is a vector of k parameters to be estimated. To state that $y(n)$ can be approximated to $\hat{y}(n)$ is to state that they differ from an error, or residual, $e(n)$, that is,

$$y(n) = \hat{y}(n) + e(n) \tag{3.15}$$

To evaluate the efficiency of the estimation, the Least Squares method defines an energy criterion, defined as the sum of the squared errors:

$$E(\theta) = \sum_{n=1}^{N} \Big(y(n) - \hat{y}(n)\Big)^2 \tag{3.16}$$

which can be written as:

$$E(\theta) = \sum_{n=1}^{N} \Big(y(n) - f(x(n), \theta)\Big)^2 \tag{3.17}$$

The values θ that best fit the model are the ones that minimize the Least Squares energy:

$$\hat{\theta} = min_{\theta} E(\theta) \tag{3.18}$$

When the model has a linear nature: $f(x, \theta) = \gamma_1(x)\theta_1 + ... + \gamma_p(x)\theta_p$, the problem can be formulated and solved using matrix notation, as follows:

$$y(n) = H(n)\theta \tag{3.19}$$

- $y(n) = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$
- $H(n) = \begin{bmatrix} \gamma_1(x_1) & \cdots & \gamma_k(x_1) \\ \vdots & \vdots & \vdots \\ \gamma_1(x_N) & \cdots & \gamma_k(x_N) \end{bmatrix}$
- $\theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_k \end{bmatrix}$ (3.20)

If the matrix H has full rank, i.e, if there is enough information in matrix H, it is possible to calculate the parameters that best fit the model and minimize the Least Squares, by first multiplying each side of the equation by the the transpose of H:

$$(H^T H)\hat{\theta} = H^T y \tag{3.21}$$

which, by inverting the matrix $H^T H$, leads to:

$$\hat{\theta} = (H^T H)^{-1} H^T y \tag{3.22}$$

3.5 Condition number

In some cases, the solution to a system of linear equations $Mx = b$ may be very sensitive to small changes in either the matrix M, or the vector b — a slight change in either can result in a significant change in the solution x.

The coefficient matrix M is called ill-conditioned whenever a small change in its values results in a large change in the solution. The *condition number* of a matrix measures how sensitive the answer is to

perturbations in the input data and to round off errors, made during the solution process, i.e, a condition number, is used to measure the degree of ill-conditioning of a matrix.

The condition number is here exposed by two definitions:

Definition 1: For a square non-singular matrix A, the condition number can be defined as:

$$k(A) = ||A||||A^{-1}|| \tag{3.23}$$

where the norm ||.|| above could be any of the norms defined for matrices. Using the usual Euclidean norm on vectors and the associated matrix norm, then the condition number is equal to the ratio of the largest singular value of matrix A to the smallest.

Definition 2: When dealing with non square matrices, the condition number can be defined as

$$k(A) = ||A||||A^{+}|| \tag{3.24}$$

where A^+ is the pseudo inverse of the matrix A. Note that for a square non singular matrix: A^+ = A^{-1}, implying that the second definition is the generalization of first to obtain the condition number of any matrix.

In a general way of speaking, larger values of k means that, changes in matrix A can lead to much larger changes in the solution. Also, as a ”rule of thumb”, if the condition number can be defined as $k(A) = 10^w$, then we may lose up to w digits of accuracy on top of what would be lost to the numerical method, due to loss of precision from arithmetic methods. The best case scenario happens for $w = 0$, which leads to a condition number of $k(A) = 10^0 = 1$. [40]

It should be highlighted that the condition number does not give the exact value of the maximum inaccuracy that may occur in the algorithm, and it generally just bounds it with an estimate.

3.6 Theoretical kinematic models

3.6.1 Model 1

The work of C.Hurter et al. [3] was used as a foundation to this first model. Their idea was to use radar measurements of position and velocity over time, and analyse the trajectory of an aircraft as one way to estimate the wind components. Analysing the direction and ground speed of one or more aircraft relative to the ground, they realized that these two had a sinusoidal shape relation, which could be explained by the influence of the wind on the aircraft.

Approximating the problem to an horizontal leveled situation, and crossing measurements of several aircraft at the same flight level, the authors successfully estimated the wind components on the North and East directions, for separate flight levels.

The airspeed of an aircraft can easily be tracked from the ”wind triangle” relation, that states that the

ground speed vector ($\overrightarrow{GS}$) at each point is the sum of its airspeed ($\overrightarrow{TAS}$) and wind ($\overrightarrow{W}$) vectors, that is,

$$\overrightarrow{GS} = \overrightarrow{TAS} + \overrightarrow{W} \tag{3.25}$$

This relation is captured in Figure 3.8. Assuming a leveled flight, and taking into account the angles of the ground course angle (χ), heading (ψ), and wind direction (χ_w), it is also possible to derive the relation,

$$GS = TAS \times cos(\psi - \chi) + W \times cos(\chi_w - \chi) \tag{3.26}$$

Equation 3.26 is the starting point of the Huel et al. [3] theoretical model, and consequently of the present work as well. In their work, the authors aimed to deduce the wind from several measurements of the ground speed and ground course through the use of radars. Since they didn't have measurements from the aircraft's heading, they assumed in a practical way that the drift angle($\psi - \chi$) was very small when flying at high speeds.

In our case, equation 3.26 is the core of our model, since we are provided of observations from a GPS and an AHRS, leaving the drift angle approximation out of the equation.

3.6.2 Model 2

Based on previous work, C.Hurter et al. [3] used a reliable approximation in a purely horizontal situation. However, the question remains whether its possible to find a solution in a broader scenario, or in other words, a solution that encompasses the vertical movement of the aircraft. The model here presented was later explored for this purpose, to find how far we could extend the theory of C. Hurter et al. [3], to obtain more precision on the results.

The main challenge is how to describe the airspeed in its three components without previous knowledge of any of them. William Premerlani [4] made an interesting point when analysing the general behaviour of the airspeed during a flight, in a presentation he gave to the air drone community. While the overall theory didn't get to be duly presented in an official paper, its statements were bold enough to be "dissected" and properly analysed.

The general set-up can be presented as follows: for a flight at a relatively high speed, it can be said that the airspeed will be mainly directed on the longitudinal axis of the aircraft; or in other words, the velocity along the x-axis will be much larger than along the other two.

This latest statement is captured in the equation below:

$$\overrightarrow{TAS}^B \approx |\overrightarrow{TAS}| \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{3.27}$$

In most cases, the sideslip angle can be assumed to be close to zero: $cos(\beta) \approx 1$ and $sin(\beta) \approx 0$. In the case of an aircraft controlled by an autopilot, the controller tends to keep the acceleration around the y-axis to be zero, which culminates in a sideslip angle very close to zero. With this assumption, equation 3.14 reduces to

$$\overrightarrow{TAS}^B \approx |\overrightarrow{TAS}| \times \begin{bmatrix} cos(\alpha) \\ 0 \\ sin(\alpha) \end{bmatrix} \tag{3.28}$$

The angle of attack can also be said to be small and, and considering this, equation 3.28 approximates to equation 3.27. Equation 3.27 is an approximation, and when subjected to wind gusts or any unexpected behaviour, it should not hold. Still, if these situations are properly dealt with, this same expression can be a safe assumption at relatively high speeds, except of course during take-off and landing stages.

One should be aware that, in the best case scenario, if the angle of attack and sideslip were to be known, equation 3.14 could be used to describe the behaviour of the aircraft on its three axis.

Expression 3.27 describes the approximation for the airspeed in the body frame, and to be able to work with it in NED frame, it should be multiplied by the appropriate rotation $R_B^I(\psi, \theta, \phi)$, yielding

$$\overrightarrow{TAS}^I \approx R_B^I(\phi, \theta, \psi) \times |\overrightarrow{TAS}|^B. \tag{3.29}$$

$$\overrightarrow{TAS}^I \approx |\overrightarrow{TAS}| \times \begin{bmatrix} cos(\psi)cos(\theta) \\ cos(\theta)sin(\psi) \\ -sin(\theta) \end{bmatrix} \tag{3.30}$$

Vector 3.30 suggests that the airspeed in the NED frame is approximately the module of the airspeed times the first row of the DCM that describes the rotation from the body to the inertial frame.

The theoretical model for this situation can thus be described as

$$\begin{bmatrix} V_N \\ V_E \\ V_D \end{bmatrix} = \begin{bmatrix} TAS \times cos(\psi)cos(\theta) + W_N \\ TAS \times cos(\theta)sin(\psi) + W_E \\ -TAS \times sin(\theta) + W_D \end{bmatrix} \tag{3.31}$$

The latest vector 3.31 represents an approximation model that describes the velocity of the aircraft in a NED frame. The theory behind it was discussed at length with TEKEVER and found to be valid, as long as it would comprise certain restrictions, as explained later in chapter 4.

Chapter 4

Implementation

This chapter starts off with the validation of the assumption of the wind remaining steady within a given time frame, followed by the numerical implementation and validation of the theoretical kinematic models described in section 3. Finally, the chapter closes up with a concisely comparison between the two alternatives of implementation, over three different scenarios.

4.1 Validation of the Wind criteria

The wind velocity is a third dimensional vector, but it is usually defined by two parameters: velocity magnitude and direction, in which the direction is contained in the horizontal plane and it points to where the wind is coming from (for example, a wind coming from the North to the South is defined as having a direction of 0°).

For the same flight level, the previous definition of wind vector can be safely assumed (basically defining the vertical component of the wind close to zero), but the wind varies in altitude, due to air density and pressure area variations.

As will be explained in the next topics, it is a requirement to assume the wind to remain constant within a certain interval, and so the first step is to study the behaviour of this parameter and confirm its low entropy in a given time window, or in other words, a low ”disorder” of the wind data. Experienced professionals can confirm that for small changes in time, the wind won't change much, but there isn't any assurance of how long, ”small” can be considered - for 1 second apart measurements it is reasonably valid to assert that the change will be significantly small, but in this work it is a requirement to operate in larger time windows. To validate how large this window can be, it was gathered and duly analysed past wind data measurements from a trusted meteorological station and from real flight estimation.

- Meteorological Stations

The Department of Atmospheric Sciences in the University of Utah provides online past data information from meteorological stations across the United States, available in the repository *MesoWest Data* [41]. Different stations supply measurements at different sampling rates, and in order to have as

much information as possible, Salt Lake station was chosen as and object of study, since it was found to present the minimum sampling rate of 5 minutes.

Daily information from 2018 was gathered and later processed and analysed as a way of quantifying the change of the wind in time. The information provided by the *MesoWest Data* repository is of upmost importance, but it should be known that it lacks some daily hours information, which were treated as outliers and taken out of the calculated statistics.

The North and East components of the wind are captured in Figures 4.1 and 4.2, respectively, for 1st of January of 2018, in the time window $t \in [18:55; 22:55]$. This window was chosen due to its wide range of information without outliers, and the station captured them for an elevation of 4226 ft, or 1288.08 m.

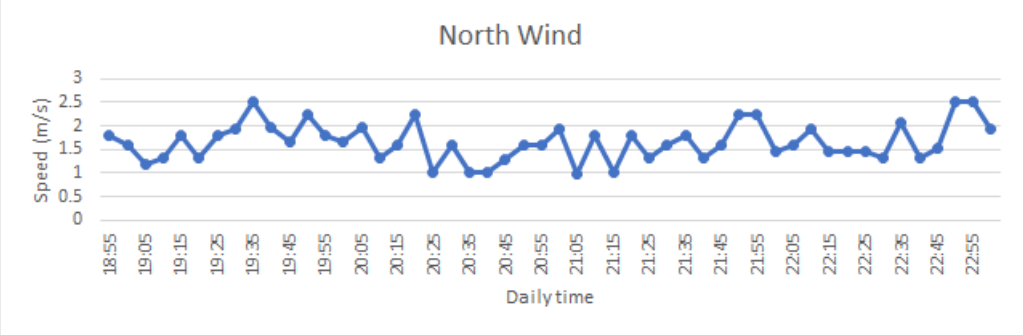


Figure 4.1: North wind, 1st January 2018.

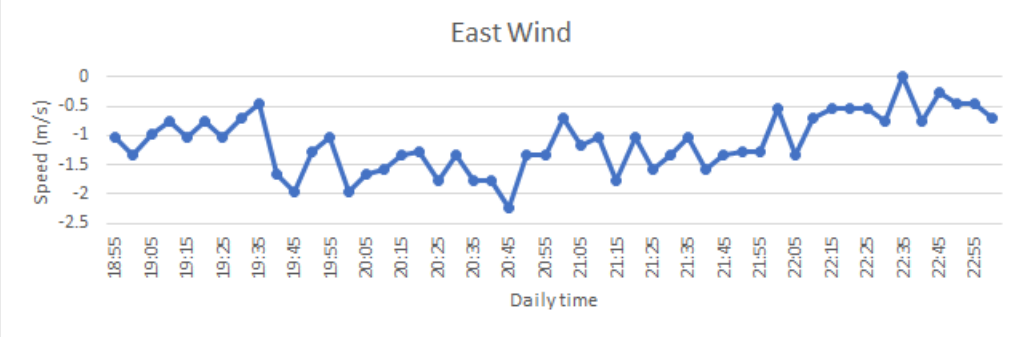


Figure 4.2: East wind, 1st January 2018.

Both figures 4.1 and 4.2 capture 4 hours of wind data, described in the North-East reference frame. At first sight, the results are visually promising - the two graphics appear to present wind speeds that vary randomly around a mean value with not much disparity. To better analyse this data, the information was processed and the mean-value, standard deviation, minimum and maximum values were taken, and are presented in table 4.1.

First inspecting the North component of the wind, the maximum and minimum values are significant to inform the total speed range of $1.541m/s$, which inform us that in 4 hours of data, the wind is contained in a rather small volume area of just $1.541m/s$. Also, with its mean value of $1.6568m/s$, this component presents a standard deviation value of only $0.3976m/s$.

Concerning the East component of the wind, and following the same reasoning as for the North,

Wind components	Mean (m/s)	Standard deviation (m/s)	Minimum (m/s)	Maximum (m/s)
W_N	1.6568	0.3976	0.9899	2.5309
W_E	-1.1212	0.4964	-2.2257	-5.0478$\times 10^{-16}$

Table 4.1: Mean, standard deviation, minimum and maximum values on wind speed, in m/s.

the maximum and minimum values present a total range of $2.2257m/s$ in which the wind data can be contained, and with its mean value in $-1.1212m/s$, it presents a standard deviation of $0.4964m/s$.

An important aspect for both components is the total area in which the data is comprised - $1.541m/s$ and $2.2257m/s$ are fairly small when looking for an amount of 4-hour data. Also, and more important, both standard deviations are bellow $\sigma = 0.5$, indicating that the dispersion of the values around its mean value is also relatively small.

The results obtained on the first day of January, 2018, follow the reasoning that the wind does change slowly in time, to the point of in a 4-hour data, remaining closely the same.

Nevertheless, this is data from a single day and to inspect this phenomenon more closely, it was decided to evaluate how the wind changes from each of its 5min measurements in a wider sample. To this end, the difference between measurements on the 15th day of each month, in the year of 2018, was calculated. Taking the absolute value of this difference, it was possible to calculate its mean and standard deviation values. Table 4.2 presents all these variables.

Month	$\mu(\lvert\Delta W_N\rvert)$	$\sigma(\lvert\Delta W_N\rvert)$	$\mu(\lvert\Delta W_E\rvert)$	$\sigma(\lvert\Delta W_E\rvert)$
January	0.3234	0.3164	0.3515	0.3514
February	0.7845	0.7468	0.7347	0.7803
March	0.6567	0.7448	0.6521	0.6729
April	0.5853	0.5294	0.5248	0.5787
May	0.6447	0.5844	0.5968	0.5938
June	0.6995	0.6373	0.6995	0.7728
July	0.6096	0.6903	0.5595	0.6072
August	0.5615	0.4966	0.6547	0.6256
September	0.7543	0.8603	0.7214	0.8654
October	0.4908	0.5182	0.5605	0.5580
November	0.2822	0.2643	0.3009	0.2989
December	0.4941	0.4844	0.5238	0.4596

Table 4.2: Wind information for the 15th day of each month, 2018, in m/s.

Table 4.2 shows broad data information from the change of the wind in time, specifically its mean and standard deviation, and this information is shown in figures 4.3 and 4.4.

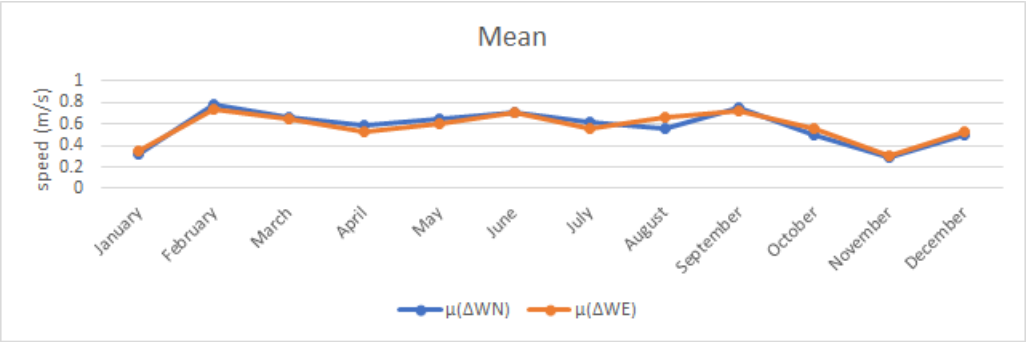


Figure 4.3: Mean of the variation of the wind in a 5min sample rate of the 15th day of each month, 2018.

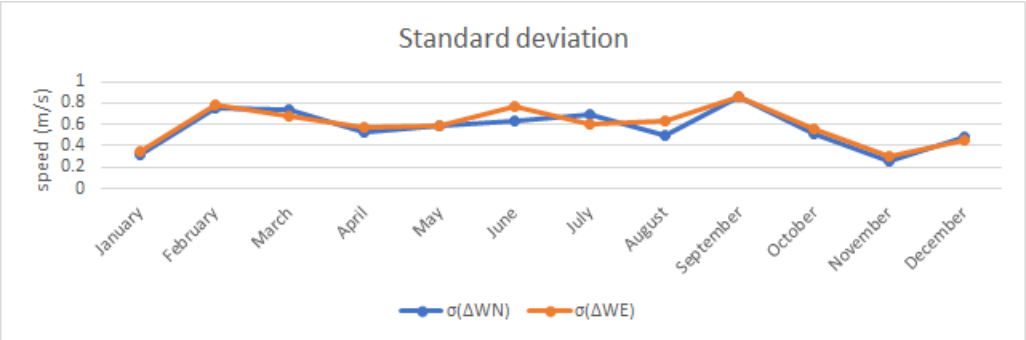


Figure 4.4: Standard deviation of the variation of the wind in a 5min sample rate of the 15th day of each month, 2018.

Figure 4.3 provides useful information about how the wind changes in a 5min period. With the maximum value of $0.7845m/s$ for the North component and $0.7347m/s$ for the East one, both in February, the two wind components present an average change in the order of $0.6m/s$, which provides good statistics for the 5min interval.

The standard deviation in figure 4.4, on another hand, has its peak value in September for the two components, reaching $0.8603m/s$ for the North and $0.8654m/s$ for the East, and an average value around $0.6m/s$ for both as well.

The statistics studied for the 1st of January, 2018, along with the ones obtained for the 15th day of each month of the same year conclude that in a 5min period, the wind variation is relatively small with a low dispersion of values, making it possible to consider the wind to be approximately constant in the given period.

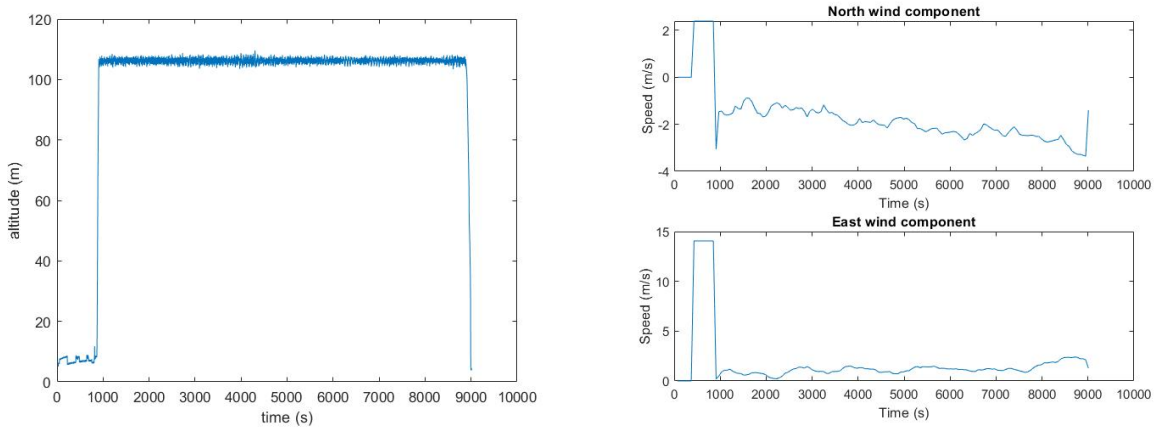
- Real flights

The information provided by the *MesoWest Data* repository was significantly important when studying the behaviour of the wind for a 5min sample rate which, eventually, was found to be large enough to be used in the system.

An important aspect to be reinforced here is that the wind is a stochastic process. Even tough tests can be done for larger time windows and present satisfying results, the wind behaviour is most unpredictable, which compels the user to make use of the smaller time window possible.

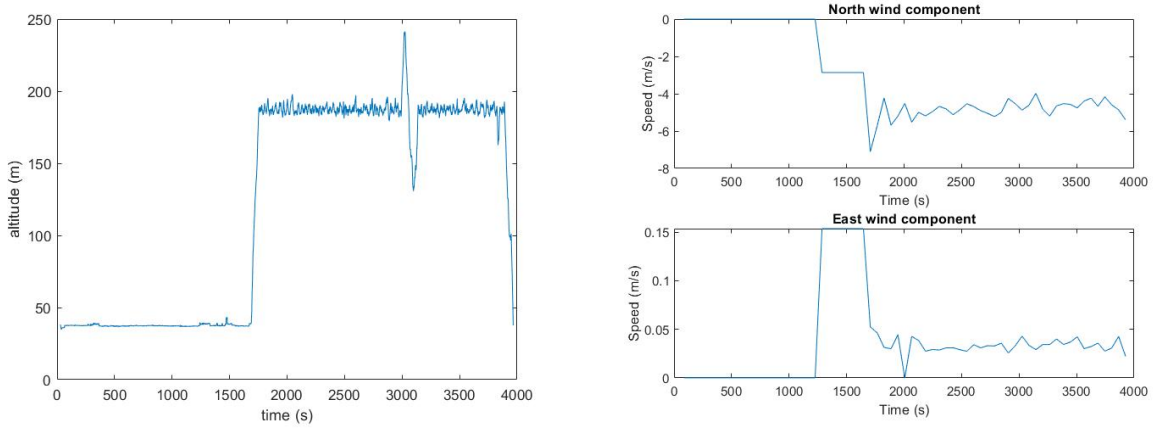
The use of a larger window doesn't invalidate the obtaining of valid results, but the chances of getting precise results lowers with its increasing value. With this in thought, it was found relevant to define a period for the maximum time window allowed, and real flight data was used as one way of reinforcing the results given by the *MesoWest Data* repository and analyse other ranges of time windows.

Three flights were provided by TEKEVER that enabled the analysis of the wind on a 60s sample rate and later, on chapter 5, the algorithm in real flight testing. The behaviour of the altitude and wind of the same flights is captured in figures 4.5, 4.6 and 4.7. More information relating their trajectory, attitude and velocities can be observed in Annexes A, B and C.



(a) Altitude. (b) Wind components.

Figure 4.5: Evolution of altitude and wind for Flight 01.

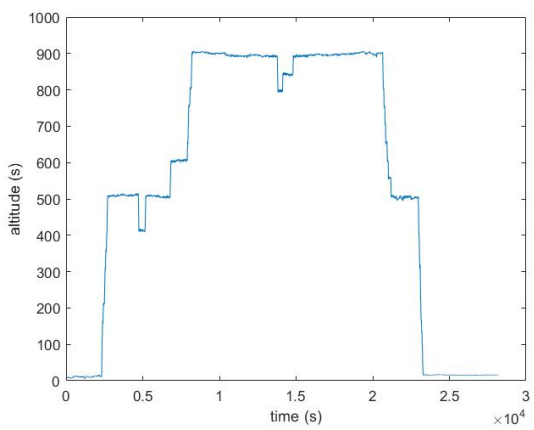


(a) Altitude. (b) Wind components.

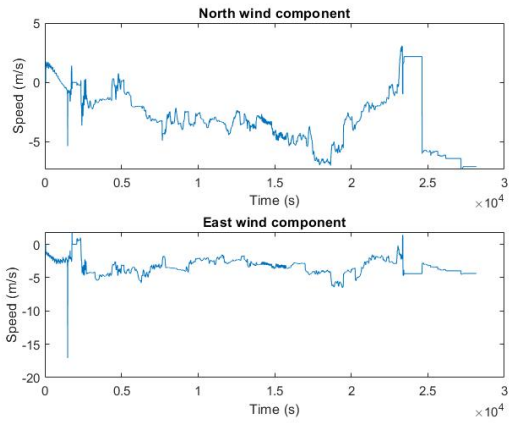
Figure 4.6: Evolution of altitude and wind for Flight 02.

It should be noted that all the wind data here provided is the result of an estimation obtained from an Extended Kalman Filter. Before take-off, there isn't enough information to estimate the wind as the aircraft is immobile in the ground, which explains the errors at the start of each flight. Once the aircraft is airborne ($t \approx 1700s$, for example in Flight 02) the estimates start converging to their real value.

All three flights are of endurance nature, the third one being the longest one, and in order to properly analyse the wind components of the wind, it was first necessary to eliminate the intervals in which the



(a) Altitude.



(b) Wind components.

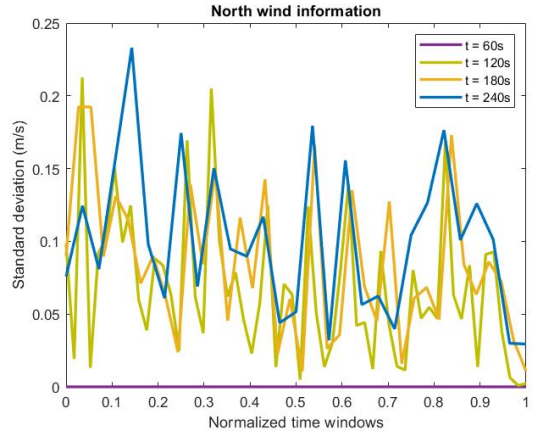
Figure 4.7: Evolution of altitude and wind for Flight 03.

aircraft wasn't in a leveled flight, since, as said before, this study is only valid for the same altitude for which the data was taken.

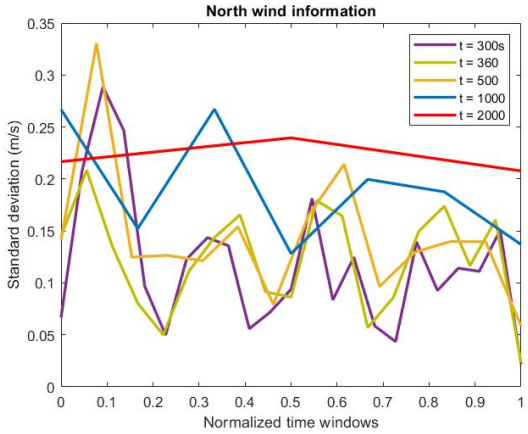
The main guideline to study the behaviour of the wind for different time windows was to analyse how disperse the values were for each of it. One way to accomplish this was to "slice" each flight in several equal chunks.

For time windows in the range of $t_{window} \in [60; 120; 180; 240; 300; 360; 500; 1000; 2000]$, the standard deviation was calculated over each chunk window, as a way to quantify the dispersion of wind values. A problem that aroused was the fact of this process resulting in many windows to analyse. As an example, if taking a total flight of $6000s$, using a time window on the range of $t_{window} = 3000s$ would result in two chunks of data and therefore two standard deviations. On the other hand, if the time window is much smaller, for instance of $60s$, it would result in 100 chunks of data with different standard deviation each.

To circumvent the problem and better analyse the behaviour of the wind, each of the obtained standard deviations were put in a graphic, with its horizontal axis representing the normalized number of time windows and the vertical one, the standard deviation of the wind in m/s. Figures 4.8 and 4.9 depict this



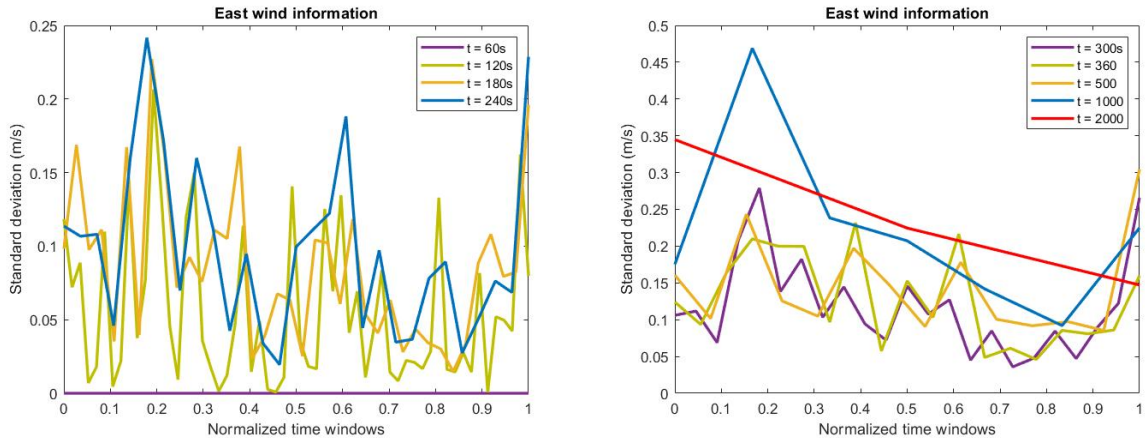
(a) t ∈ {60; 120; 180; 240}.



(b) t ∈ {300; 360; 500; 1000; 2000}.

Figure 4.8: Evolution of standard deviation with increasing time windows (North component).

normalization for the first flight, in the case of the North and East components of the wind, respectively. The same information for the second and third flight can be observed in appendix B and C, respectively.



(a) t ∈ {60; 120; 180; 240}.

(b) t ∈ {300; 360; 500; 1000; 2000}.

Figure 4.9: Evolution of standard deviation with increasing time windows (East component).

Figures 4.8 and 4.9 present a method to visually understand how the wind is dispersed along a time window of increasing size. Before heading to any conclusions, its noticeable that the standard deviation for both components when $t_{window} = 60s$ is zero, and this happens due to the sampling rate of the wind measurements to be also $60s$. However, this doesn't happens with the third flight, since a different aircraft was used in that situation, with a sampling rate of $5s$.

In a general way of speaking, one can visually understand that with the increasing value of the time window, also the standard deviation of its values increases, reaching its peak value at $\sigma_{max} = 0.45m/s$ for a $t_{window} = 1000s$. It is also visually understood that until $t_{window} = 500s$, both components of the wind are contained in a standard deviation volume of around $0.25m/s$. The bigger difference appears in the East component for a $t_{window} = 1000s$ and $t_{window} = 2000s$, where the standard deviation reaches $0.45m/s$ and $0.35m/s$, respectively.

In an attempt to compare the results from all three flights, table 4.3 and 4.4 encompasses the mean values of each of the standard deviations of all three flights.

The second flight was the only one in which it was not possible to obtain results for a time window of $t_{window} = 2000s$ due to the fact of the aircraft not remaining in a leveled flight that long.

Time window	60	120	180	240	300	360	500	1000	2000
flight 01	0	0.06763	0.08453	0.1014	0.1175	0.1225	0.1444	0.1912	0.2213
flight 02	0	0.2048	0.2422	0.2551	0.2489	0.2551	0.2939	0.325	———
flight 03	0.07825	0.1344	0.1646	0.2437	0.2311	0.3233	0.393	0.4045	0.4339

Table 4.3: Mean values of the standard deviations of W_N for different time windows (in m/s).

Time window	60	120	180	240	300	360	500	1000	2000
flight 01	0	0.05585	0.08508	0.09769	0.1179	0.1272	0.1449	0.2212	0.2388
flight 02	0	0.1511	0.1866	0.2005	0.1982	0.1891	0.1918	0.2146	———
flight 03	0.0692	0.1178	0.1342	0.1732	0.1869	0.1994	0.2764	0.3201	0.5409

Table 4.4: Mean values of the standard deviations of W_E for different time windows (in m/s).

Both tables 4.3 and 4.4 are important to the fact that they retain information from all flights. It is possible to observe an increasing evolution of the dispersion of the values with the increasing size of the time window for all three flights, aside one or two cases, as it happens for the second flight from a change of $180s$ to $240s$, and then to 300, in the East component of the wind, or for the third flight in the same intervals.

Altogether, it can be said that the wind does changes slowly in time. Using the standard deviation to quantify the dispersion of values in a given interval, it was possible to conclude that the smaller the time window, the smaller will be its variation. In this study, while $2000s$ does confirm to have a small standard deviation (maximum of $0.5409m/s$), it is advisable not to exceed this value and stick to a lower value, if possible, as a way to maintain the maximum possible precision. In the end, it depends strongly on the accuracy and reliability required for the results.

In this work, $t = 360s$ was defined as the maximum interval at which the wind could be considered to remain constant. This time was chosen by testing the algorithm in real flights, and inferring it to be a time frame large enough.

4.2 Algorithm and restrictions

In order to integrate each of the kinematic models in a LMS algorithm, it was first necessary to rewrite each of them in such a way as to bring them to the general form

$$H\theta = y \tag{4.1}$$

where H is the system model, θ the parameters to be estimated, and finally y the observation matrix. Equation 4.1 will be revised in more detail later in the chapter for each of the kinematic models. After obtaining a linear nature, the algorithm is in charge of calculating the parameters that best fit the given model, using the algebraic manipulation

$$\theta = (H^T H)^{-1} H^T y \tag{4.2}$$

Expression 4.2 represents the numerical calculation of the LMS parameters that best fit the given model and it is valid as long as the restrictions inherent to the model approximations are validated.

4.2.1 Restrictions

For both models, the flight data must meet some requirements in order to be validated: The plane should present an approximate horizontal trajectory; the matrix H should be well conditioned; the acquisition of data should be collected as fast as possible and the mean square error should be as low as possible.

The two kinematic models are restricted to an approximate leveled flight, and to quantify the dispersion of altitude measurements in the cluster of data, the first tuning parameter $\boldsymbol{\Delta h_{max}}$ appears as limited by a maximum altitude rate allowed: $\boldsymbol{\Delta h_{max} < \gamma_h}$.

Another restriction is the interval of time for the acquisition of data, and a major constrain for this is the wind. As stated before, the wind is a stochastic process, and if not accounting for wind gusts, the only certainty one can have is that for small variations of time, the wind speed should be approximately the same. However, to use the Least Mean Squares, there is required a trajectory entropy. That means taking measurements in the order of several seconds, depending strongly on how fast the aircraft can change its attitude. In previous work, C. Hurter et al. [3], used a cluster of data in the order of 1 to 2 hours, and, empirically, experienced engineers back up the fact that the wind varies slowly for the same flight level. Nevertheless, it should be clear that the longer the time order of the cluster, the less accurate will be the results, and as one way of consolidating this statement, the validation and specification of a maximum threshold of time γ_t, was done in section 4.1.

The next restriction is defined to quantify how well-conditioned is the matrix H, i.e, to quantify if there is relevant information in the matrix H to achieve a valid result, and it is evaluated through the use of a limited threshold: $\boldsymbol{k(H) < \gamma_k}$, where $\gamma_k = 1$ presents the scenario for which the matrix is best conditioned.

Lastly, the mean square error is taken out using the estimated parameters to appraise the difference between these and the real values. This parameter is also evaluated through the use of a maximum threshold allowed: $\boldsymbol{MSE < \gamma_{MSE}}$.

The selection of each of the thresholds varies for both kinematic models. As such, this process was done empirically, and will be detailed later in the chapter, using simulated flights, and in chapter 5, using real flight data.

4.2.2 Algorithm

The implementation of both models on the LMS presents very specific restrictions. For this reason, it was decided to be used as a back up system to increase the robustness of the standard air data system, and identify if the pitot tube is working as expected.

The base algorithm first evaluates the past neighbouring points for a given point P in time and space, taking into account a time window $\boldsymbol{t_1}$, and if all restrictions, previously described in 4.2.1, are achieved, it is allowed to carry on the estimation. If, during an horizontal flight maneuver, there isn't enough entropy in the measurements to obtain a good estimation, the algorithm rewinds to the start and extends the neighbouring points to a bigger time window $\boldsymbol{t_2}$. This procedure is repeated in loop until the algorithm achieves an estimation, up to a maximum time window $\boldsymbol{t_{max}}$.

If the estimation is not possible at the maximum time window $\boldsymbol{t_{max}}$, for example in the case when the plane starts describing a straight trajectory, the auto-pilot can continue to compare its values obtained from the standard Air Data System with the latest wind estimations as a way to keep the robustness required. This, however, should only be valid for a certain amount of time $\boldsymbol{time_{hold}}$, for which, if outdated, should signal a warning to let the auto-pilot know that the last estimations are no longer valid and it should maneuver the plane to update them when possible.

The back up this system provides whats can be seen as ”virtual robustness”. While continually estimating the parameters, there is a valid way of comparing them with the ones given by the standard ADS, and even in cases where the estimation is not possible, for example in a straight trajectory, as long as the point in time and space of comparison is contained in the $\boldsymbol{time_{hold}}$.

Using the last wind estimation as invariable for the next moments, even if the wind speed does change more than expected when comparing with the standard ADS results, the values will still remain in a neighbouring area and are enough to state that the pitot tube is reading correct values. However, if the ADS ends up giving enormous errors when comparing with the last wind estimations, it is clear that the pressure sensor is obstructed and at this point, procedures according to each of the users necessities should be done, for example an emergency landing could be deployed.

Summarizing, there are five important steps to follow:

1. Filter the neighbouring information for a point P in time and space;
2. Examine if the retrieved data describes approximately an horizontal flight;
3. Quantify the entropy in matrix H through its condition number, and calculate the minimum square error;
4. In case the estimation is not possible, increase the time window of Δt up to a maximum of $t = t_{max}$ and repeat steps 1, 2 and 3;
5. If the estimation is still not possible in t_{max}, check if the estimations are outdated and decide whether to signal a warning or move forward to the next point in time and space P'.

The information just described can be visualized in the flowchart described in figure 4.10, and it represents the core of the algorithm. The use of it can be inserted in a wide variety of systems, depending on the needs of the user.

Most autopilots can fly an aircraft without airspeed information quite ”safely” by using some form of throttle model, though envelope protection may be compromised. In this work, it is suggested for the framework to signal a warning, and, in case of confirming that the pitot tube is faulty, to deploy an emergency landing procedure, but it is equally valid to deploy a throttle model based flight. At this point, it is really up to the user to decide how to incorporate the algorithm in its needs. The verification and validation of the wind criteria was thoroughly studied in section 4.1, and a time window of $t_{window} = 360s$ has been chosen to be the maximum time frame to be considered in the algorithm. For this reason, the variable t_{hold}, was also considered to be $360s$.

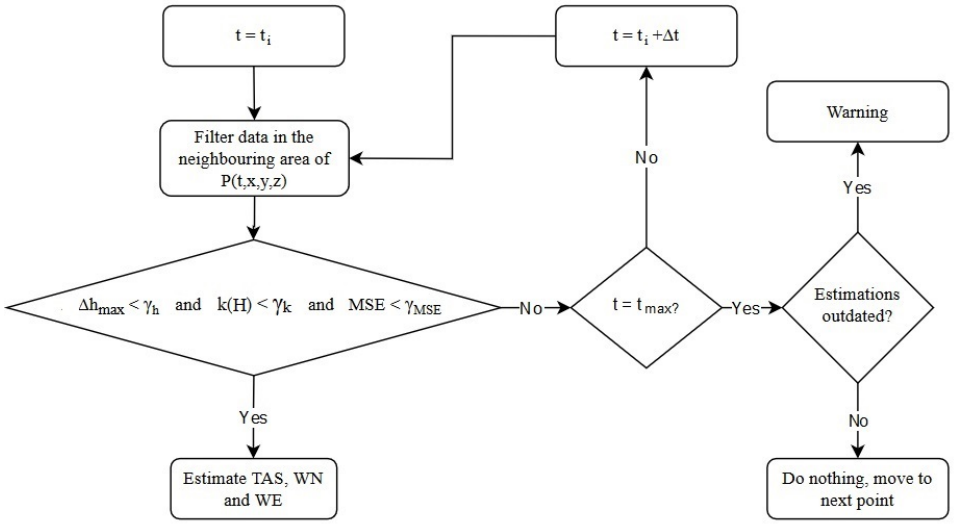


Figure 4.10: Flowchart of the estimation algorithm.

On the validation of both models, there are several key aspects to be studied:

- Impact of the attitude angles;
- Impact of the rate of climb.
- Convergence of the algorithm when subjected to a sudden wind speed change or a slow-varying altitude;

The last item, concerning the convergence of the algorithm, is directly related to the first two, as will be explained later in the validation of the methods. These are of special interest due to presenting limitations to both the approximations considered in the kinematic models.

Of the several tuning parameters involved, one is of special interest. The introduction of the altitude rate is important to evaluate whether it is possible to obtain satisfying estimates when the aircraft is continuously changing altitude, preferable at a low pace. This would allow the aircraft to be less dependent of a strict horizontal flight.

4.3 Numerical Kinematic Model 1

Apart from the reasonable approximation to the kinematic modeling of the problem, described in equation 3.26, C. Hurter et al. [3] contributed by proposing a solution to the problem that made use of the Least Mean Squares algorithm applied to a cluster of data.

Differentiating on the sensor data used, and excluding the assumption of the drift angle to be zero, the implementation of this first model is based on the applicability of placing it in a LMS algorithm, under specific conditions, as one way to find the parameters that best fit a given data. Even though the same theory is applied, the goal of C. Hurter et al. [3] is very different from the one here presented, as will be quickly understood in the following paragraphs that explain how the events unfold.

Prior to heading directly to the implementation, it is important to evaluate if the model can be cast in the canonical form of equation 4.1, defining the observations and parameters to be estimated. Considering an airspace volume over a time interval, and assuming that the airspeed (ensured by the autopilot most of the time) and wind, remain constant within this volume, equation 3.26 can be applied to a given array of size data of N elements, and manipulating it algebraically results in

$$\underbrace{\begin{bmatrix} cos(\psi_1 - \chi_1) & cos(\chi_1) & sin(\chi_1) \\ \vdots & \vdots & \vdots \\ cos(\psi_N - \chi_N) & cos(\chi_N) & sin(\chi_N) \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} TAS \\ W_N \\ W_E \end{bmatrix}}_{\theta} = \underbrace{\begin{bmatrix} SpeedEN_1 \\ \vdots \\ SpeedEN_N \end{bmatrix}}_{y} \tag{4.3}$$

Expression 4.3 contains the indication of each matrix when matching the whole expression to the LMS basic relation 4.1. It is simple to distinguish that the parameters to estimate are exactly the true airspeed of the aircraft TAS, and both longitudinal components of the wind, W_N and W_E, all measured in the NED frame, while the observation matrix is composed by the norm of the longitudinal ground speed, $SpeedEN$. The matrix H, describing the system, is composed by measurements of ground course angle, χ, and the heading angle, ψ.

The GPS is in charge of providing measurements of the ground speed and ground course, while the AHRS of the heading angle. It should be highlighted that all these measurements are previously filtered states.

4.3.1 Verification and Validation

The validation of the first kinematic model incorporated in the algorithm described in subsection 4.2.2 began with the testing of it on a controlled environment. The simulator *X-Plane 10* was used to this end. Initially, the glider aircraft **ASK 21** was manually controlled in the simulator, but it was found rather difficult to maneuver the plane the same way as required. Hence, TEKEVER provided simulated flights of their drones encompassing several situations in the simulator just referred.

To validation terms, two autopilot flights, simulated in the *X-plane10* environment, are here presented, and information regarding their flight, namely attitude angles, trajectory and positioning in the NED frame, GPS velocity, and wind velocities are presented in annex D and E.

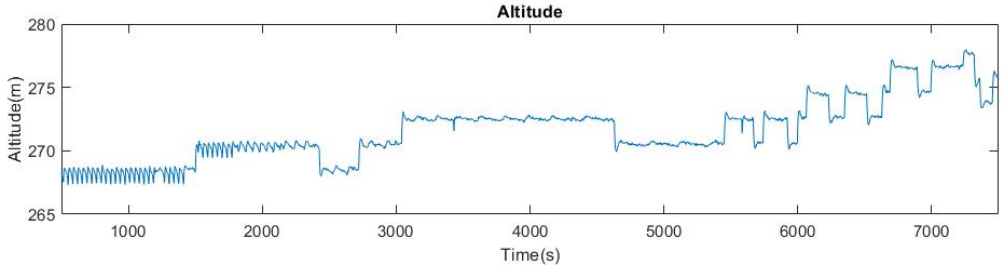
The following study starts off with three main topics: the validation of the algorithm, the analysis of the impact of the yaw rate in the condition number, and the analysis of the behaviour of the algorithm when subjected to an altitude maneuver.

Validation in standard conditions along with sudden wind speed change

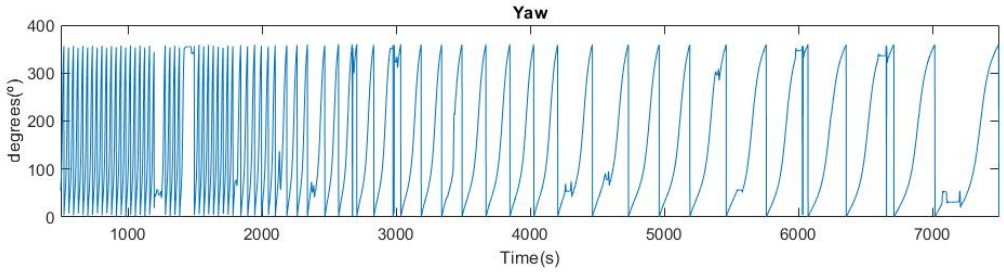
Autopilot Flight 01 is composed of various intervals in which the aircraft is kept at the same leveled altitude. The trajectory of the aircraft, shown in annex D, displays a flight with several loiter stages, for different radius and locations.

The choosing of this flight for the validation of the algorithm was based on two particularities: First, the variation in altitude is relatively small. Secondly, at about $t = 5545s$, for the same altitude, the wind was intentionally modified by a significant amount. The characteristics of this flight allow us to make a first analysis, and validate the model for all its approximation considerations. In addition, it allow us to observe how the algorithm behaves, when subjected to a sudden change in the wind.

Due to the flight's large duration, Figure 4.11 depicts a zoom on the time frame $t \in$ [500s, 7500s], as one way to better observe the vertical trajectory of the aircraft and its yaw.



(a) Altitude.



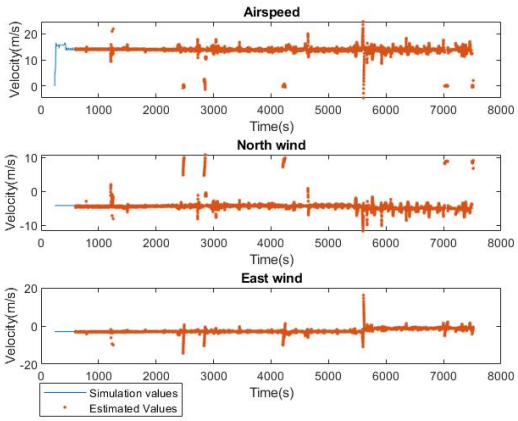
(b) Yaw.

Figure 4.11: Altitude and yaw angle during simulated flight 01.

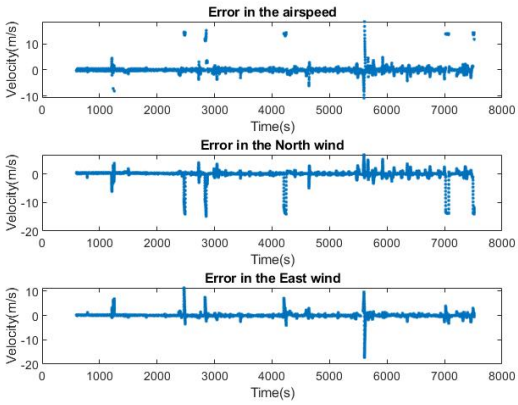
The altitude is steady most of the time, and the yaw starts off with a continuous rate of around $\dot{\psi} = 10° /s$, that decays over time, containing also steady intervals. Since the altitude is nearly constant in all its stages, with small differences over time, the horizontal leveled condition is secured.

The first test had the goal to assess that the algorithm could hold all estimations and run smoothly. To this end, the algorithm was executed, starting at a $t_{window} = 20s$, with a time step of $\Delta t = 20s$, up to a maximum $t_{max} = 360s$.

In order to seize all relevant maneuvers in the yaw, it was found necessary to first tune the matrix's condition number. As explained in chapter 3, the condition number measures how sensitive the answer is to perturbations in the input data and to round off errors, during the solution process. The metric used to quantify the accuracy of the lost digits in the estimation was based on the formula $k(H) = 10^w$, where w equals to the w digits that may be lost. To this end, first, a condition number of $k(H) = 100$ was tested, along with a mean square error of $MSE = 1m/s$ and altitude rate of $h_r = 1m/s$, as pictured in Figure 4.12.



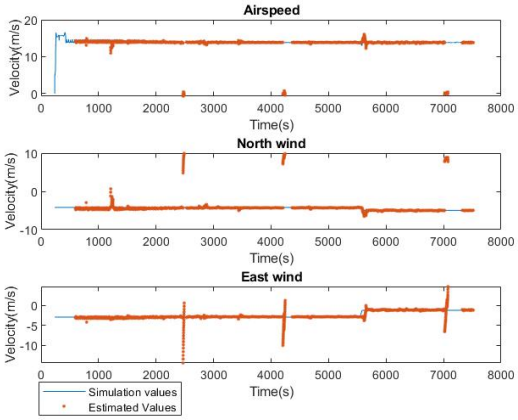
(a) Estimated parameters.



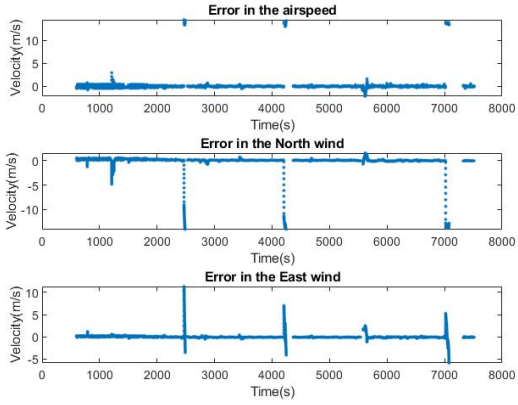
(b) Errors associated.

Figure 4.12: Autopilot Flight 01 estimates and errors associated $k(H) = 100$, $MSE = 1m/s$, $h_r = 1m/s$.

It is clear that there is a large amount of overestimates, leading to errors in the order of $e = 20m/s$. This is mainly to the chosen condition number. With a $k(H) = 100$, it is possible to lose up to $w = 2$ digits accuracy. Decreasing this value to $k(H) = 10$ ($w = 1$), results in a significant improvement of the estimates identification, demonstrated next,



(a) Estimated parameters.

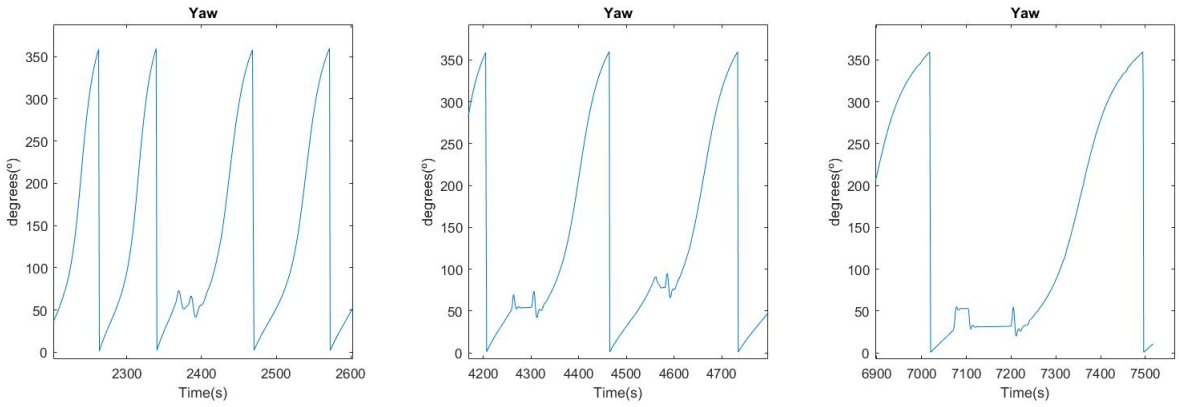


(b) Errors associated.

Figure 4.13: Autopilot Flight 01 estimates and errors associated $k(H) = 10$, $MSE = 1m/s$, $h_r = 1m/s$.

Figure 4.13 demonstrates a decrease in the overestimates, but remains with four areas in which it is still clear that they exist. These overestimates are the result of divergent values in the yaw cluster, where there is a momentary opposite direction of the yaw, as observed in figure 4.14, for three sample windows.

It was found that the undesired values occur whenever there is a steep correction of the aircraft when changing from one loiter to another one with a different radius and location.

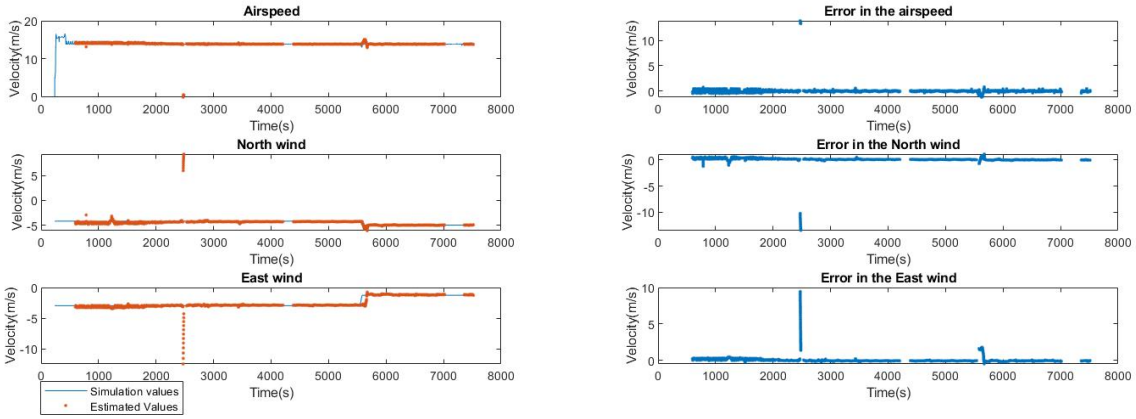


(a) $t \in [2200s, 2600s]$. (b) $t \in [4050s, 4800s]$. (c) $t \in [6900s, 7500s]$.

Figure 4.14: Zoom picture of the Yaw in several intervals.

Some of these situations are automatically ”caught” by the condition number restriction, while others with a more steep behaviour are not. This problem can be directly dealt with, by looking at these values as outliers and removing the undesired measures, or by lowering the condition number again.

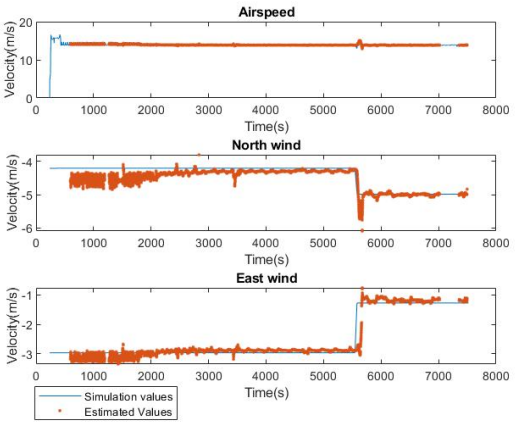
Lowering the value of the condition number to $k(H) = 5$ (w = $log_{10}(5)$ = 0.6990), leads to a decrease in the overestimates as pictured in Figure 4.15.



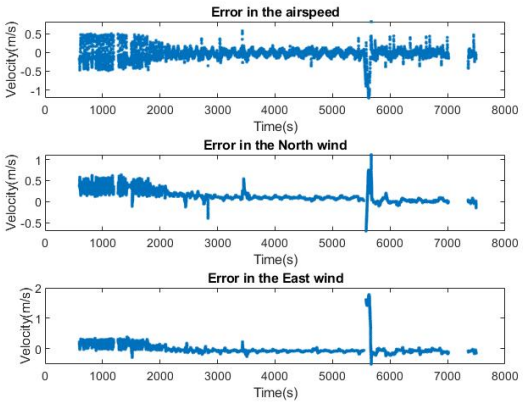
(a) Estimated parameters. (b) Errors associated.

Figure 4.15: Autopilot Flight 01 estimates and errors associated $k(H) = 5$, $MSE = 1m/s$, $h_r = 1m/s$.

After lowering the condition number down to $k(H) = 5$, most of the overestimates were successfully eliminated, except for an area around $t = 2300s$. Continuing to lower the condition number can solve this, but also continue to delete points of satisfying results. As said before, the existing overestimates are located in areas of a sudden correction of the yaw, that can be handled by locating the outliers and deleting them. The final result, after filtering the outliers, is presented in figure 4.16.



(a) Estimated parameters.



(b) Errors associated.

Figure 4.16: Autopilot Flight 01 estimates and errors associated $k(H) = 5$, $MSE = 1m/s$, $h_r = 1m/s$ (filtered).

Figure 4.16 presents promising results, as the algorithm correctly identifies all leveled intervals. With errors mainly contained in a volume area around zero, the highest peeks occur, for the airspeed, when there is a higher variation of the true airspeed then expected, and in the wind, when it encounters a sudden change in wind. Also, an important aspect to notice is the convergence of the algorithm during and after the transition of the wind at $t = 5545s$.

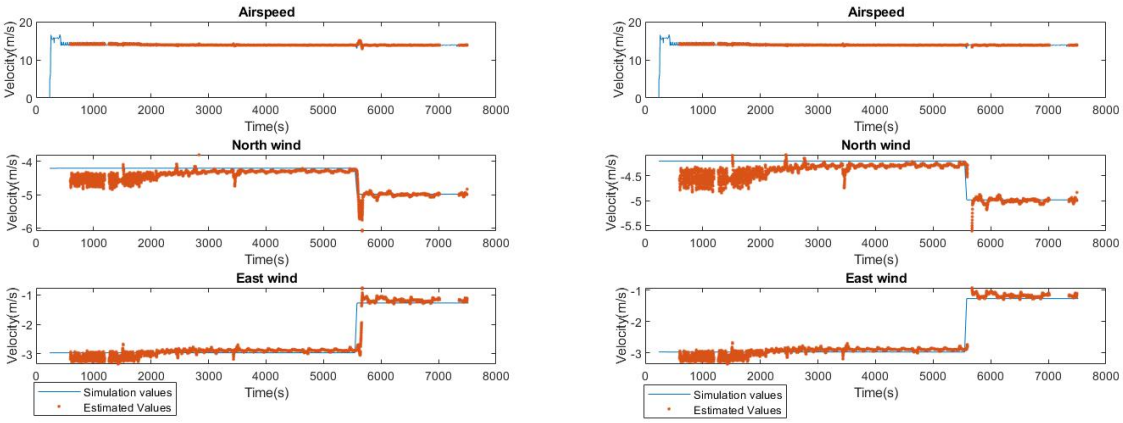
While all the errors are satisfying, the MSE was kept high. This suggests that the condition number of matrix H secures most of errors, but there should be made a compromise between these two parameters. The change of the h_r parameter is not very relevant here, since the variation in altitude is very small, but it is interesting to tune the MSE to a lower value.

Decreasing the MSE parameter to $MSE = 0.5m/s$ results in the same results as presented in Figure 4.16, suggesting that the tuning parameter $MSE = 1m/s$ can be updated to $MSE = 0.5m/s$ without loss of data.

Continuing to decrease the value to a smaller value of $MSE = 0.1m/s$, as pictured in Figure 4.17, it is noticeable the restriction of the algorithm regarding the transition of change in the wind, and while it may present some errors in its transition, these values are considered to be justifiable.

A $MSE = 0.5m/s$ is then seen as an acceptable value, considering that the algorithm is able to continue running during a wind transition, with acceptable errors associated.

The results here presented are of great importance. Since the model assumes all three parameters to remain constant in the cluster of data, the LMS finds the parameters that best fit the model. These errors are all duly justified and should provide one certainty: While it is possible to estimate the parameters and obtain pretty good results, if a big error should appear, its reason may be explained to the unexpected change of parameters, previously assumed to be constant, as it might happen in the presence of wind gusts, or a sudden acceleration in the airspeed.



(a) $MSE = 0.5m/s$.

(b) $MSE = 0.1m/s$.

Figure 4.17: Flight 01 estimates for constant $k(H) = 5$ and $h_r = 1m/s$, and variable $MSE = 0.5m/s$.

Yaw rate impact on the condition number

The impact the yaw rate has in the present study is of special interest. In the case of taking a large time window, the aircraft having information in the range of $\psi \in [0°, 360°]$, and matrix H being well conditioned, the algorithm will most likely provide good estimates. However, when dealing with a linear flight or when working with smaller time windows, this might not happen, and it is therefore relevant to study how the turning rate of the aircraft influences the results.

As one way to accomplish this, matrix H from equation 4.3 was studied. The rate of the yaw directly influences the rate of the ground course angle, and as a way of simplifying the analysis, the difference $\psi - \chi$ was here assumed to be zero ($\psi = \chi$). It was calculated the condition number for five static time windows in the interval $t_{window} \in$ [20s, 60s, 120s, 180s, 360s], versus the yaw rate, as follows,

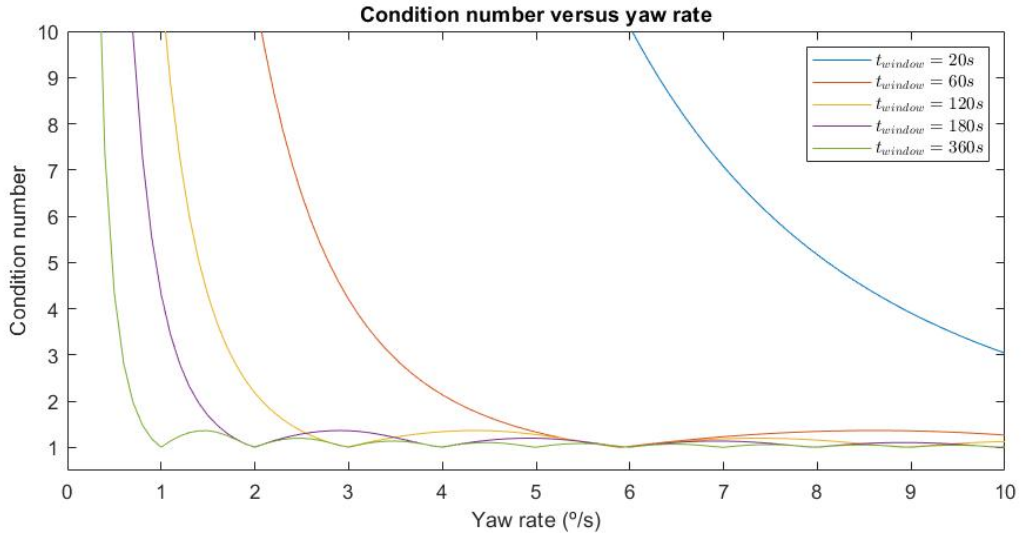


Figure 4.18: Condition number vs yaw rate.

Figure 4.18 shows the condition number vs yaw rate graph for several time windows and with a ceiling value of $k(H) = 10$. For a time window of $20s$, it takes at least a value of $\dot{\psi} = 6° /s$ to reach $k(H) = 10$,

being necessary higher values of it to eventually reach $k(H)$ close to 1. Also, at $\dot{\psi} = 10° /s$, the lowest value of the condition for a $20s$ time window is of $k(H) = 3$.

Increasing the time window, the function $k(\dot{\psi})$ is defined by an increasingly steeper slope until it reaches a value of $k(H) = 1$, point at which it is the limit of each function. As a result of the increase of the time window, the minimum value of $\dot{\psi}$ to obtain a $k(H) = 10$ decreases, as it eventually reaches $\dot{\psi} = 0.35° /s$ for the largest time frame of $t_{window} = 360s$.

These results were somehow already predicted, as it is intuitive to expect a higher turning rate to obtain reasonable results, when dealing with small time frames.

The large range of the yaw rate makes it possible to visualize this behaviour in Autopilot Flight 01. To examine this topic, there were used in the algorithm non-variable time windows of static values of multiple of 10, in the range $t_{window} \in$ [10s, 20s, 30s, 40s].

Before heading to a conclusion, it should be highlighted again that, the yaw rate of the aircraft has its peek at the beginning of the flight and decays over time. In section D.2, the evolution of the algorithm during the increasing time windows reveals that for a $10s$ time window, it is impossible to identify anything. Increasing to a $20s$ time window, the interval for which the yaw is at its highest rate makes it possible to run the algorithm, and with successive increases of $10s$ on the time window, it enables the algorithm to progressively capture slower rates of the yaw.

With this study, it is possible to assume a value of $t_{window} = 20s$ as the minimum time window to use in the given algorithm, where it can identify maneuvers described by a yaw rate contained in the interval $\dot{\psi} \in [6° /s, 10° /s]$, considering a maximum condition number of $k(H) = 10$.

Altitude variation case study

Autopilot Flight 02 was mainly an object of study for a situation of altitude variation. Even tough the model was taken in mind using a leveled flight, if the altitude of the aircraft varies slowly, it could be enough to hold the estimations.

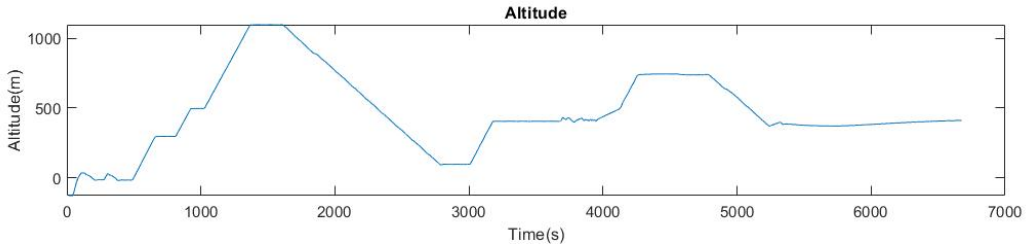
First, one important aspect to be aware is the distinction between the term **altitude rate** as a tuning parameter, and the **rate of climb** (RoC) of the aircraft. While close to one another, these should not be confused. The altitude rate concerns the change in altitude of the aircraft in a given time window, which can vary up to $t = 360s$, and the rate of climb is the vertical speed of the aircraft at each time step. However, both concepts have equal value when the aircraft progresses at a constant vertical speed, during the complete time of the data acquisition.

The aerial vehicles, used in this work, demonstrate a slow descent/climb rate, in the order of $RoC = 1m/s$. As a way to evaluate the limitations of the algorithm here presented, it is important to assess if it should be expected to obtain information, in such situations.

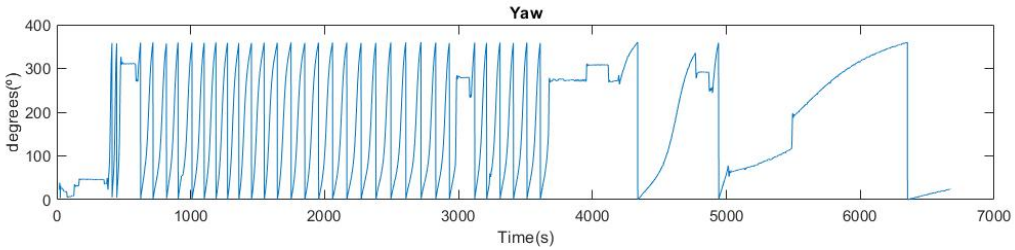
Before quantifying the climb rate, a variable with enormous impact is the time window for the acquisition of data. The maximum rate of climb found for the simulated flights was of $RoC = 2m/s$, and as such, it is a significant value to be studied. If accounting for a $5secs$ period, the change in the altitude will be of $\Delta h = 5m$, which is relatively small. However, depending on the entropy on the yaw measurements, if the time window, by chance, hits $t = 360s$, it would result on a total altitude variation of $\Delta h = 360m$.

From the beginning, the assumption of a leveled flight was assumed, and the chance of there being a total variation of $\Delta h = 360m$, makes worth studying if a low climb rate can be the enough condition to allow the estimation to go, even for a period that long.

Figure 4.19 picture the altitude and yaw evolution during Flight 02.



(a) Altitude.



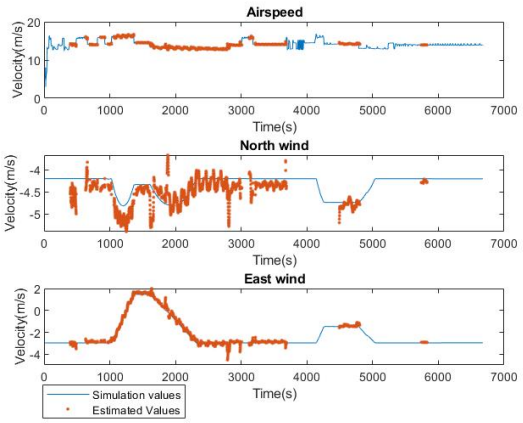
(b) Yaw.

Figure 4.19: Altitude and yaw angle during simulated autopilot Flight 02.

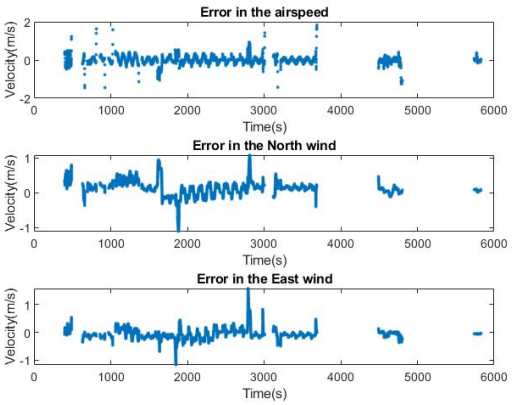
Besides the several stages in which the aircraft remains at a constant level, the interval $t \in$ [1611s - 2788s] is of special interest, due to its big slope from a height of 1096m down to 92m, at a $RoC = -1m/s$. Also, in the interval $t \in$ [1025s - 1300s] the aircraft describes a steep climb, at a $RoC = 2m/s$.

What makes it so important is the continuous yaw rate during both intervals. It is an amount of conditions that present a possibility of inspecting the limitations of the algorithm.

Based on the results of the previous flight, autopilot Flight 02 was executed with the tuning parameters $k(H) = 10$, $MSE = 0.5m/s$, and $h_r = 2m/s$, in order to capture all climb maneuvers.



(a) Estimated parameters.

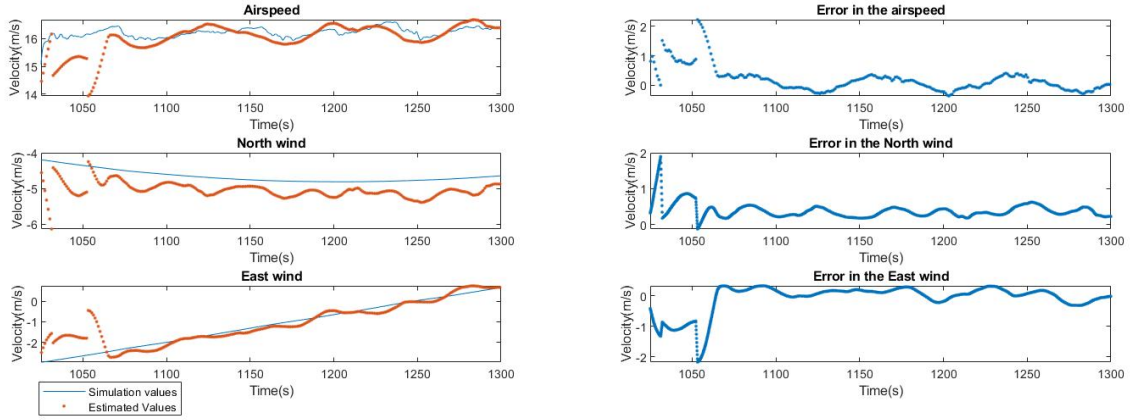


(b) Errors associated.

Figure 4.20: Autopilot Flight 02 estimations and errors associated, $k(H) = 10$, $MSE = 0.5m/s$, $h_r = 2m/s$.

Figures 4.21 and 4.22 depict a zoom in the intervals $t \in$ [1025s, 1300s] and $t \in$ [1600s, 2800s], respectively.

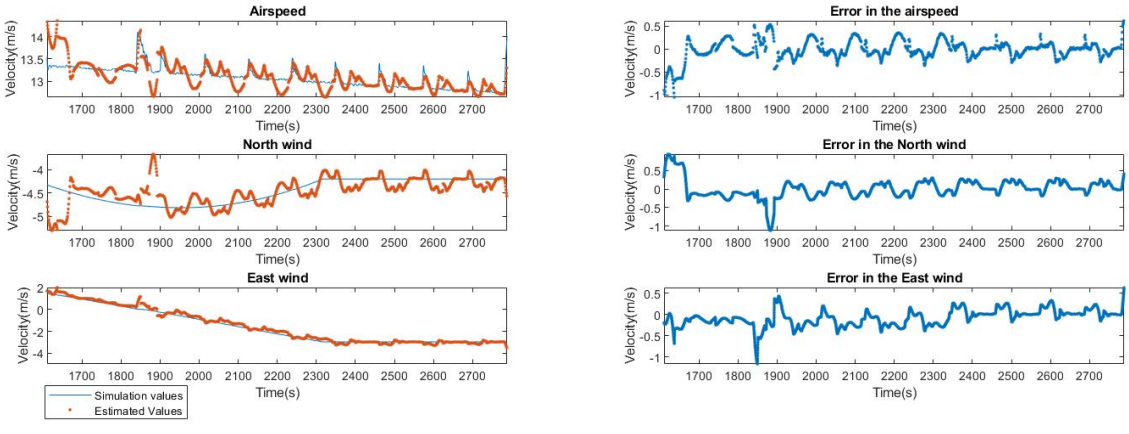
On both intervals, even tough the aircraft is in a climb/descent maneuver, the fact of it describing a ”rich” yaw maneuver enables the algorithm to run smoothly and with low errors associated. It should be noted that the higher errors occur whenever the aircraft is making a transition from a leveled altitude to a climb/descent maneuver, rather than the climb/maneuver itself.



(a) Estimated parameters.

(b) Errors associated.

Figure 4.21: Zoom picture of Figure 4.20 in $t \in$ [1025s, 1300s].



(a) Estimated parameters.

(b) Errors associated.

Figure 4.22: Zoom picture of Figure 4.20 in $t \in$ [1600s, 2800s].

The algorithm was tested up to a maximum $RoC = 2m/s$, and confirmed to continue running in this interval. Most aircraft's usually fly at low rates of climb, and at this point, considering values of RoC up to $2m/s$, it is expected for the algorithm to work, as long as the aircraft is duly excited in its yaw.

The most important aspect to take in mind is the quality of the information regarding the yaw. As long as the rate of climb doesn't compromise the quality of the information contained in matrix H, it is expected for the algorithm to correctly identify the estimates. Otherwise, the condition number and the mean square error parameters should be able to reject the response.

The end result of the flight provides us the information that during an slow varying altitude change, if given to the aircraft the enough yaw rate, it is possible to continue estimating the necessary parameters. With the variation of altitude, so does the wind components change, as well as for the airspeed. However, due to their small variations, it is made possible to carry on an estimation that eventually converges. Of course, one should be aware that whenever there is higher variation on any of the components, a higher error is going to show up, but will eventually converge. This is obvious in the errors encountered, the airspeed continues to provide the higher errors, where it peeks at around $e_{TAS} = 2m/s$, and both components of the wind at about $e_{WN} = e_{WN} = 1m/s$. This happens precisely where there is a bigger variation of their true value, which are assumed as constant values by the algorithm and are the inherent result of a sudden change in the rate of climb.

The outcome of the analysis of both flights validate the first kinematic model for its approximation conditions and also demonstrate that, when encountered by a sudden change in airspeed or wind speed, or during a steep climb/descent, the algorithm correctly converges as long as it the aircraft performs a yaw turn.

The end result of both flight's analysis, provides us with a promising online back-up solution to the estimation of the airspeed and wind.

4.4 Numerical Kinematic Model 2

The implementation behind the second numerical model is the culmination of the fusion of Parleamii's work [4], implemented along with the approach taken by C. Hurter et al [3], by applying a cluster of data to a Least Mean Squares algorithm. To this end, model 3.31, described in section 3.6.2, had to be formulated in a way it could be incorporated in a Least Squares.

Just as in the previous case, considering an airspace volume over a time interval, and assuming that the airspeed and wind remain constant within this volume, model 3.31 was applied to a given array of size data of N elements, and manipulating it in a way of defining a model that follows a linear nature, resulted in expression 4.4.

$$\underbrace{\begin{bmatrix} cos(\theta_1)cos(\psi_1) & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ cos(\theta_N)cos(\psi_N) & 1 & 0 & 0 \\ cos(\theta_1)sin(\psi_1) & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ cos(\theta_N)sin(\psi_N) & 0 & 1 & 0 \\ -sin(\theta_1) & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -sin(\theta_N) & 0 & 0 & 1 \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} TAS \\ W_N \\ W_E \\ W_D \end{bmatrix}}_{\theta} = \underbrace{\begin{bmatrix} VN_1 \\ \vdots \\ VN_N \\ VE_1 \\ \vdots \\ VE_N \\ VD_1 \\ \vdots \\ VD_N \end{bmatrix}}_{y} \tag{4.4}$$

The major difference when comparing to the first kinematic model is the inclusion of the vertical component of the ground speed, along with the pitch angle.

In expression 4.4, it is clarified the indication of each matrix when matching the whole expression to the LMS relation 4.1. The expression is now composed of three main ”chunks” of data on both the H matrix, as well as in the y matrix, in which the first N rows are directed to the North component, the second N rows to the East one and the last N rows to the Down one. The new variables, V_N, V_E and V_D are the three axis components of the ground course velocity, obtained from a GPS, and the angles of pitch θ and yaw ψ are the attitude angles obtained from the AHRS.

Following the same reasoning as in section 4.3, and rewriting this model in the form 4.2, it is possible to incorporate the second kinematic model in the LMS algorithm and estimate the parameters that best fit the model.

4.4.1 Verification and Validation

Even though this model can be seen as an extension of the previous one, the conditions for which both can run should not differ much. As stated before, the wind components for different flight levels are most likely to differ, and for that reason, the cluster of data should not contain too much entropy on the altitude. Also, the airspeed should not differ much in module, and it should be given the aircraft enough attitude excitation in the yaw and/or pitch.

It is expected for the inclusion of the pitch angle to bring advantages. In real flights, the aircraft will never remain perfectly at a constant pitch, and as such, it is expected for the additional information of this angle in matrix H to provide more accurate estimates.

Once again, autopilot Flight's 01 and 02, provided by TEKEVER, simulated in the environment of *X-Plane 10*, were used to verify and validate the kinematic model implemented in a Least Squares algorithm. The tuning of the several parameters were also chosen empirically, as a way to study both the flights. Also, the verification and validation of the second model follows typical topics, as done for the first one: the validation of the algorithm under standard conditions, the impact of the yaw and pitch in the condition number, and the analysis of the behaviour of the algorithm when subjected to an altitude maneuver.

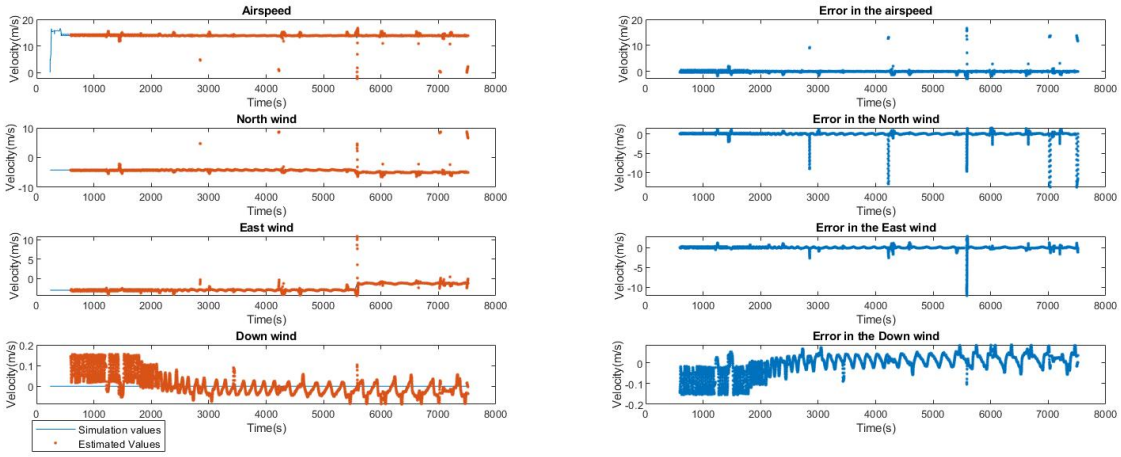
Validation in standard conditions along with sudden wind speed change

As stated before, autopilot Flight 01 contains a special trajectory that allows the validation of the model to its specifications, and investigate its behaviour when subjected to a sudden change in the wind.

As also seen before, the altitude is steady most of the flight, and therefore, the algorithm is expected to acquire estimations, as long as the yaw keeps having a non-zero rate. This doesn't happens for the entirety of the flight, as there are small ”gaps” in time where there is a steady yaw, observed in Figure 4.11, and previously discussed in the first algorithm section. Furthermore, at $t = 5545s$, there is a sudden change of the wind, that allows to observe the behaviour of the algorithm in such situations.

Same way as for the first model, the first tests were done in a way to validate the algorithm. Starting at

a $t_{window} = 20s$, the program was executed with a time step of $\Delta t = 20s$, up to a maximum $t_{max} = 360s$, and using the same tuning parameters: $k(H) = 100$, $MSE = m/s$, and $h_r = 1m/s$, the results are the ones described in Figure 4.23

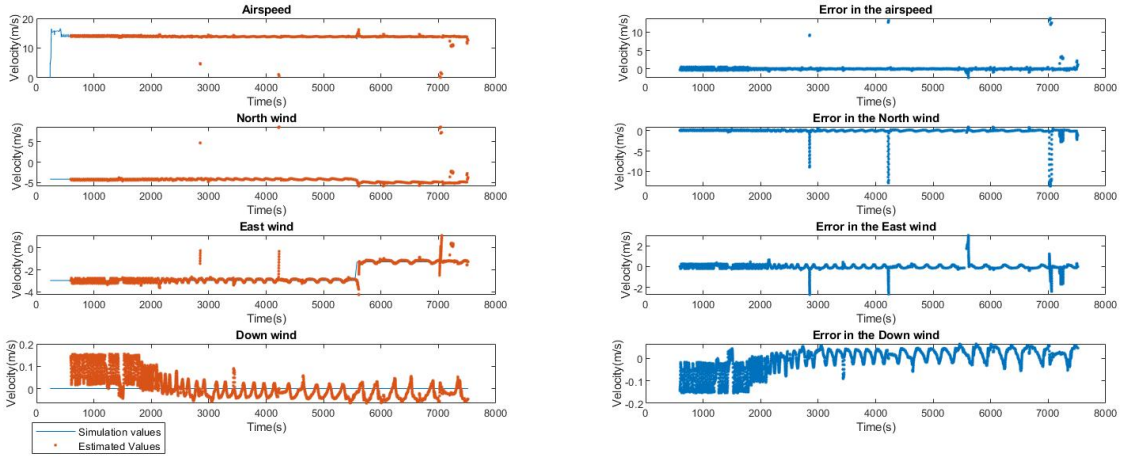


(a) Estimated parameters.

(b) Errors associated.

Figure 4.23: Autopilot Flight 01 and errors associated ($k(H) = 100$, $MSE = 1m/s$, $h_r = 1m/s$).

With a high condition number, the results are somewhat expected. The existence of overestimates is clear as it affects the scope of the graphics to large errors in the order of $e_{TAS} = 20m/s$, $e_{WN} = -14m/s$, and $e_{WE} = -12m/s$. The downward component of the wind, however, doesn't appear to be much affected, containing maximum errors in the order of $e_{WD} = -0.2m/s$. Decreasing the condition number to a value of $k(H) = 10$ affects positively the estimates as it restrains some of the overestimates, pictured next,



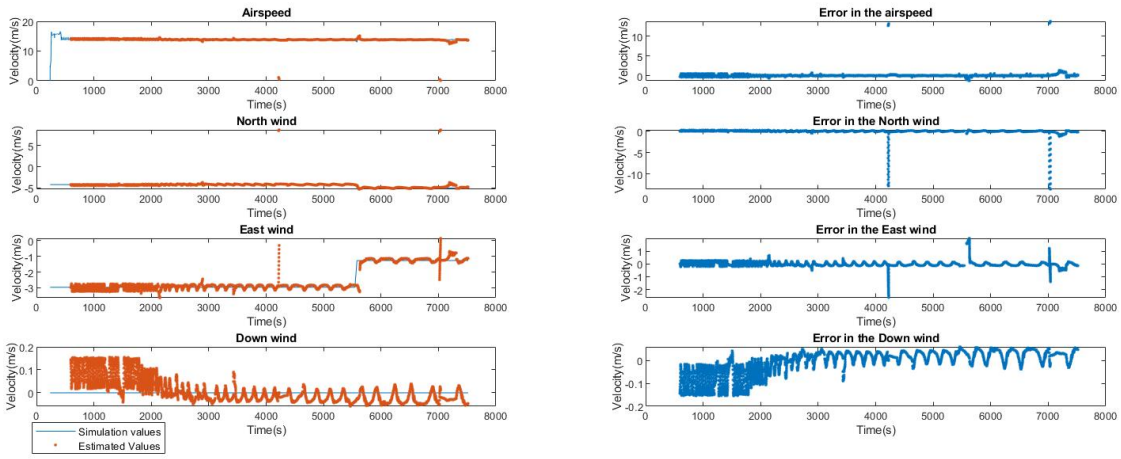
(a) Estimated parameters.

(b) Errors associated.

Figure 4.24: Autopilot Flight 01 and errors associated ($k(H) = 10$, $MSE = 1m/s$, $h_r = 1m/s$).

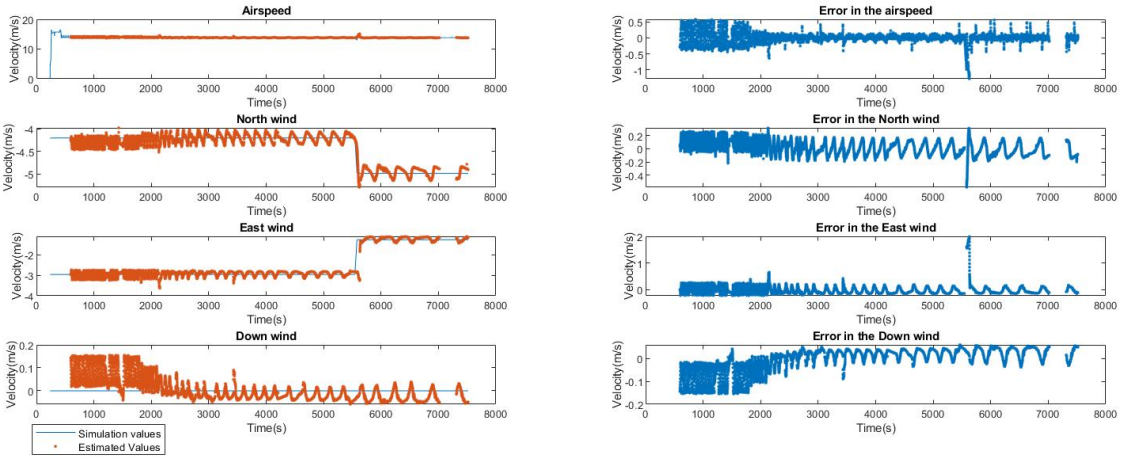
In wake of the results obtained from the tuning of the condition number for the first model, it is reasonable to say that the results just derived were already expected. By decreasing the condition

number to $k(H) = 5$ (Figure 4.25), the presence of overestimates is still clear, and to avoid the deletion of valid results, through the continuous decrease of the condition number, the next step was to filter the cluster of data for outliers, making it possible to obtain more clear results (Figure 4.26), for the same $k(H)$.



(a) Estimated parameters. (b) Errors associated.

Figure 4.25: Autopilot Flight 01 and errors associated $(k(H) = 5, MSE = 1m/s, h_r = 1m/s)$.



(a) Estimated parameters. (b) Errors associated.

Figure 4.26: Autopilot Flight 01 and errors associated $(k(H) = 5, MSE = 1m/s, h_r = 1m/s)$ (filtered).

Right away, it is noticeable the oscillatory behaviour of the estimates when comparing to Figure 4.16. This phenomenon can be explained by the inclusion of the cosine of the pitch angle in the information matrix, as will be analysed in the next topics. Therefore, this characteristic introduces an oscillatory response on the estimations.

Nevertheless, all parameters follow a reasonable response to the true value, for the constant wind intervals, and the ones right after its variation. The errors obtained, using the second kinematic model, end up reaching satisfactory results, keeping the errors of all components at a small volume area around

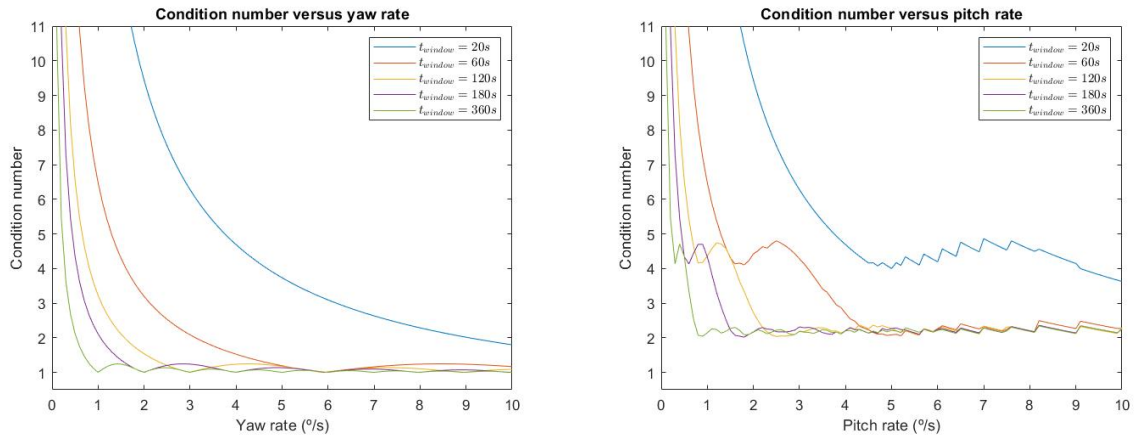
zero. A special characteristic of this model is the estimation of the downward component of the wind W_D. The simulator assumes this wind component to be zero at all flight stages and the downward wind speed ends up being the error itself, when comparing to the zero value, assumed by the simulator.

Yaw and pitch rates impact on the condition number

The second model's matrix H has the particularity of being dependent of two attitude angles, pitch and yaw, rather than the unique yaw contribution on the first model.

The inspection of the impact both variables have on the condition number is therefore more complex and, in order to analyse both contributions, first, one variable was progressively changed while the other was kept at a constant value of zero. It should be noted that, since the yaw has a range of 2π(rad), its rate describes a continuous "positive" turn, as in the case of the pitch, this variable has to be contained in an interval of $\theta \in \left[-\frac{pi}{2}, \frac{pi}{2}\right]$(rad), and therefore, the effect of the rate was expressed as a sinusoidal trajectory of the aircraft (whenever it reached a value of $\theta = \frac{pi}{2}$ or $\theta = -\frac{pi}{2}$, it would impose an opposite and equal magnitude value of the same rate).

Figure 4.27 pictures the results for the influence of both attitude angles in the matrix's condition number, over the situation where the other is equal to zero.



(a) Condition number vs yaw rate, for $\theta = 0$.

(b) Condition number vs pitch rate, for $\psi = 0$.

Figure 4.27: Condition number relation with yaw and pitch.

For a pitch angle equal to zero, the influence of the yaw on the condition number is somewhat close to the first model case, pictured in Figure 4.18. Since the structure of both model matrices is different, so are the excepted results for the impact of the yaw on the condition number. However, it is noticeable the same behaviour, where larger windows capture slower rates, and for a minimum time window of $20s$, the minimum yaw rate is of $\dot{\psi} = 1.8° /s$, considering a ceiling condition number value of $k(H) = 10$.

On the other side, for a yaw rate equal to zero, the behaviour of the condition number with the change of the pitch rate is less uniform. With the increase of the pitch rate, each of the time windows functions decrease more or less uniformly, until it is reached a point where there is an overshoot and ends up with another drop of the values to a minimum condition number of $k(H) = 2$. In the case of the minimum time

window of $20s$, the overshoot is less steep and appears much later then in the previous cases, around $\dot{\theta} = 4.5° /s$, but for the same condition number $k(H) = 4$. Also for a maximum rate of $\dot{\theta} = 10° /s$, the minimum condition number for this window is of $k(H) = 3.63$.

The fusion of both variables on the calculation of the condition number, for a time window of $60s$, is pictured in figure 4.28. Overall, besides all facts just described for the cases where $\dot{\psi} = 0$, and $\dot{\theta} = 0$, the image describes a fundamental information: While the continuous increase of the yaw rate approximates the condition number to a value of $k(H) = 1$, the pitch adds to it an oscillatory nature.

Everything points to the oscillatory contribution of the pitch to be related to the fact of this variable to be contained in the first and fourth quadrants, where opposite angles share the same cosine value.

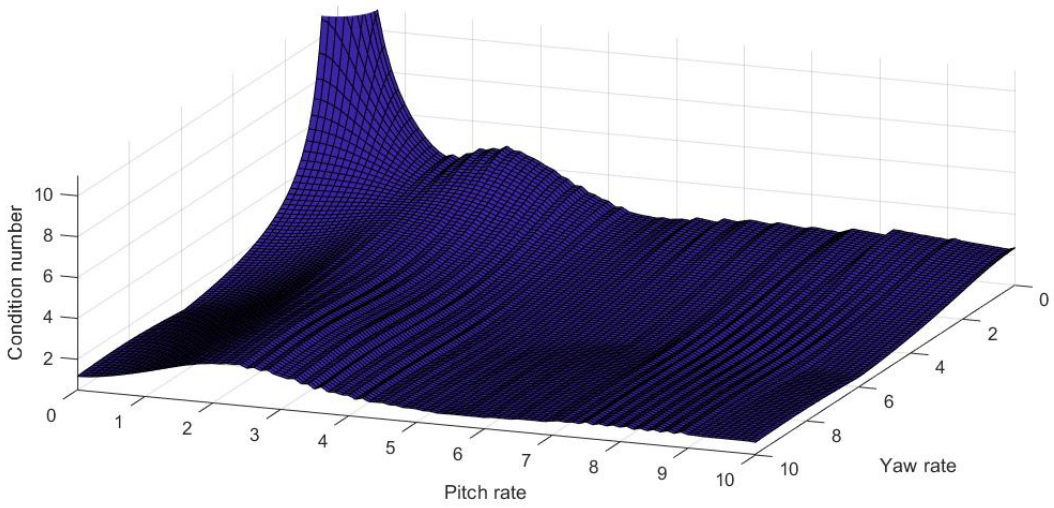


Figure 4.28: Condition number vs yaw rate vs pitch rate ($t_{window} = 60s$).

Autopilot Flight 01 presents a high range of yaw rates, while presenting pitch rates very close to zero. This makes it possible to study once again the impact of the yaw on time windows of static values. Using time windows in the range $t_{window} \in$ [10s, 20s, 30s, 40s], the algorithm was executed and the results are pictured in annex D.3. With the highest yaw rate of the aircraft in the beginning of the flight, it is possible to estimate results from a $10s$ time window, contrarily to the first kinematic model. The results for this time window is the major difference from the previous model as, just as before, with the successive increase in the time windows, it is possible to progressively capture slower rates of the yaw.

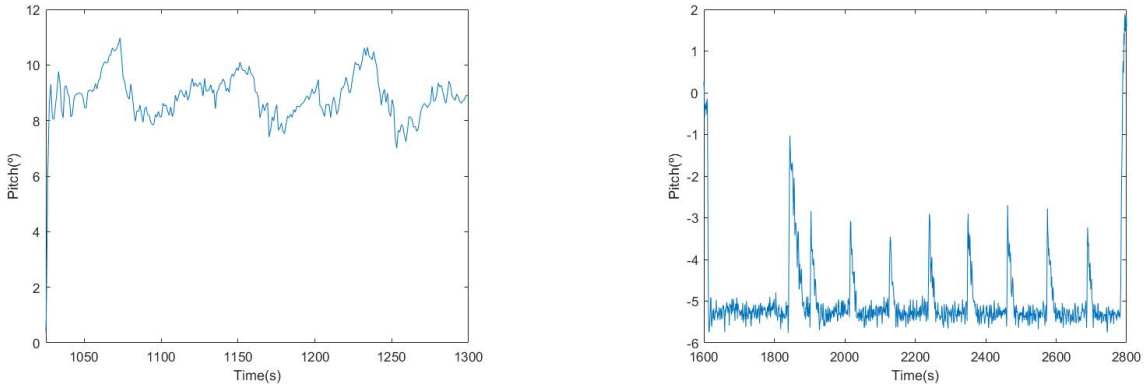
Even tough a $10s$ time window was found to enable the well functioning of the algorithm, this value is very close to the $20s$ found for the first model, and as a way to run the experimental data at the same conditions, the minimum time window for both models was defined as $20s$.

Altitude variation case study

The analysis made in autopilot Flight 01 succeeded in validating the second kinematic model to its requirements, but just as for the first model, it was found necessary to study the boundaries of the algorithm, specially when slow-varying the altitude of the aircraft, while exciting the aircraft's attitude.

To this end and just as for the first model, autopilot Flight 02 was an object of study. The flight stages in which there is a steady altitude combined with excitation of the attitude are easily identified, as pictured in Figure 4.30, but the main focus at this point is in the interval $t \in$ [1025s - 1300s], where the aircraft performs a climb maneuver of $RoC = 2m/s$, and in $t \in$ [1611s - 2788s], where there is a big slope from a height of 1096m down to 92m, at a RoC = $-1m/s$.

In both these intervals there is a continuous rate of the yaw which secures the entropy of this variable, while the pitch presents a different scenario. Figure 4.29 pictures the pitch angle in both the intervals, where it is noticeable that in the first one ($t \in$ [1025s - 1300s]) there is low entropy on the measures, and the value of the pitch is kept at a value close to 9° with variations of about 1°. On the other side, in the interval $t \in$ [1611s - 2788s], while the pitch is mainly at a -5° value, there are big variations of this attitude angle, translated by the high peeks that consequently leads to a larger entropy than in the first interval.

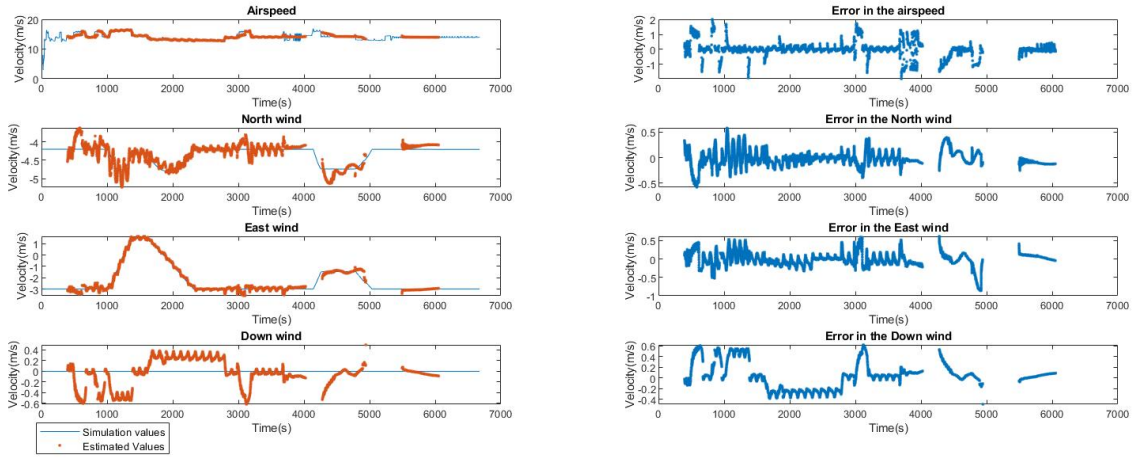


(a) Estimated parameters.

(b) Errors associated.

Figure 4.29: Zoom picture of the pitch angle.

Autopilot Flight 02 was executed with the same parameters as the ones for the first model, with the tuning parameters $k(H) = 10$, $MSE = 0.5m/s$, and $h_r = 2m/s$, as pictured next,



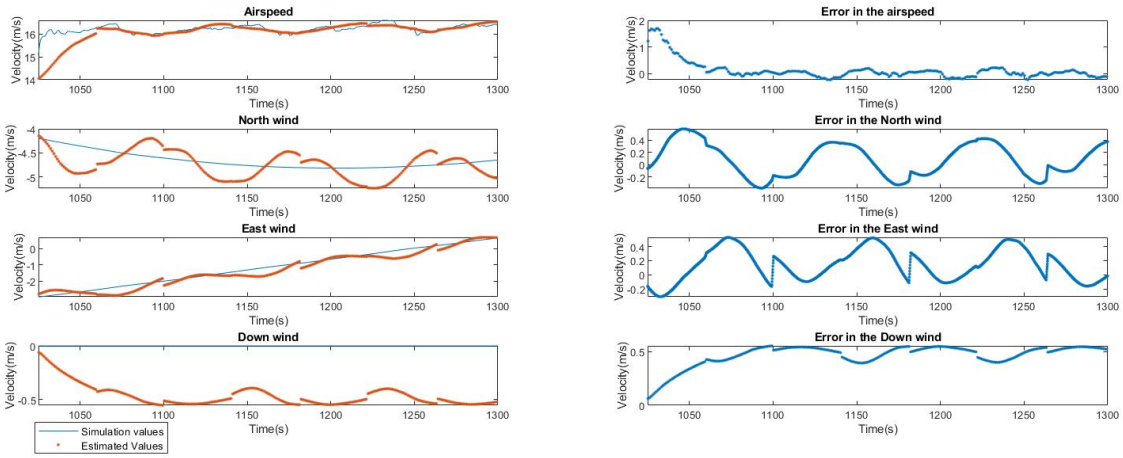
(a) Estimated parameters.

(b) Errors associated.

Figure 4.30: Flight 02 and errors associated ($k(H) = 10$, $MSE = 0.5m/s$, $h_r = 1m/s$).

Figures 4.31 and 4.32 picture a zoom of image 4.30 in the intervals $t \in$ [1025s, 1300s], and $t \in$ [1600s, 2800s], respectively.

In a situation of a steep climb, at a $RoC = 2m/s$, a continuous yaw rate of $\dot{\psi} = 4.12° \, /s$, and a pitch rate close to zero, with some oscillations, the results are pictured in Figure 4.31, where it is possible to observe very satisfying results. During the entirety of the maneuver, there is a good follow up of the estimates, when comparing to the true values, but there are noticeable high errors, especially in the airspeed, for $t = 1025s$, that comes from the faster variation of the true airspeed, and that is not taken into account on the estimates.

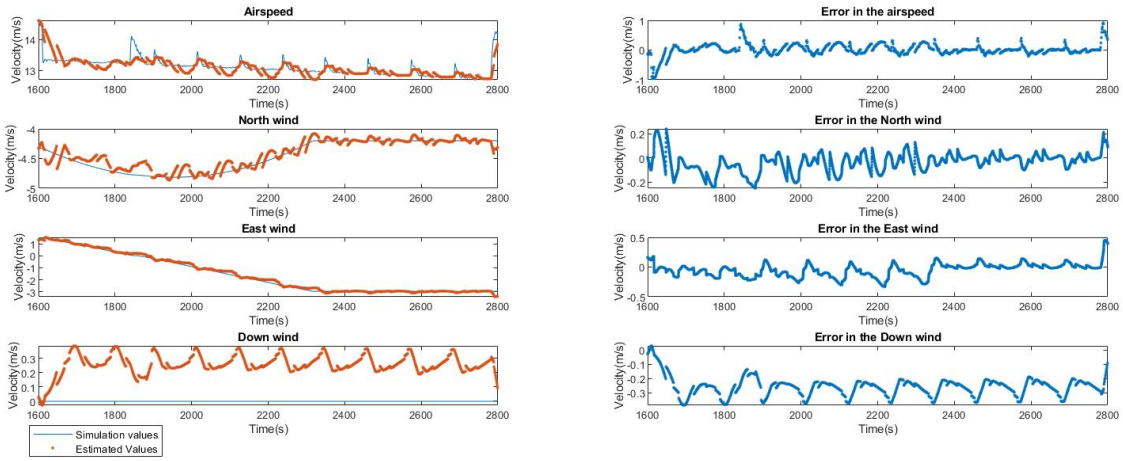


(a) Estimated parameters.

(b) Errors associated.

Figure 4.31: Zoom picture of Figure 4.30 in $t \in$ [1025s, 1300s.]

Figure 4.32 pictures a descent stage of the aircraft at a $RoC = -1m/s$, with a continuous yaw rate of $\dot{\psi} = 3.54° \, /s$, but with a less uniform pitch. The results continue to be satisfying, but the oscillatory nature of it is much more intensified, which results from the higher variation of the pitch, that, as explained before, is lead by uncertainty in the matrix H.



(a) Estimated parameters.

(b) Errors associated.

Figure 4.32: Zoom picture of Figure 4.30 in $t \in$ [1600s, 2800s].

Overall, the results obtained in all situations are quite satisfying, as the algorithm correctly identifies all valid intervals, and is able to identify the low-varying climb/descent maneuvers with quite good precision. As expected there are still significant errors in points in time and space, where there is a sudden change in variables, namely in the time frame where the rate of climb equals $2m/s$, where biggest errors still occur, namely at transition of change in the real variables.

One major fact to be concluded is the oscillatory nature of the estimates when varying too much the pitch rate, which adds uncertainty to the results. Nonetheless, the estimates are able to keep it up and eventually converge, whenever subjected to unexpected situations, like the variation of wind, or a change in the altitude. With this in mind, the validation of the current kinematic model for the conditions it was designed to work for is secured. Also, same way as concluded for the first model, as long as the yaw describes a ”rich” entropy, it should be able to provide satisfying results, always making a compromise with the entropy in the pitch, since it may also include an oscillatory nature to the response.

4.5 Comparison of models

In the previous sections there were validated two possibilities to implement the desired algorithm. In both situations it is possible to estimate important information required to flight, both when subjected to the strict requirements they were modeled to, but also when encountered by unexpected circumstances, like a wind gust, airspeed acceleration, or slow-varying altitude, as all estimates end up converging.

From a quick glance over, from the results just provided in the previous sections, it is possible to understand that, while both models were able to retain satisfactory results, the second one presents solutions closer to the real variables, but the downside is that the inclusion of the pitch can jeopardize the estimations. With the validation confirmed, it is important to quantify how relevant are the results from both models, so we can understand if the increase in complexity inherent to the second model, makes worth its use rather then the first one. To this end, three special situations were studied:

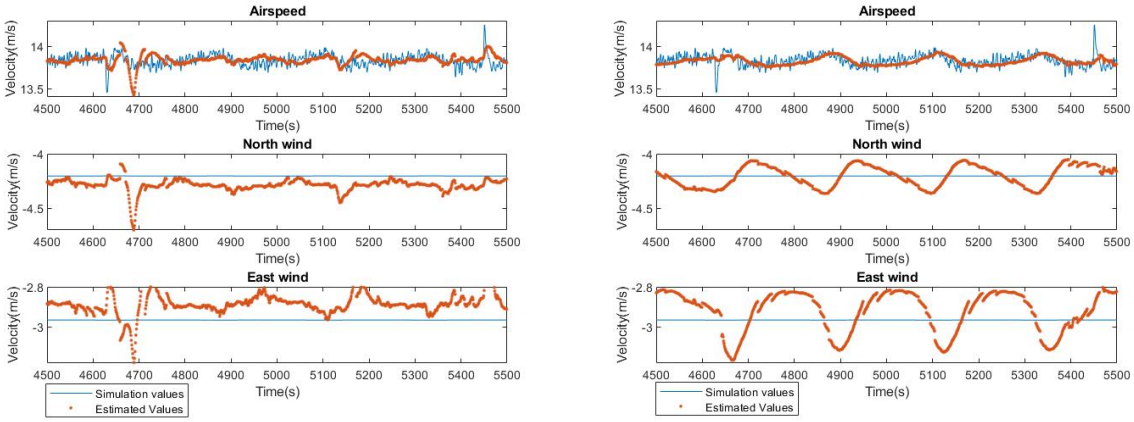
- Uniform loiter maneuver: Flight 01 - $t \in$ [4500s, 5500s];
- Variation of wind: Flight 01 - $t \in$ [5000s, 6000s];
- Slow-varying altitude: Flight 02 - $t \in$ [1800s, 2600s];

It should be highlighted that even tough the downward component of the wind appears in the second model implementation, it was left out of the comparisons, considering that the same is not obtained in the implementation with the first one.

Uniform loiter maneuver

The first interval is described as having all the required conditions for the algorithm to run: the aircraft is kept at the same height, while maneuvering the aircraft in a loiter, which secures the necessary entropy on the yaw. As seen earlier in the chapter, both kinematic models enable the estimation of the necessary

parameters in this situation, and the results for both, in the time frame $t \in$ [4500s, 5500s], are pictured in Figure 4.33.



(a) First kinematic Model.

(b) Second kinematic Model.

Figure 4.33: Algorithm estimates in $t \in$ [4500s, 5500s].

At first glance it is possible to notice that, in all variables, there is a closer answer to the real values with the second kinematic model implementation than with the first, even tough the oscillatory nature, already referred before, is clearly noticed. The error graphics for both situations is presented in annex D.4, and Table 4.5 summarizes the mean, standard deviation and absolute maximum values for both implementation errors.

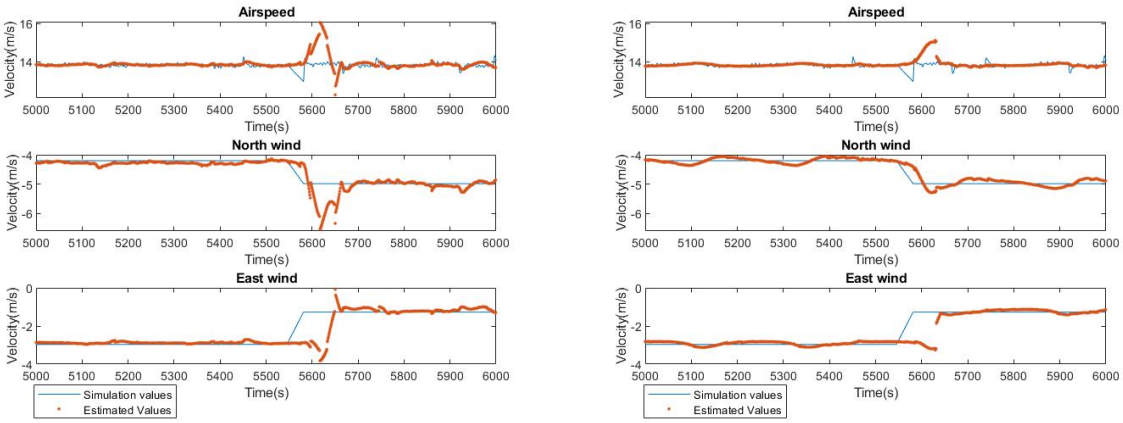
	Kinematic Model 1			Kinematic Model 2		
Errors	$\mu(m/s)$	$\sigma(m/s)$	$\lvert e_{max}\rvert(m/s)$	$\mu(m/s)$	$\sigma(m/s)$	$\lvert e_{max}\rvert(m/s)$
e_{TAS}	-0.0054	0.0827	0.3597	0.0086	0.0654	0.4348
e_{W_N}	0.0833	0.0575	0.4961	0.0012	0.0950	0.1618
e_{W_E}	-0.0806	0.0508	0.2586	-0.0473	0.0989	0.1873

Table 4.5: Statistic values of the errors in $t \in$ [4500s, 5500s].

Regarding the airspeed, both models present solutions with similar error statistics, where the mean value is very close to zero, with the standard deviation also very close to one another and a maximum error value justified by the sudden change in the real value at $t = 5450s$. On the other hand, both wind components have a smaller mean value on the second kinematic model, but also present higher standard deviation, inherent of the oscillatory response. Furthermore, the maximum values of the error is mitigated in the second model.

Variation of Wind

The scenario in which there is a clear change of the wind, for the same leveled flight, is pictured in Figure 4.34, where in the time frame $t \in$ [5000s, 6000s], it describes the estimations moments before, during, and after the transition of wind value. In the Figure, the results for the implementation of both models is presented as one way to visually identify their behaviour.



(a) First kinematic Model.

(b) Second kinematic Model.

Figure 4.34: Algorithm estimates in $t \in$ [5000s, 6000s].

At about $t = 5545$ there is a change in the wind components that lasts about $50s$ until it stagnates for the rest of the flight. Before this transition, both models correctly identify all parameters while clearly noticing the same oscillatory response on the second model. In the transition time frame of $50s$ there are large errors in all components of the first model, while these are mitigated in the second model. This is also clear due to information regarding the errors, pictured in annex D.5, and information regarding mean, standard deviation and absolute maximum values of all parameters errors is presented in Table 4.6.

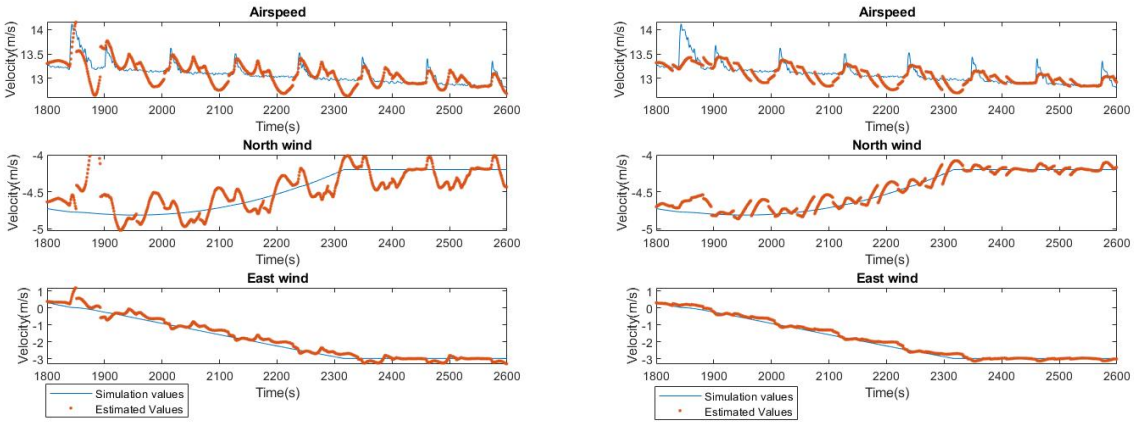
Continuing on the transition area, the first model presents maximum errors of airspeed, north and east wind speeds much larger on the first model, rather than the second. Besides this, in the given time frame, while all mean, standard deviation, and maximum errors are relatively small for both models, the second one contains smaller values on all statistical parameters, indicating that the second model has a better follow up response during these situations.

	Kinematic Model 1			Kinematic Model 2		
Errors	$\mu(m/s)$	$\sigma(m/s)$	$\lvert e_{max}\rvert(m/s)$	$\mu(m/s)$	$\sigma(m/s)$	$\lvert e_{max}\rvert(m/s)$
e_{TAS}	-0.0691	0.3336	2.157	-0.0469	0.2256	0.4475
e_{W_N}	0.0974	0.2248	1.558	-0.0166	0.1192	0.5807
e_{W_E}	0.0202	0.4759	2.558	0.0601	0.4150	1.989

Table 4.6: Statistic values of the errors in $t \in$ [5000s, 6000s].

Slow-varying altitude

The final scenario is one of great importance: a slow-varying altitude maneuver. It has been seen before that both models are able to run in this circumstance, but it is important to check in detail how much different one is from another. It is therefore presented the estimation at the time frame $t \in$ [1800s, 2600s] in Figure 4.35, for both implementations, where there is a slow-varying altitude change, at a $RoC = -1m/s$.



(a) First kinematic Model.

(b) Second kinematic Model.

Figure 4.35: Algorithm estimates in $t \in$ [1800s, 2600s].

When dealing with a change in altitude, one inherent consequence is the variation in the wind components, which is visually noticed by the blue lines in the graphics. Up until $t = 2300s$ the altitude is continuously changing, and from then on is kept the same.

The errors for both situations is presented in annex E.2, and Table 4.7 complements the information in the annex, comprising information regarding mean, standard deviation and absolute maximum values for the errors.

From the graphics it is visually observed that both implementations on the altitude change is smooth, but from the error functions, it is concluded that the second model presents a higher mean for the airspeed and north wind component, while having smaller standard deviations for all parameters and lower peek errors for both wind components.

	Kinematic Model 1			Kinematic Model 2		
Errors	$\mu(m/s)$	$\sigma(m/s)$	$\lvert e_{max}\rvert(m/s)$	$\mu(m/s)$	$\sigma(m/s)$	$\lvert e_{max}\rvert(m/s)$
e_{TAS}	0.0166	0.1767	0.5316	0.0403	0.1529	0.8669
e_{W_N}	-0.0202	0.2046	1.080	-0.0419	0.0718	0.2498
e_{W_E}	-0.0971	0.2092	1.168	-0.0699	0.1107	0.3316

Table 4.7: Statistic values of the errors in $t \in$ [1800s, 2600s].

Overall, it is concluded that the complexity inherent to the implementation of the second kinematic model on the algorithm doesn't present enough advantages to be chosen over the first one. In all situations in which, it presented more satisfying results, the difference in the errors was not utterly big, besides the fact of it adding an oscillatory nature to the response, that could be amplified by the quality of the pitch angle measurements.

Chapter 5

Experimental results

The solution to the estimation of the several parameters was built on the adaptation of two separate kinematic models in a Least Mean Squares algorithm, that fused data of attitude and gps velocity. In chapter 4, the implementation of the proposed algorithm was validated using two kinematic models, along with the analysis of some important aspects concerning the limitations of the algorithm in both cases. With the validation concluded in a controlled environment, followed the experimental testing for Flight 01, 02, and 03, previously referred in section 4.2.2. This chapter presents the implementation of the algorithm previously described, with both kinematic models, in real flight situations and the subsequent results.

5.1 Standard estimation values of the triplet TAS, W_N, and W_E

In real flight situations, the impact of the environment affects all measurements, also translated into noise effects. This is not present in the simulation environment used in the previous chapter, and therefore all values of airspeed and wind field were assumed as equal to their true value.

Contrarily to what was described before, in real situations the true value of these parameters is not know, but the use of sensors and filtering techniques enable the autopilot to estimate them. A basic architecture of the standard method to estimate these values is pictured in Figure 5.1, where it was kept as simple as possible.

In the diagram, it is possible to observe an Extended Kalman Filter block that receives information from measurement sensors as a GPS, INS, and pitot tubes, and data from the aircraft model, upon receiving inputs from the control surface commands.

The Kalman filter is an algorithm that incorporates all the available information from the knowledge of the system and measurement dynamics, statistical description of the system noises, measurement errors, uncertainty in the dynamics models, and any available information about initial conditions for variables of interest [42]. It is a wide-range used technique for numerous situations, in which there is the necessity to estimate variables of unknown value, and it is therefore exploited by TEKEVER and many other companies, where there is made a special emphasis on the Extended Kalman Filter, also know as the non-linear Kalman Filter, that linearizes the non-linear functions for each estimate.

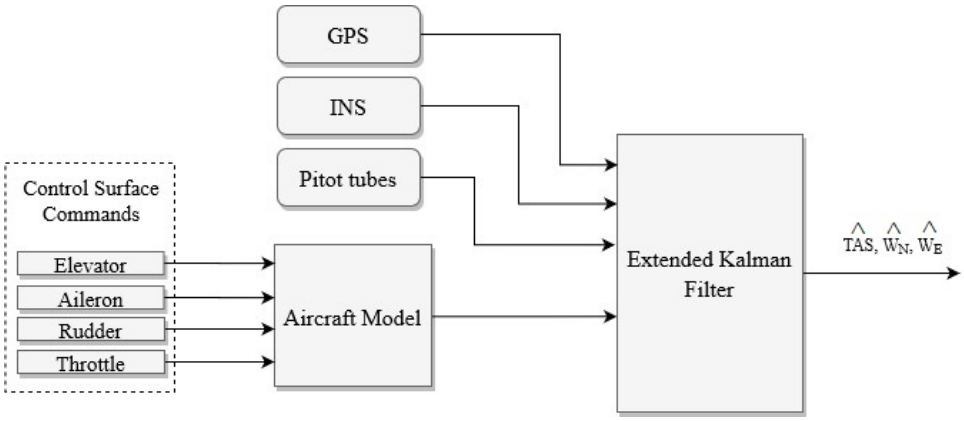


Figure 5.1: Triplet TAS, W_N, and W_N estimation architecture.

This is a large discussion topic that will not be looked upon very close, since it has no direct impact on the work here provided. However, since the parameters of reference for the following sections are supplied from this kind of system, it is of important matter to refer it. Also, due to using the values of this system as reference value, the term **residual** arises to replace the **errors**.

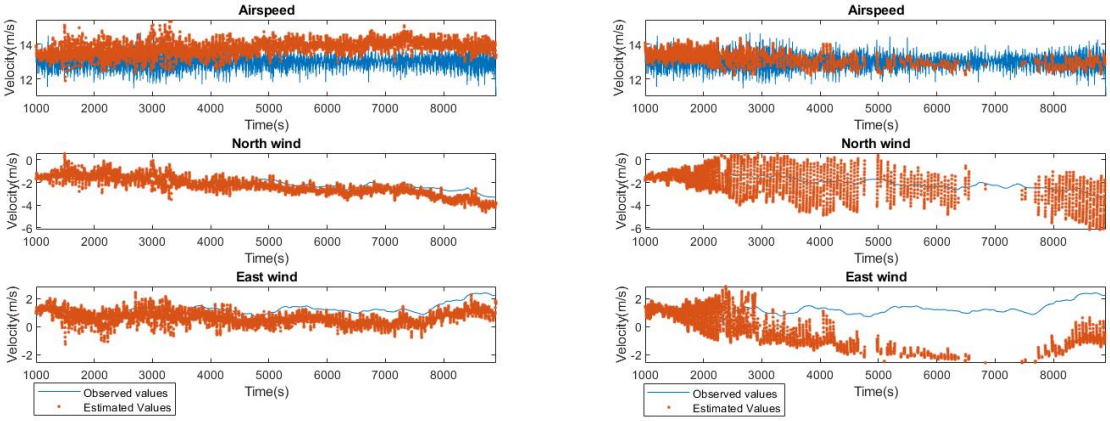
In chapter 4, the term **error** was used to evaluate the deviation from the estimated values to their true value. Now, the **residual** is used as a metric of evaluation, and is defined as the difference between the observed value (in this case, given by the standard EKF architecture) and the estimated value of the quantity of interest.

5.2 Experimental tests

As stated before, data from three flights was provided by TEKEVER, as a way to test if the work here provided could be used in real-life mission situations. The same flights were previously used to study the behaviour of the wind in the vicinity of each considered time step, in section 4.1, and each describes the type of missions, the algorithm was originally built for.

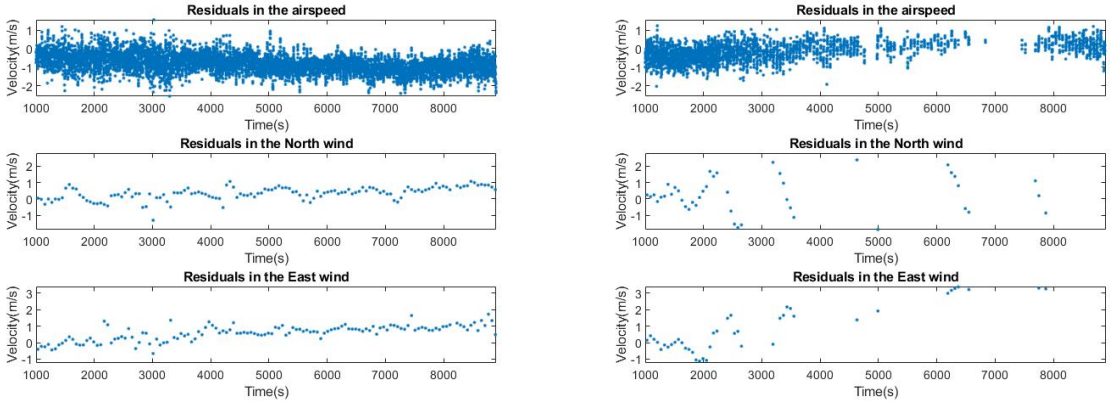
All flights were executed with the same tuning parameters, considered as standard values, of $k(H) = 10$, $MSE = 1m/s$, and $h_r = 2m/s$. For each of the flights both the first moments, in which the aircraft was kept at the ground, and the last ones, where the aircraft is parachuted, were rejected, as one way to visually observe the results in a clear resolution. Also, for every experimental test is associated the respective residuals graphics and a table containing complementary information regarding mean, standard deviation and absolute maximum residual values.

Flight 01



(a) First kinematic Model. (b) Second kinematic Model.

Figure 5.2: Flight 01 experimental outcome.



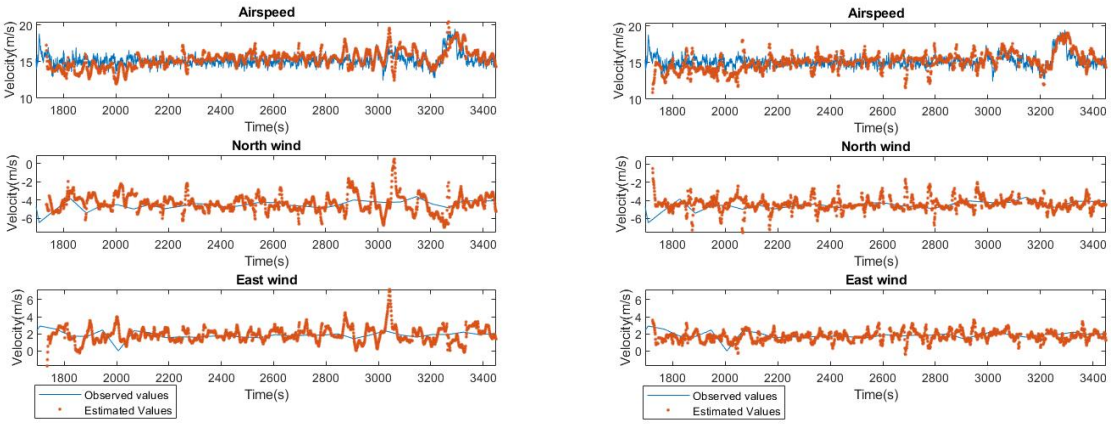
(a) First kinematic Model. (b) Second kinematic Model.

Figure 5.3: Flight 01 residuals.

	Kinematic Model 1			Kinematic Model 2		
Residuals	$\mu(m/s)$	$\sigma(m/s)$	$\lvert r_{max}\rvert(m/s)$	$\mu(m/s)$	$\sigma(m/s)$	$\lvert r_{max}\rvert(m/s)$
r_{TAS}	-0.7989	0.4889	2.4060	-0.1671	0.4975	2.0301
r_{W_N}	0.3370	0.3870	1.3040	0.2452	1.0741	2.3912
r_{W_E}	0.5913	0.4591	1.7230	0.8381	1.4012	3.3742

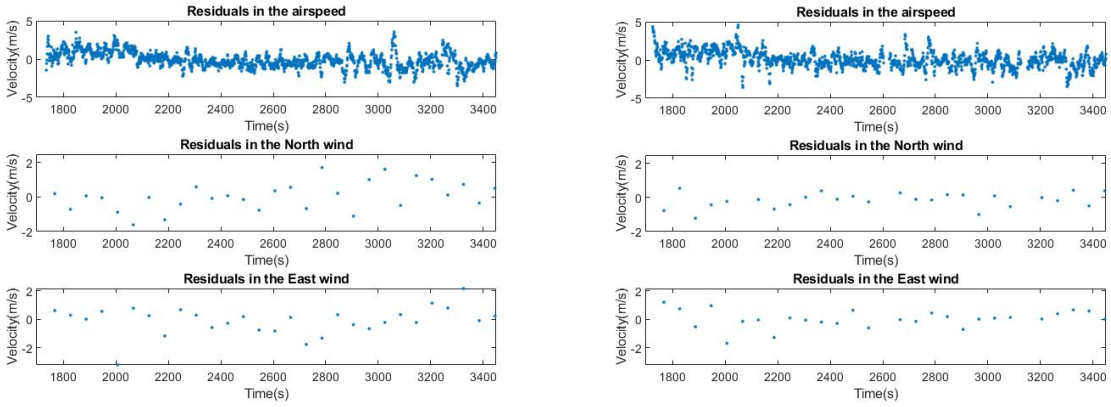
Table 5.1: Flight 01's statistic values of the residuals.

Flight 02



(a) First kinematic Model. (b) Second kinematic Model.

Figure 5.4: Flight 02 experimental outcome.



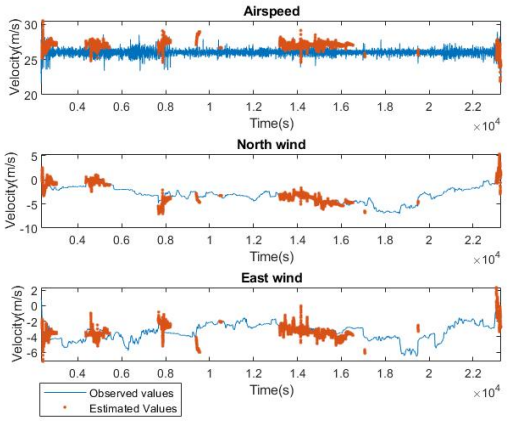
(a) First kinematic Model. (b) Second kinematic Model.

Figure 5.5: Flight 02 residuals.

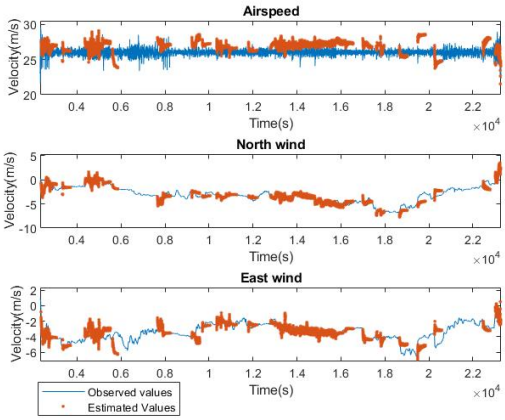
	Kinematic Model 1			Kinematic Model 2		
Residuals	$\mu(m/s)$	$\sigma(m/s)$	$\lvert r_{max}\rvert(m/s)$	$\mu(m/s)$	$\sigma(m/s)$	$\lvert r_{max}\rvert(m/s)$
r_{TAS}	-0.1434	1.0321	3.5401	0.07937	1.0721	4.5772
r_{W_N}	0.0518	0.8291	1.7171	-0.1581	0.4410	1.2143
r_{W_E}	-0.1892	0.9063	3.2190	0.0034	0.6273	1.6921

Table 5.2: Flight 02's statistic values of the residuals.

Flight 03

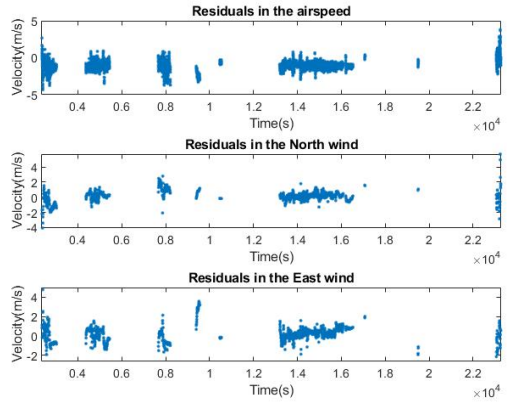


(a) First kinematic Model.

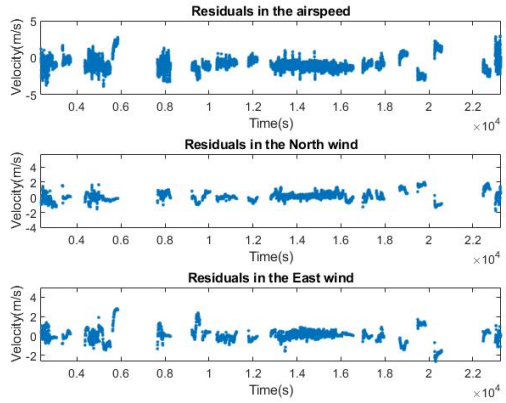


(b) Second kinematic Model.

Figure 5.6: Flight 03 experimental outcome.



(a) First kinematic Model.



(b) Second kinematic Model.

Figure 5.7: Flight 03 residuals.

	Kinematic Model 1			Kinematic Model 2		
Residuals	$\mu(m/s)$	$\sigma(m/s)$	$\lvert r_{max}\rvert(m/s)$	$\mu(m/s)$	$\sigma(m/s)$	$\lvert r_{max}\rvert(m/s)$
r_{TAS}	-1.081	0.6162	4.3353	-0.8677	0.8803	-3.8823
r_{W_N}	0.1454	0.7296	4.9452	0.1287	0.6091	1.9912
r_{W_E}	0.1905	0.8036	4.7833	0.0744	0.7203	2.8283

Table 5.3: Flight 03's statistic values of the residuals.

General overview

The experimental testing for Flight 01, 02, and 03, along with their residual values is pictured from Figure 5.2 up to 5.7.

Flight 01 and 02 represent two flights of vigilance, described by a huge amount of time in a loiter maneuver. Flight 03, for another side, is much longer than the other two and is described by several intervals in which the aircraft contains loiter stages and transitions between them.

From $t = 1000s$ up to $t = 9000s$, Flight 01 contains the necessary conditions to provide estimates, and both models are able to "catch" the maneuver. The first kinematic model implementation describes a good response follow up in all three components with a mean residual of the airspeed at $r_{TAS} = -1.7989m/s$ and the wind field with $r_{W_N} = 0.3370m/s$ and $r_{W_E} = 0.5913m/s$. The second model, however, while providing smaller mean residuals in the airspeed, and the north component of the wind, depicts larger standard deviations in all parameters, besides the fact of it being unable to be executed in full time period. These two downsides can be explained by the oscillatory behaviour of the response that, as observed in the figures, is intensified in time, specially in the north component of the wind.

The results in the second flight are interesting, because both models yield similar solutions. They both run during the full extent of the interval $t = 1750s - 3450s$, and the statistical values of the residuals are somewhat close to one another. The two have very small mean values for all the residuals, with the second model containing the smaller standard deviations, aside for the airspeed.

Lastly, the final flight is the longest one and contains information from different flight situations. Aside from the earlier flights, that were described by a specific mission stage - the loiter, this one presents also steady stages. From the naked eye, it is possible to conclude that the second model is able to identify more intervals than the first, which are the smaller intervals of yaw variation. This was already expected, since the analysis of the condition number in the previous section already demonstrated that the structure of the second model matrix was advantageous in the captation for smaller yaw rates. Lastly, and in a general way, the mean values of the residuals continue to be small, with a large value of r_{TAS} close to $1m/s$ for both implementations, and rather small standard deviations in all parameters under $\sigma = 0.9m/s$.

The maximum residuals found in all situations was always found to be fairly large, but the analysis for this characteristic was left for last, since the reasons inherent to these situations are the approximations taken in hand when modeling the algorithm. By assuming all the parameters to be constant in the given time window, directly results in higher discrepancies when comparing to real measurements that include more variations than assumed. Also, even tough the EKF has a reputation to provide results that are very close to the true values of the parameters, there remains associated errors, that culminates in a higher difference in the residuals.

The results in all three flights were very satisfying and both models secured most of the estimates in all situations, within an acceptable error. Also, the contrast in results the second model leads to, when compared to the first one, are of small improvement, and in some cases even worst, when there is a more intense oscillatory behaviour in the response. Besides the increase in the computation complexity it comprises, due to the entanglement matrices, the results it provides are not of significant relevance.

From the start it was expected to obtain values with lower accuracy than the ones provided by the Kalman filter due to the approximation nature of the solution, but it is enough to serve as a ”guideline”. It should be noted that if it were possible to encompass more information of the attitude of the aircraft in the kinematic model, a third kinematic model would arise that could decrease the oscillatory behaviour observed in the second model, and with the possibility to provide more accurate information for all the given parameters.

Chapter 6

Conclusions

The work presented in this document was motivated by a growing necessity to find alternative methods capable of replacing conventional sensors. Most systems take use of a classic Extended Kalman Filter, or other derived Kalman filters, fusing data from several sensors, as is the case of pitot tubes, GPS, accelerometers, gyroscopes, etc, together with a dynamic model of the aircraft to estimate in-flight parameters such as true airspeed, wind speed, angle of attack and sideslip.

The continuous restrictions to have smaller and lighter systems, specially in drones, limits the amount of sensors to be fit in an aircraft, and every system is kept as simple as possible. The pitot tube is an example of a sensor that is usually installed in accordance with at least two others, in order to keep a physical redundancy in the values. However, the instalment of this sensor implies having heavier equipment and a higher system complexity, and the solution is to keep the numbers of these sensors down to two or even just one.

The true airspeed of the aircraft is a core parameter that depends strongly of pitot tubes sensors, and the main focus of this work was to find an alternative to this sensor, that could provide enough redundancy to the already placed sensors.

From the beginning this work was a challenge. There are researchers that looked at this problem and tried to solve it using at least one pitot tube, or even deleting it, but using wind vanes, incorporated in an Extended Kalman Filter. Even so, the main objective was to completely eliminate the pitot tube, as one way to be independent of this sensor. Also, it was presented at a point, that it would be interesting to make a system completely independent of the aircraft as well, since it is know that the attainment of an aircraft's model is something complex, expensive and time-consuming.

Alongside TEKEVER, there were done some brainstorming ideas that could be put into action, and one of the first was the interpretation of the Parleami's theory, that was found to be extremely interesting. Later, with the approach of C. Hurter et al. and in discussion with TEKEVER, it was found relevant to also incorporate the Parleami's theory in the given solution.

With the exclusion of the pitot tubes, wind vanes, and the dynamic model of the aircraft, it was found a way of reaching all the required parameters using only data provided by an AHRS and a GPS, given that the aircraft would flight in a certain way for algorithm to capture maneuvers.

6.1 Achievements

The major achievement of the present work was the design of a system capable of being incorporated in any aircraft, and able to estimate the airspeed and wind field, under certain conditions. Although the conditions for which it was designed are restricted to an horizontal flight, together a rich yaw maneuvering, and assuming the parameters to remain constant within a given time, it is possible to retain satisfying results for small variations of altitude and of the assumed constant parameters.

The associated errors are acceptable since it is the result of an approximated modulation of the aircraft's airspeed incorporated in a Least Squares algorithm, which is not filtered or "corrected" as it happens in usual Kalman Filters. The larger errors occur whenever there is a bigger variation in any of the parameters than expected, which tends to occur when the aircraft suddenly speeds up, or in the transition to start a climb/descent maneuver. If duly evaluated, these situations of larger errors can be locally rejected by an operator, with the assumption that the response of the system will eventually converge to its real value.

Also, a second modulation was incorporated in the algorithm as an attempt to include the pitch to obtain more accurate parameter values. However, the results were not so different when comparing to the first approach, that was simpler and less complex computationally. Besides, it was led to believe that the fact of the pitch to remain in the first and fourth quadrants, and opposite angles to share the same cosine function, to affect the response with an oscillatory behaviour.

This being said, an on-line estimation of the triplet TAS, W_N, and W_E is executed by the system here provided and supplies a "virtual redundancy" to the already existing standard air data system. The extent of its use relies on the operator using it, and when in doubts if the pitot tube is obstructed, it is always possible to maneuver the aircraft in a way for the algorithm to "catch" the necessary information and to get an idea of the parameter's magnitude.

6.2 Future Work

The work here presented achieved the desired results, but there is always room for improvement.

A first development for future work is the modulation of the kinematic of the aircraft. As was seen, a change in the kinematic model used ends up with distinct results that can or not, be better than the ones tested here. An option for this would be to attempt to incorporate the third attitude angle, the roll, in the kinematic model, which could be enough to, alongside the pitch and yaw, estimate more reliable values of the three parameters.

In the present work, the condition number of the matrix was found to be a good metric of the measurements in the cluster data, but it still requires a lot of computational effort. An optimization of this step could provide a significant improvement in the performance of the system.

Lastly, the Least Squares is used as a prime tool to fit the cluster data into the model and estimate the most suitable parameters. Nevertheless, its use requires the multiplication of matrices that can be as large as the chosen window, and the use of an alternative method to analyse the data, could also provide

an optimal and faster solution to the problem, or at least less susceptible to outliers, as for example the use of Least trimmed .

Bibliography

[1] D. G. of Civil Aeronautics. FINAL AVIATION ACCIDENT REPORT ALW-301 PUERTO PLATA. Technical report, feb 1996.

[2] B. d'Enquêtes et d'Analysespour la sécurité de l'aviation civile. Final Report on the accident on 1st June 2009 to the Airbus A330-203 . Technical report, jul 2012.

[3] C. Hurter, R. Alligier, D. Gianazza, S. Puechmorel, G. Andrienko, and N. Andrienko. Wind parameters extraction from aircraft trajectories. *Computers, Environment and Urban Systems*, pages pp 1–16, Feb. 2014. doi: 10.1016/j.compenvurbsys.2014.01.005.

[4] William Premerlani. IMU Wind Estimation. `https://diydrones.com/forum/topics/wind-estimation-without-an`, 2009. Online; accessed 10 March 2019.

[5] G. J. NAZAROFF, L. D. LEVSEN, and G. A. HEWER. Placement of observer eigenvalues. *AIAA Journal*, 10(12):1686–1688, Dec. 1972. doi: 10.2514/3.6704.

[6] T. DUFFY. Decoupled estimation techniques applied to trajectory reconstruction. In *Astrodynamics Conference*. American Institute of Aeronautics and Astronautics, Aug. 1976. doi: 10.2514/6.1976-820.

[7] J. JOSEPH ZEIS, H. LAMBERT, R. CALICO, and D. GLEASON. Angle of attack estimation using an inertial reference platform. In *15th Atmospheric Flight Mechanics Conference*. American Institute of Aeronautics and Astronautics, Aug. 1988. doi: 10.2514/6.1988-4351.

[8] T. Thacker. Use of state estimation calculate angle of attack position error from flight test. *M.S. Thesis, GAE/M/85J-3, Wright-Patterson AFB, Ohio: Air Force Institute of Technology*, Oct 85.

[9] D. Freeman. Angle of attack computation system. *AFFDL- TR-73-89. Wright-Patterson AFB, Ohio: Air Force Flight Dynamics Laboratory*, Oct 73.

[10] B. Petrov and R. S. et al. Determining angles of attack and sideslip by signals from accelerometers installed on board an aircraft. *"Izvestiya Vysshikh Uchebnykh Zavedeniy, 'Priborostroyeniye'," Vol 18, Nr 10*, 1975.

[11] J. OLHAUSEN. The use of a navigation platform for performance instrumentation on the YF-16 flight test program. In *13th Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics, Jan. 1975. doi: 10.2514/6.1975-32.

[12] K. Wise. Flight testing of the x-45a j-UCAS computational alpha-beta system. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*. American Institute of Aeronautics and Astronautics, Aug. 2006. doi: 10.2514/6.2006-6215.

[13] A. Murch. A flight control system architecture for the NASA AirSTAR flight test infrastructure. In *AIAA Guidance, Navigation and Control Conference and Exhibit*. American Institute of Aeronautics and Astronautics, Aug. 2008. doi: 10.2514/6.2008-6990.

[14] S. Hansen, M. Blanke, and J. Adrian. Diagnosis of UAV pitot tube defects using statistical change detection. *IFAC Proceedings Volumes*, 43(16):485–490, 2010. doi: 10.3182/20100906-3-it-2019.00084.

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

[16] M. Fravolini, M. Pastorelli, S. Pagnottelli, P. Valigi, S. Gururajan, H. Chao, and M. Napolitano. Model-based approaches for the airspeed estimation and fault monitoring of an unmanned aerial vehicle. In *2012 IEEE Workshop on Environmental Energy and Structural Monitoring Systems (EESMS)*. IEEE, Sept. 2012. doi: 10.1109/eesms.2012.6348401.

[17] S. Gururajan, M. L. Fravolini, H. Chao, M. Rhudy, and M. R. Napolitano. Performance evaluation of neural network based approaches for airspeed sensor failure accommodation on a small UAV. In *21st Mediterranean Conference on Control and Automation*. IEEE, June 2013. doi: 10.1109/med.2013.6608784.

[18] F. A. P. Lie and D. Gebre-Egziabher. Synthetic air data system. *Journal of Aircraft*, 50(4):1234–1249, July 2013. doi: 10.2514/1.c032177.

[19] M. B. Rhudy, M. L. Fravolini, Y. Gu, M. R. Napolitano, S. Gururajan, and H. Chao. Aircraft model-independent airspeed estimation without pitot tube measurements. *IEEE Transactions on Aerospace and Electronic Systems*, 51(3):1980–1995, July 2015. doi: 10.1109/taes.2015.130631.

[20] M. Pachter, N. Ceccarelli, and P. Chandler. Estimating MAVs heading and the wind speed and direction using GPS, inertial, and air speed measurements. In *AIAA Guidance, Navigation and Control Conference and Exhibit*. American Institute of Aeronautics and Astronautics, Aug. 2008. doi: 10.2514/6.2008-6311.

[21] A. F. Rodríguez, E. Andersen, J. M. Bradley, and C. N. Taylor. Wind estimation using an optical flow sensor on a miniature air vehicle. *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference 2007*, 3:2791–2798, 2007. doi: 10.2514/6.2007-6614.

[22] N. D. Divitiis. Wind estimation on a lightweight vertical-takeoff-and-landing uninhabited vehicle. *Journal of Aircraft*, 40(4):759–767, July 2003. doi: 10.2514/2.3155.

[23] J. W. Langelaan, N. Alley, and J. Neidhoefer. Wind field estimation for small unmanned aerial vehicles. *Journal of Guidance, Control, and Dynamics*, 34(4):1016–1030, July 2012. doi: 10.2514/1.52532.

[24] J. H. Lee, H. E. Sevil, A. Dogan, and D. Hullender. Estimation of maneuvering aircraft states and time-varying wind with turbulence. *Aerospace Science and Technology*, 31(1):87–98, Dec. 2013. doi: 10.1016/j.ast.2013.09.009.

[25] M. B. Rhudy, Y. Gu, J. N. Gross, and H. Chao. Onboard wind velocity estimation comparison for unmanned aircraft systems. *IEEE Transactions on Aerospace and Electronic Systems*, 53(1):55–66, Feb. 2017. doi: 10.1109/taes.2017.2649218.

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

[27] M. Kayton and W. Fried. *Avionics Navigation Systems*. A Wiley-Interscience publication. Wiley, 1997. ISBN 9780471547952.

[28] R. Collinson. *Introduction to Avionics Systems*. Springer Netherlands, 2002. ISBN 9781402072789.

[29] Collins Aerospace. Air Data Systems. `https://utcaerospacesystems.com/product_gallery/air-data-systems/`. Online; accessed 20 September 2019.

[30] E. A. Haering. Airdata Measurement and Calibration Airdata Measurement and Calibration. *Nasa Technical Memorandum*, (December), 1995.

[31] AirTeamImages. ATR 42-500. `https://www.airteamimages.com/atr-42__-_168077.html`. Online; accessed 20 September 2019.

[32] A. Helfrick. *Principles of Avionics*. Avionics Communications, 2010. ISBN 9781885544278.

[33] A. Helfrick. *Principles of Avionics*. Avionics Communications, 2010. ISBN 9781885544278.

[34] J. Valasek. Small unmanned aircraft: Theory and practice. *Journal of Guidance, Control, and Dynamics*, 36(1):344–345, Jan. 2013. doi: 10.2514/1.61067.

[35] Basic Air Data. Coordinate System. `https://www.basicairdata.eu/knowledge-center/background-topics/coordinate-system/`, 2013. Online; accessed 5 October 2019.

[36] J. Oliveira. Apontamentos de estabilidade de voo. Instituto Superior Técnico. 2016.

[37] L. Debnath. The legacy of leonhard euler – a tricentennial tribute. *International Journal of Mathematical Education in Science and Technology*, 40(3):353–388, Apr. 2009. doi: 10.1080/00207390802642237.

[38] Frontmatter. In *Global Positioning Systems, Inertial Navigation, and Integration*, pages i–xxvi. John Wiley & Sons, Inc. doi: 10.1002/9780470099728.fmatter.

[39] R. Beard and T. McLain. *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press, 2012. ISBN 9781400840601.

[40] E. Cheney and D. Kincaid. *Numerical Mathematics and Computing*. Cengage Learning, 2012. ISBN 9781133103714.

[41] Department of Atmospheric Sciences, University of Utah. MesoWest Data. `https://mesowest.utah.edu/`, 2013. Online; accessed 12 December 2009.

[42] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, Mar. 1960. doi: 10.1115/1.3662552.

Appendix A

TEKEVER flight 01

A.1 Flight characteristics

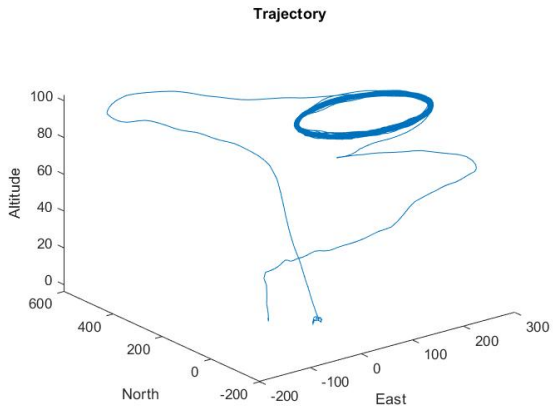


Figure A.1: Three dimensional trajectory.

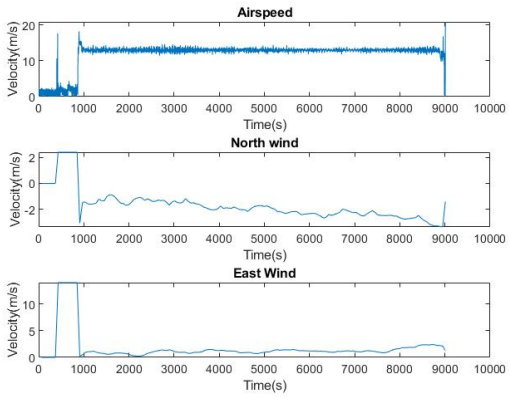


Figure A.2: Simulated airspeed and wind components.

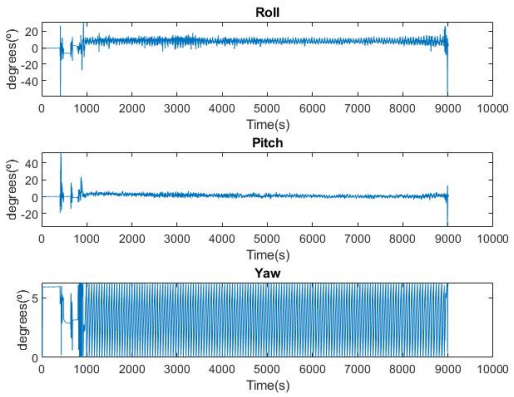


Figure A.3: Euler angles.

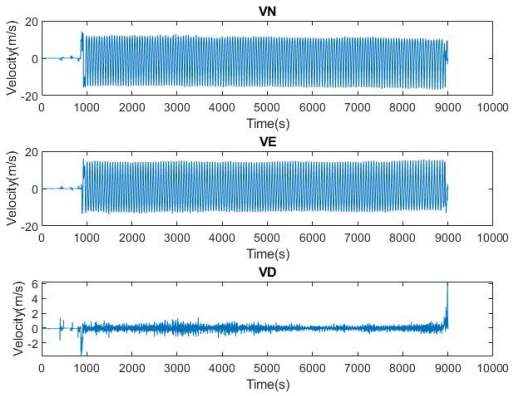


Figure A.4: Ground speed components (NED frame).

Appendix B

TEKEVER flight 02

B.1 Flight characteristics

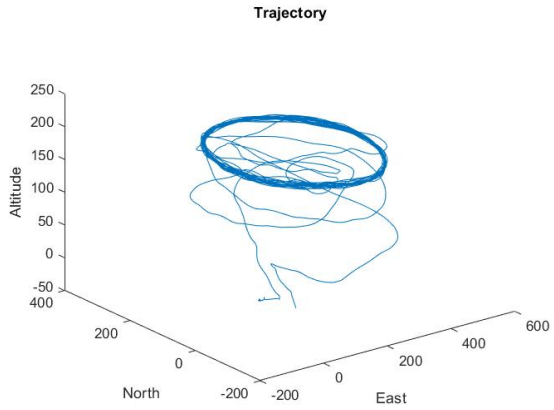


Figure B.1: Three dimensional trajectory.

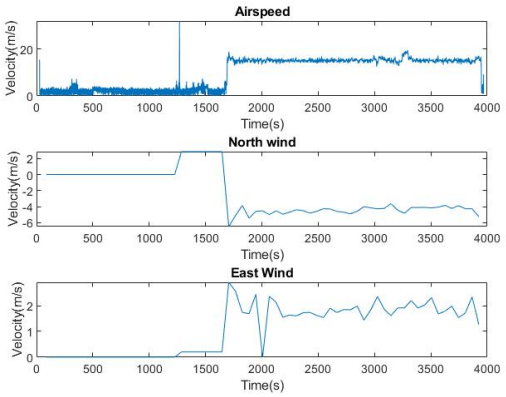


Figure B.2: Simulated airspeed and wind components.

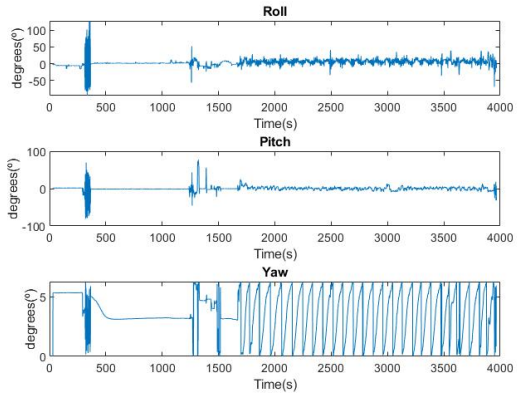


Figure B.3: Euler angles.

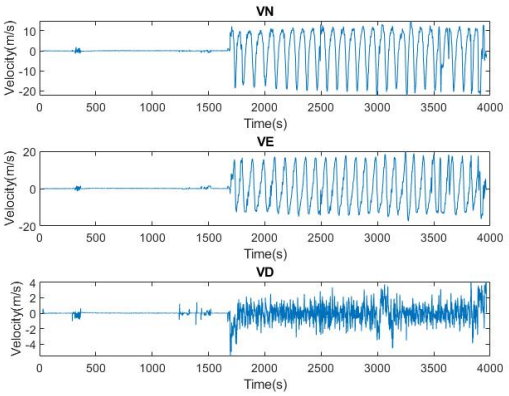
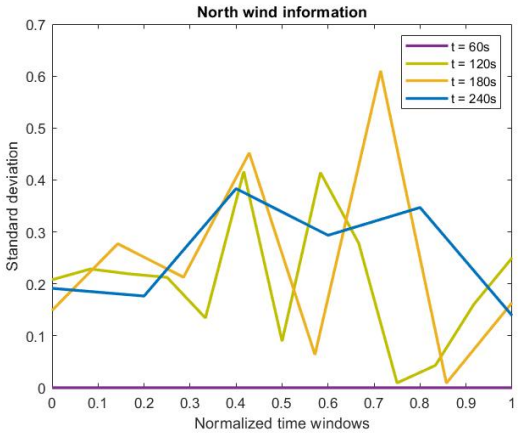
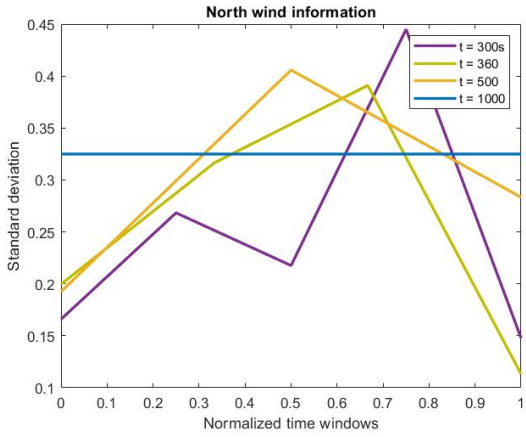


Figure B.4: Ground speed components (NED frame).

B.2 Standard deviation study

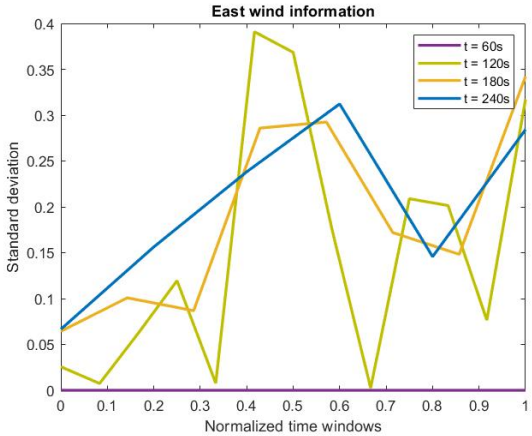


(a) t ∈ {60; 120; 180; 240}.

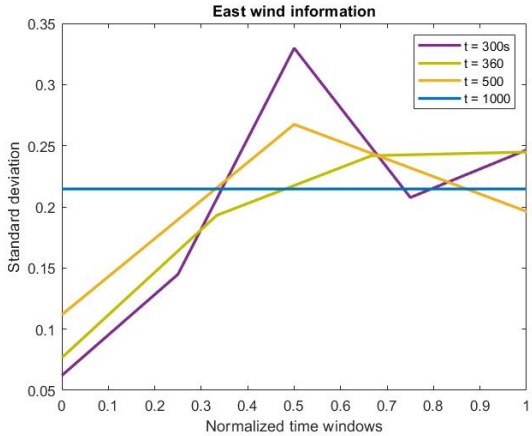


(b) t ∈ {300; 360; 500; 1000}.

Figure B.5: Dispersion of wind values with increasing time windows, North component.



(a) t ∈ {60; 120; 180; 240}.



(b) t ∈ {300; 360; 500; 1000}.

Figure B.6: Dispersion of wind values with increasing time windows, East component.

Appendix C

TEKEVER flight 03

C.1 Flight characteristics

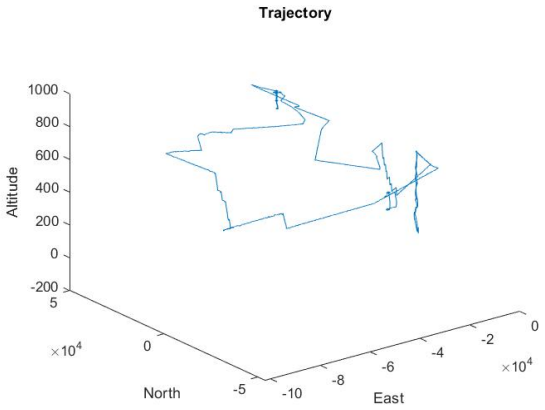


Figure C.1: Three dimensional trajectory.

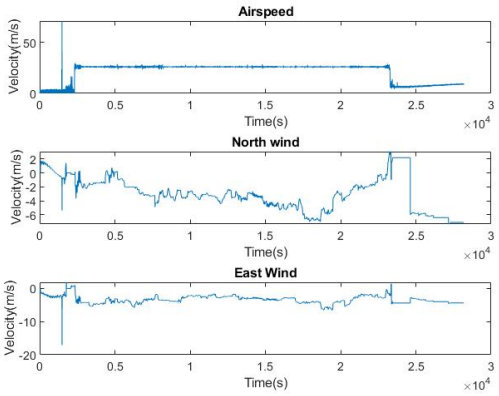


Figure C.2: Simulated airspeed and wind components.

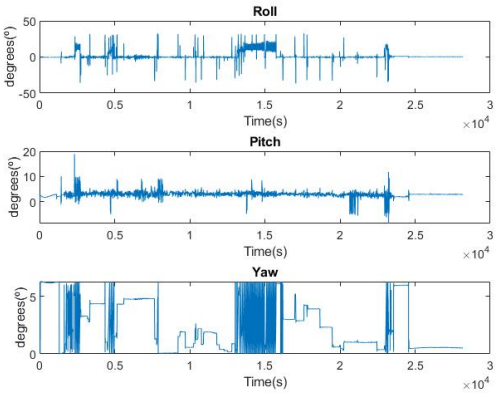


Figure C.3: Euler angles.

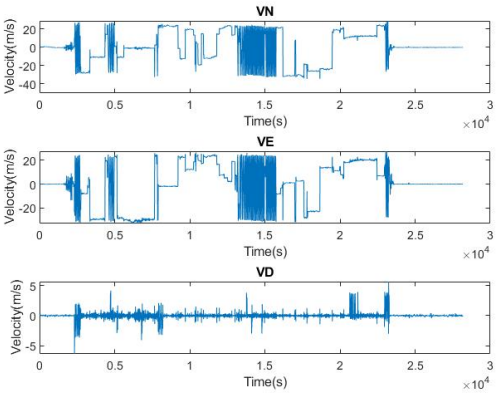
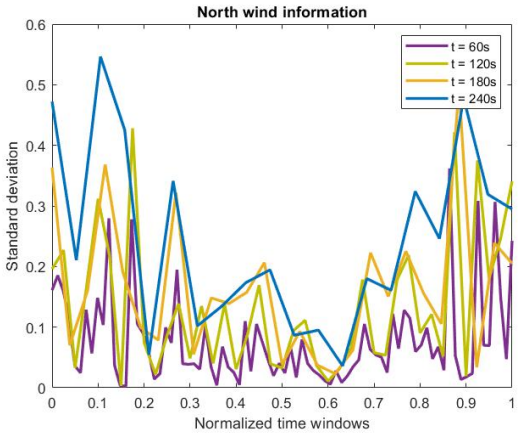
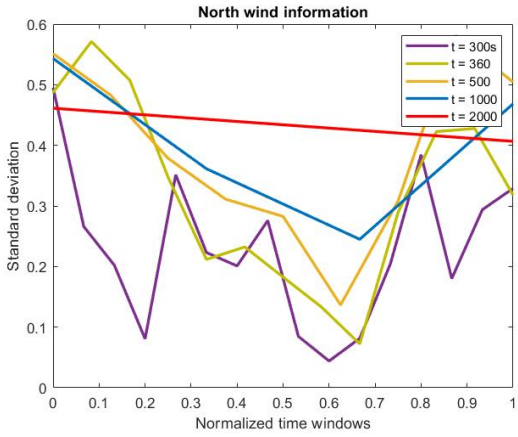


Figure C.4: Ground speed components (NED frame).

C.2 Standard deviation study

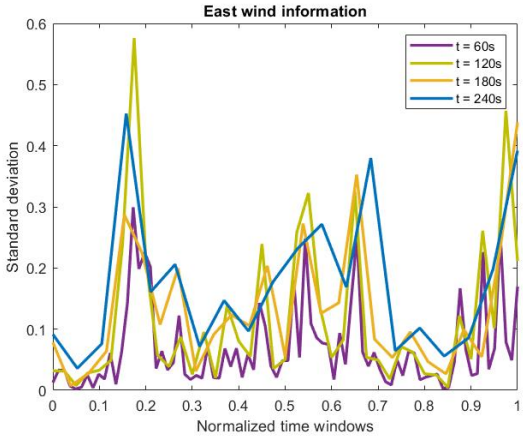


(a) t ∈ {60; 120; 180; 240}.

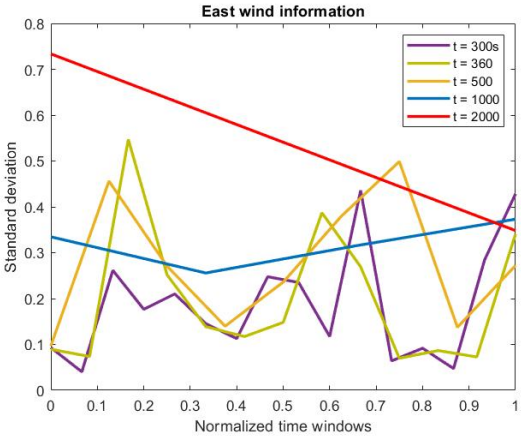


(b) t ∈ {300; 360; 500; 1000; 2000}.

Figure C.5: Dispersion of wind values with increasing time windows, North component.



(a) t ∈ {60; 120; 180; 240}.



(b) t ∈ {300; 360; 500; 1000; 2000}.

Figure C.6: Dispersion of wind values with increasing time windows, East component.

Appendix D

Simulated autopilot flight 01

D.1 Flight characteristics

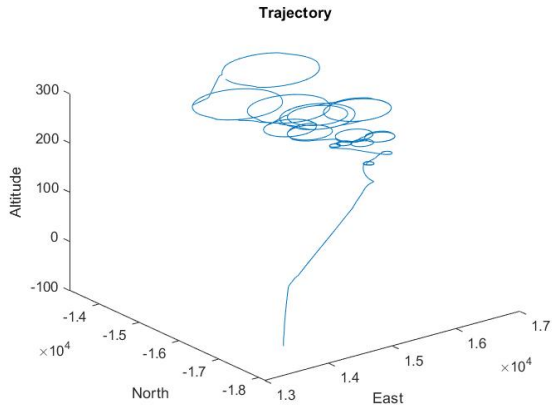


Figure D.1: Three dimensional trajectory.

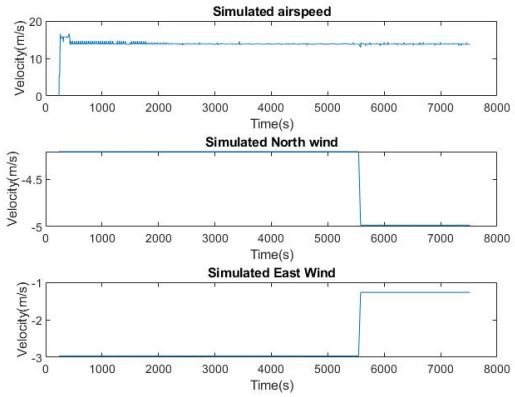


Figure D.2: Simulated airspeed and wind components.

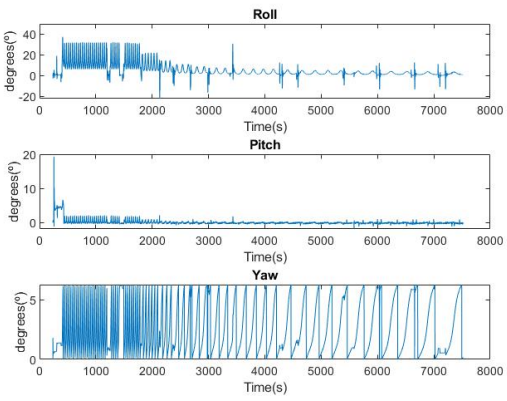


Figure D.3: Euler angles.

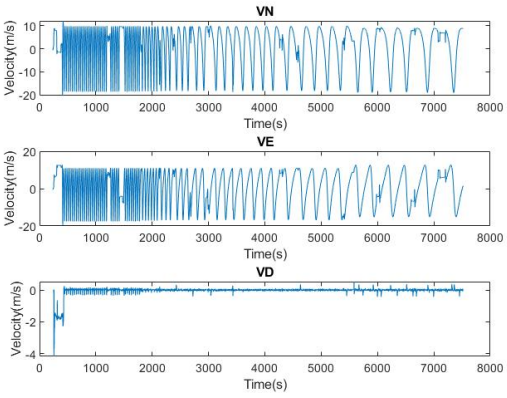


Figure D.4: Ground speed components (NED frame).

D.2 Results for static windows, kinematic model 1

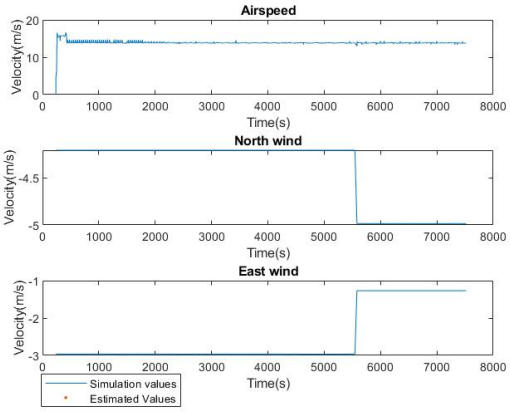


Figure D.5: Kinematic model 1, static window of $t_{window} = 10s$.

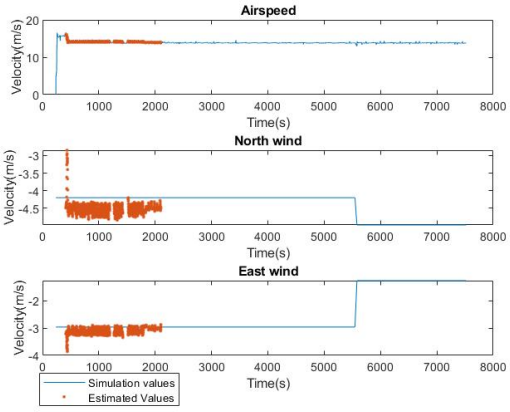


Figure D.6: Kinematic model 1, static window of $t_{window} = 20s$.

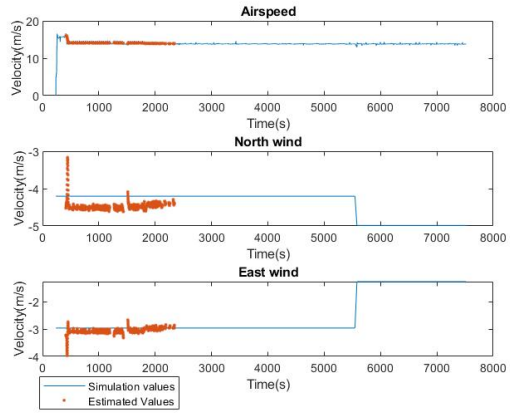


Figure D.7: Kinematic model 1, static window of $t_{window} = 30s$.

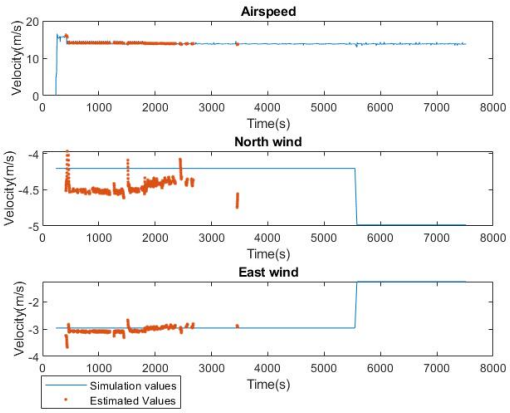


Figure D.8: Kinematic model 1, static window of $t_{window} = 40s$.

D.3 Results for static windows, kinematic model 2

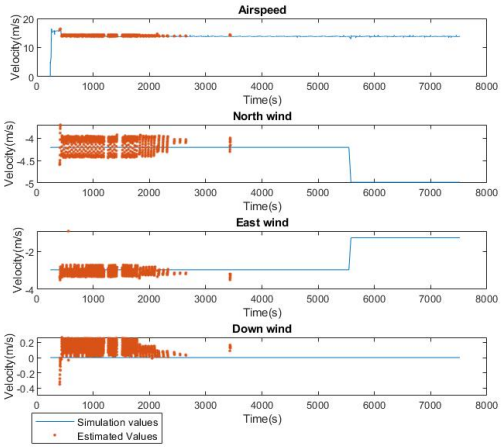


Figure D.9: Kinematic model 2, static window of $t_{window} = 10s$.

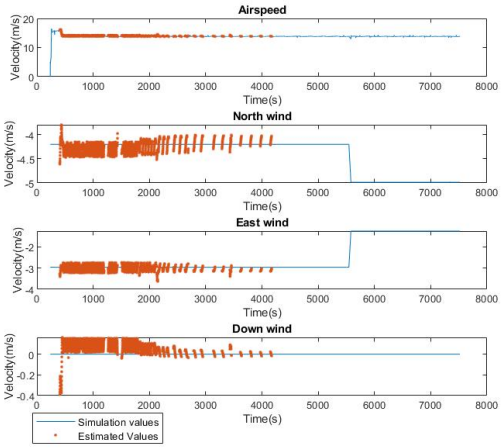


Figure D.10: Kinematic model 2, static window of $t_{window} = 20s$.

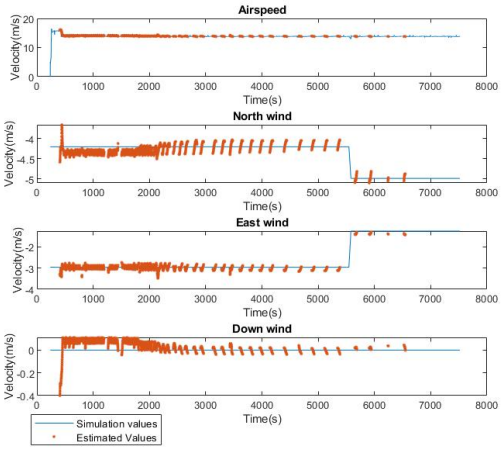


Figure D.11: Kinematic model 2, static window of $t_{window} = 30s$.

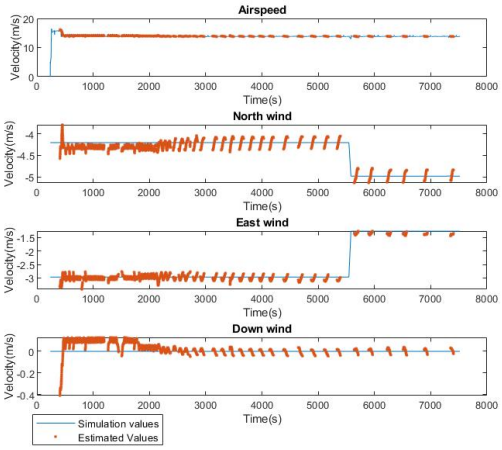


Figure D.12: Kinematic model 2, static window of $t_{window} = 40s$.

D.4 Comparison of models - uniform loiter maneuver

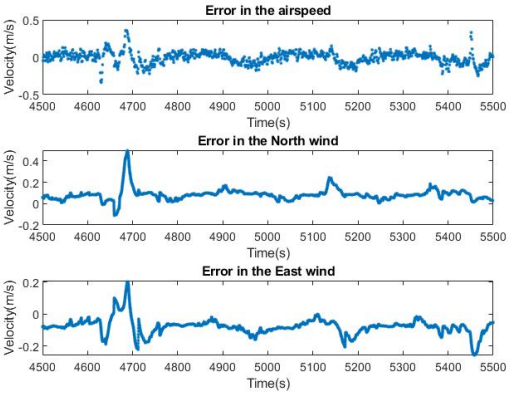


Figure D.13: Yaw maneuvering - First kinematic model.

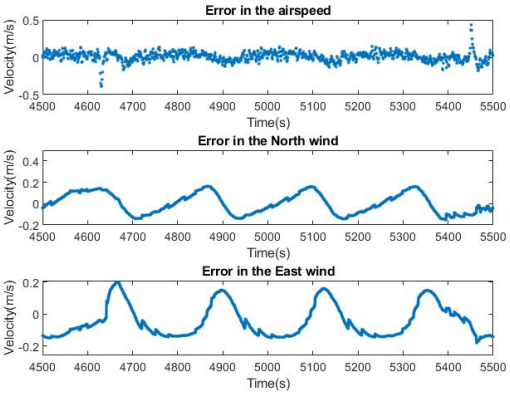


Figure D.14: Yaw maneuvering - Second kinematic model.

D.5 Comparison of models - variation of wind

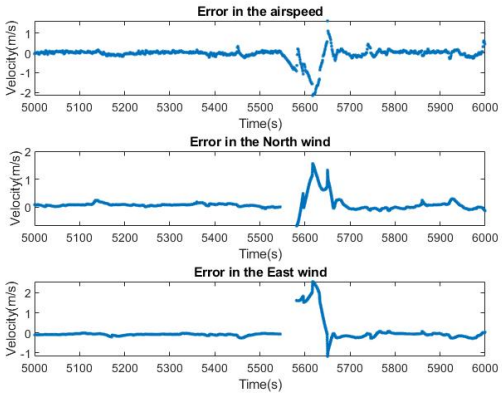


Figure D.15: Variation of wind - First kinematic model.

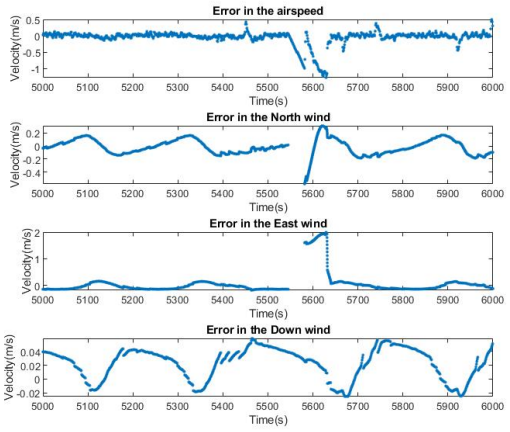


Figure D.16: Variation of wind - Second kinematic model.

Appendix E

Simulated autopilot flight 02

E.1 Flight characteristics

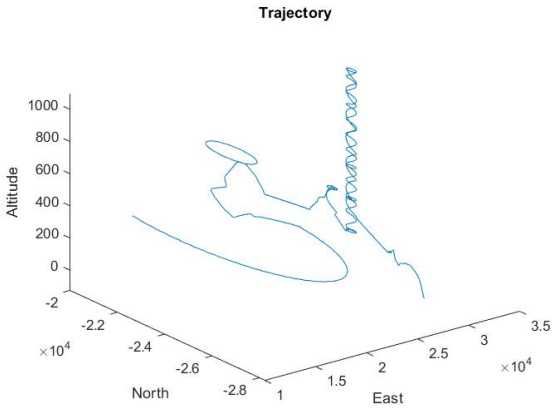


Figure E.1: Three dimensional trajectory.

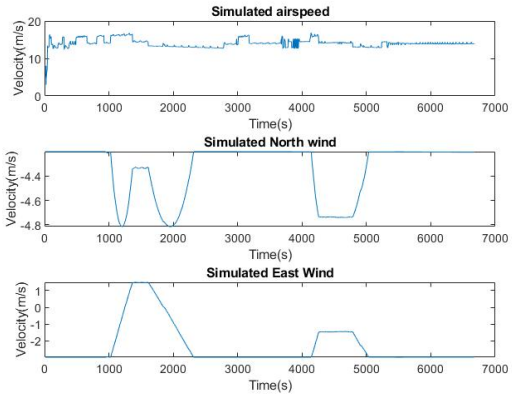


Figure E.2: Simulated airspeed and wind components.

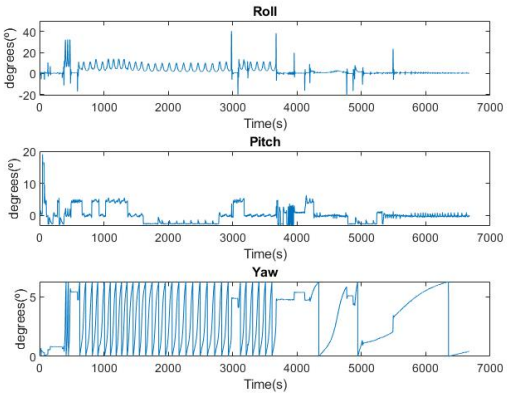


Figure E.3: Euler angles.

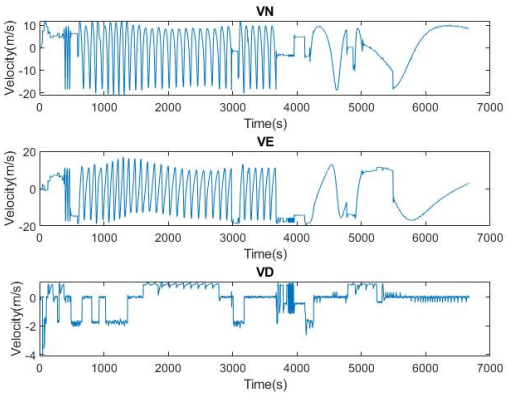


Figure E.4: Ground speed components (NED frame).

E.2 Comparison of models - Slow-varying altitude

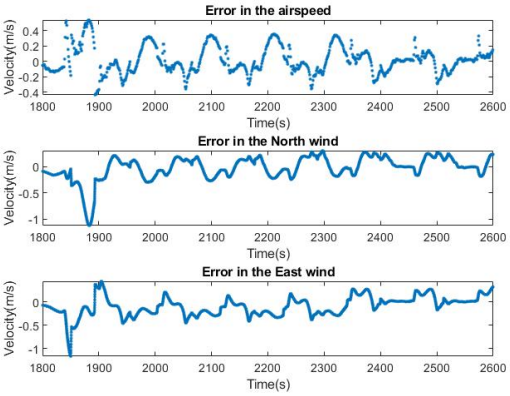


Figure E.5: Slow-varying altitude - First kinematic model.

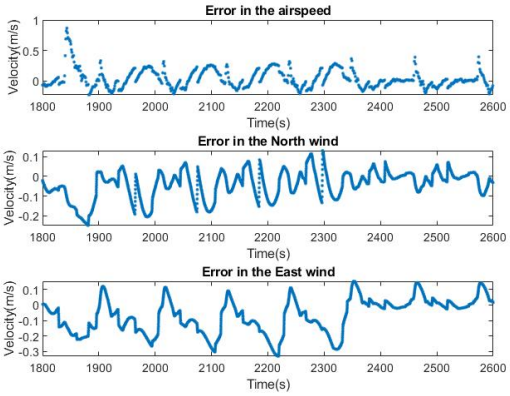


Figure E.6: Slow-varying altitude - Second kinematic model.