Unmanned Aerial Vehicle Positioning Using a Phased Array Radio and GNSS Independent Sensors

Carl Rapp
Abstract

This thesis studies the possibility to replace the global navigation satellite system (GNSS) with a phased array radio system (PARS) for positioning and navigation of an unmanned aerial vehicle (UAV). With the increase of UAVs in both civilian and military applications, the need for a robust and accurate navigation solution has increased. The GNSS is the main solution of today for UAV navigation and positioning. However, the GNSS can be disturbed by malicious sources, the signal can either be blocked by jamming or modified to give the wrong position by spoofing. Studies have been conducted to replace or support the GNSS measurements with other drift free measurements, e.g. camera or radar systems.

The position measurements from PARS alone is shown not to provide sufficient quality for the application in mind. The PARS measurements are affected by noise and outliers. Reflections from the ground makes the PARS elevation measurements unusable for this application. A root mean square error (RMSE) accuracy of 10 m for a shorter flight and 198 m for a longer flight are achieved in the horizontal plane. The decrease in accuracy for the longer flight is assumed to come from a range bias that increases with distance due to the flat earth approximation used as the navigation frame.

Positioning based on PARS aided with a filter and other GNSS independent sensors is shown to reduce the noise and remove the outliers. Five filters are derived and evaluated: a constant velocity extended Kalman filter (EKF), an inertial measurement unit (IMU) aided EKF, an IMU and barometer aided EKF; a converted measurements Kalman filter (CMKF) and a stationary Kalman filter (KF). The IMU and barometer aided EKF performed the best results with a RMSE of 8 m for a shorter flight and 106 m for a longer flight. The noise is significantly reduced compared to the standalone PARS measurements.

The conclusion is that PARS can be used as a redundancy system with the IMU and barometer aided EKF. If the EKF algorithm is too computational demanding, the simpler stationary KF can be motivated since the accuracy is similar to the EKF. The GNSS solution should still be used as the primary navigation solution as it is more accurate.
Acknowledgments

I would like to thank the people at UMS Skeldar, especially my supervisor Ola Härkegård for the input and support along the way.

At Linköpings university I would like to thank my supervisor Robin Forsling and examiner Gustaf Hendeby for the continuous feedback and interest in this thesis.

Lastly I would like to thank my girlfriend for all the encouragement and support along the way.
## Contents

<table>
<thead>
<tr>
<th>Notation</th>
<th>ix</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Purpose</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Problem Formulation</td>
<td>2</td>
</tr>
<tr>
<td>1.4 Limitations</td>
<td>2</td>
</tr>
<tr>
<td>1.5 Related Work</td>
<td>2</td>
</tr>
<tr>
<td>1.6 Thesis Outline</td>
<td>3</td>
</tr>
<tr>
<td>2 Theory and Background</td>
<td>5</td>
</tr>
<tr>
<td>2.1 Coordinate Frames</td>
<td>5</td>
</tr>
<tr>
<td>2.1.1 Earth-Centered Earth-Fixed Frame {e}</td>
<td>5</td>
</tr>
<tr>
<td>2.1.2 Geodetic Frame WGS84 {g}</td>
<td>5</td>
</tr>
<tr>
<td>2.1.3 Local Tangent-Plane Frame {t}</td>
<td>7</td>
</tr>
<tr>
<td>2.1.4 Body Frame {b}</td>
<td>7</td>
</tr>
<tr>
<td>2.2 Coordinate Transformations</td>
<td>8</td>
</tr>
<tr>
<td>2.2.1 Euler Angles</td>
<td>8</td>
</tr>
<tr>
<td>2.2.2 ECEF to NED Transformation</td>
<td>8</td>
</tr>
<tr>
<td>2.3 Sensors</td>
<td>9</td>
</tr>
<tr>
<td>2.3.1 Phased Array Radio</td>
<td>9</td>
</tr>
<tr>
<td>2.3.2 Accelerometer</td>
<td>12</td>
</tr>
<tr>
<td>2.3.3 Barometer</td>
<td>13</td>
</tr>
<tr>
<td>2.4 Bayesian Filtering</td>
<td>13</td>
</tr>
<tr>
<td>2.5 Kalman Filter</td>
<td>15</td>
</tr>
<tr>
<td>2.5.1 State-Space Model</td>
<td>15</td>
</tr>
<tr>
<td>2.5.2 Algorithm</td>
<td>15</td>
</tr>
<tr>
<td>2.5.3 Extended Kalman Filter</td>
<td>16</td>
</tr>
<tr>
<td>3 Phased Array Radio Navigation Analysis</td>
<td>17</td>
</tr>
<tr>
<td>3.1 Experimental Setup</td>
<td>17</td>
</tr>
<tr>
<td>3.1.1 Ground Panel Pose</td>
<td>17</td>
</tr>
<tr>
<td>3.1.2 Ground Truth Distance and Azimuth to UAV</td>
<td>18</td>
</tr>
</tbody>
</table>
3.1.3 Approximation to 2D .............................................. 18
3.2 Sensor Characteristics ............................................. 20
  3.2.1 Range ............................................................. 21
  3.2.2 Azimuth ......................................................... 21

4 Radio Link Based Navigation ........................................ 27
  4.1 Basic Algorithm ................................................... 27
  4.2 Kinematics ......................................................... 28
  4.3 IMU Aided Algorithm ............................................. 29
  4.4 IMU and Barometer Aided Algorithm ......................... 30
  4.5 Converted Measurements Kalman Filter ....................... 31
  4.6 Stationary Kalman Filter ...................................... 31

5 Experimental Evaluation .......................................... 33
  5.1 Experimental Setup ............................................. 33
    5.1.1 Time Synchronization and Delay ......................... 33
    5.1.2 Ground Truth ............................................... 33
  5.2 Scenarios ......................................................... 34
  5.3 Results .......................................................... 34
    5.3.1 RMSE Evaluation ........................................... 35
    5.3.2 Position Estimates ......................................... 36
    5.3.3 Step Response ............................................... 40
    5.3.4 Outlier Rejection ........................................... 40
    5.3.5 IMU and Barometer Aided Algorithm ................. 41
    5.3.6 Time Complexity ........................................... 45

6 Conclusion and Future Work .................................... 47
  6.1 Conclusion ...................................................... 47
  6.2 Future Work .................................................... 48

A Filter and Flight Parameters ................................... 51
  A.1 Flight Parameters ............................................... 51
  A.2 Filter Parameters ............................................... 51
    A.2.1 Basic Filter ............................................... 52
    A.2.2 IMU Filter .................................................. 52
    A.2.3 Barometer Filter ........................................... 52
    A.2.4 CMKF and stationary KF .................................. 53

Bibliography .......................................................... 55
### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviations</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARS</td>
<td>phased array radio system</td>
</tr>
<tr>
<td>DOA</td>
<td>direction of arrival</td>
</tr>
<tr>
<td>RTT</td>
<td>round trip time</td>
</tr>
<tr>
<td>UAV</td>
<td>unmanned aerial vehicle</td>
</tr>
<tr>
<td>ENU</td>
<td>east north up</td>
</tr>
<tr>
<td>NED</td>
<td>north east down</td>
</tr>
<tr>
<td>ECEF</td>
<td>earth-centered earth-fixed</td>
</tr>
<tr>
<td>RTK</td>
<td>real time kinematics</td>
</tr>
<tr>
<td>GNSS</td>
<td>global navigation satellite system</td>
</tr>
<tr>
<td>KF</td>
<td>Kalman filter</td>
</tr>
<tr>
<td>RWE</td>
<td>random walk error</td>
</tr>
<tr>
<td>MMSE</td>
<td>minimum mean-squared error</td>
</tr>
<tr>
<td>MAP</td>
<td>maximum a posterior</td>
</tr>
<tr>
<td>EKF</td>
<td>extended Kalman filter</td>
</tr>
<tr>
<td>STD</td>
<td>standard deviation</td>
</tr>
<tr>
<td>RMSE</td>
<td>root means square error</td>
</tr>
<tr>
<td>CV</td>
<td>constant velocity</td>
</tr>
<tr>
<td>CMKF</td>
<td>converted measurements Kalman filter</td>
</tr>
<tr>
<td>IMU</td>
<td>inertial measurement unit</td>
</tr>
</tbody>
</table>
This chapter introduces the studied problem. The chapter begins with a motivation of why the problem is studied and how the solution would help in the future development of unmanned aerial vehicle (UAV). Then the purpose of the study is presented, followed by a specified problem formulation and limitations of the study. The chapter ends with related work in the subject and an outline of the following chapters in the thesis.

1.1 Motivation

The UAVs of today are heavily dependent on global navigation satellite systems (GNSS) for positioning. The benefit of GNSS is the high accuracy, where a high-end GNSS receiver can have an accuracy of under one meter [9]. But there are some drawbacks with GNSS, malicious sources can interfere with the true GNSS signal and disturb the accuracy. Some examples of malicious sources are jamming and spoofing which are described in [15] and [23]. It is also possible that the GNSS signals are low in specific areas. If an UAV loses the GNSS signal, it will start to dead reckoning using the onboard sensors, e.g. inertial measurement unit (IMU), magnetometer and pressure sensors. The problem with dead reckoning is that the calculated position drifts over time. Some sort of drift free measurements need to be implemented in the system. In this thesis a phased array radio system (PARS) is used as the drift free measurements to calculate the position of the UAV. The communication link is used for streaming video and data to the ground station but it can also be used to get bearing, elevation and distance to the UAV. The complete system is described in Section 2.3.
1.2 Purpose

The purpose of this thesis is to study the quality of the PARS measurements and how other GNSS independent sensors can be fused with the PARS measurements to get a more accurate position estimate.

1.3 Problem Formulation

The problem is divided into the following three questions that shall be answered:

- How good are the measurements from PARS compared to GNSS measurements?
- What sensors and states should be used for best positioning regarding complexity of the filter?
- Is it possible to navigate the UAV based on measurements from PARS?

1.4 Limitations

Due to time constraints of the thesis and limited experiments, some limitations are added.

A direction of arrival (DOA) algorithm needs to be used to get the bearing and elevation angles to the target from a PARS. This is not studied in this thesis, an existing program is used to calculate those angles. The program is considered a black box with no insight.

The study is limited to offline testing in MATLAB with recorded flight data. The time delay from the pre-processing (DOA calculations) and sending of data from PARS to the UAV is neglected.

Due to the limited time for this thesis, the attitude of the UAV assumes to be known along with the accelerometer bias in the z-axis.

1.5 Related Work

The thesis is inspired by work done by the Norwegian University of Science and Technology (NTNU). Three studies with the same PARS sensor and a fixed-wing UAV are presented in this section.

In paper [3] the PARS measurements are used to get the absolute position for a fixed-wing UAV. A barometer is added to increase the vertical accuracy.

A nonlinear observer is derived in [4] that estimates position, velocity and attitude of a fixed-wing UAV. The observer fuses range and bearing measurements from the PARS with the measurements from an on-board inertial measurement unit, a magnetometer and a barometer. An experiment with the fixed-wing UAV of about 35 minutes of flight time with a maximum distance of 5.35 km from the ground station showed a position accuracy of 26.3 m. A real-time kinematic (RTK) GNSS solution is used as ground truth.
The PARS observer mentioned in paper [4] is extended with a spoofing detector in [5]. A spoofing attack can be detected if the estimates from the RTK GNSS and the estimates from the PARS observer start to deviate. If spoofing is detected the estimates from the observer are used for navigation.

The Northern Research Institute (NORUT) is also working with the same PARS sensor, their results and experiences were discussed over a meeting.

### 1.6 Thesis Outline

Chapter 2 introduces relevant theory and background needed for the thesis.

Chapter 3 evaluates the PARS measurements compared to the GNSS positions.

Chapter 4 introduces the derived filters.

Chapter 5 introduces the evaluation of the filters presented in Chapter 4. First the experimental setup is explained, followed by the result from the experiments.

Chapter 6 aims to answer the questions stated in Section 1.3 using the results presented in Chapter 5. The chapter also includes future work that is outside the scope of this thesis but is interesting for future implementation.
This chapter covers basic theory and background needed for the rest of the thesis. First
the relevant coordinate frames are introduced followed by transformations between coor-
dinate frames. Thereafter follows a short description of the system and the sensors used
in this thesis. Finally, a brief summary of Bayesian filtering, Kalman filter and extended
Kalman filter.

2.1 Coordinate Frames

In navigation it is essential to describe an object in terms of position, motion and orien-
tation. To describe those, a coordinate frame needs to be used. A frame is defined by a
rigid body that can be used to establish directions and distances. A coordinate system is
defined in this frame to describe those directions and distances. The different frames used
in this thesis are presented below.

2.1.1 Earth-Centered Earth-Fixed Frame {e}

The earth-centered earth-fixed (ECEF) frame is described in Figure 2.1. The coordinate
system is fixed in the earth with its origin in the center of the earth. Further, the z-axis
is pointing from the center of the earth to the true north pole, the x-axis is pointing from
the center of the earth to the intersection between the equator and the IERS Reference
Meridian (IRM) and the y-axis is pointing from the center to the intersection between the
equator and the 90 degree east meridian.

2.1.2 Geodetic Frame WGS84 {g}

A Geodetic frame uses an earth model to determine the position of an object relative to
the surface of earth instead of the center of earth as in ECEF. The earth model used in this
thesis is the WGS84, it is an ellipsoid model that describes the surface of the earth and is used by the NAVSTAR global positioning system (GPS). The ellipsoid model is defined by the equator radius $R_0$ and the polar radius $R_p$. The model is also defined by $R_0$ and the eccentricity $e$ or the flattening $f$ of the ellipsoid. The parameters for WGS84 are defined in Table 2.1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_0$</td>
<td>6 378 137.00000m</td>
</tr>
<tr>
<td>$R_p$</td>
<td>6 356 752.31425m</td>
</tr>
<tr>
<td>$f$</td>
<td>1/298.257223563</td>
</tr>
<tr>
<td>$e$</td>
<td>0.0818191908425</td>
</tr>
</tbody>
</table>

The coordinates for the WGS84 are described in longitude $\lambda$, latitude $\phi$ and the geodetic height $h$. The coordinate system is also described in Figure 2.1. According to [6, eq. (2.22)] the point

$$P_g = \begin{pmatrix} \lambda \\ \phi \\ h \end{pmatrix},$$

(2.1)

can be transformed to the ECEF frame as

$$P_e = \begin{pmatrix} x_e \\ y_e \\ z_e \end{pmatrix} = \begin{pmatrix} (N_e + h) \cos \phi \cos \lambda \\ (N_e + h) \cos \phi \sin \lambda \\ (N_e(1 - e^2) + h) \sin \phi \end{pmatrix},$$

(2.2)
2.1 Coordinate Frames

Figure 2.2: Body frame.

where

$$N_c = \frac{R_0}{\sqrt{1 - e^2 \sin^2 \phi}}.$$  \hfill (2.3)

2.1.3 Local Tangent-Plane Frame \{t\}

The local tangent-plane is defined as a north-east-down (NED) or an east-north-up (ENU) coordinate frame depending on the application. The NED frame is described in Figure 2.1. The origin is fixed relative the surface of earth, the \(z\)-axis is pointing in the direction of the negative surface normal to the plane, the \(x\)-axis is pointing in the north direction and the \(y\)-axis is pointing to east. The ENU frame is also a right-handed coordinate system with the \(z\)-axis pointing in the direction of the surface normal to the plane. To keep it a right-handed coordinate system the \(x\)-axis is directed to east while the \(y\)-axis is directed to north.

2.1.4 Body Frame \{b\}

To determine the orientation and position of an object relative a navigation frame a fixed coordinate system needs to be introduced in the object. The body frame is defined as in Figure 2.2 with the \(x\)-axis in the forward direction, the \(y\)-axis to the right and the \(z\)-axis is pointing down. The origin is placed in the center of mass of the object.
2.2 Coordinate Transformations

This section describes the coordinate transformation matrices used in the thesis. The notation used for describing the motion is

\[ \mathbf{x}_\beta, \]  

(2.4)

where \( \beta \) is the frame which the motion is represented in and \( \mathbf{x} \) is the state vector containing, e.g. positions, velocity and acceleration. A coordinate transformation matrix uses the following notation

\[ \mathbf{x}_t = \mathbf{R}_b^t \mathbf{x}_b + \mathbf{t}_b^t, \]  

(2.5)

where \( \mathbf{x}_t \) is some state vector for an object in the NED frame, \( \mathbf{R}_b^t \) is the rotation matrix from the body frame to the NED frame, \( \mathbf{x}_b \) is the state vector for the object in the body frame and \( \mathbf{t}_b^t \) is called the translation vector which is the vector from the body frame origin to the local NED frame origin.

2.2.1 Euler Angles

Euler angles are an intuitive way to describe the attitude of a body relative a fixed reference frame. The Euler angles

\[ \boldsymbol{\Phi} = \begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix} \]  

(2.6)

are defined in Figure 2.2, where the yaw rotation is denoted \( \psi \), the pitch rotation is denoted \( \theta \) and the roll rotation is denoted \( \phi \). The pitch angle is defined in the range \(-\frac{\pi}{2} < \theta \leq \frac{\pi}{2}\) to get rid of duplicated sets of angles that represent the same attitude. The Euler angles are also affected by a singularity when \( \theta = \pm \frac{\pi}{2} \), which results in the roll and yaw angle becoming indistinguishable. To avoid the singularity and duplicated angles the unit quaternions representation of the attitude can be used instead. This is not necessary in this work since the UAV will not exceed the dangerous pitch angle. Euler angles can be used to transform one vector represented in one frame to another. The rotation matrix in [22, eq. (1.3-10)] can be transposed to get the rotation matrix from body frame to the NED frame as

\[ \mathbf{R}_b^t = \begin{pmatrix} c_\theta c_\psi & -c_\phi s_\psi + s_\phi s_\theta c_\psi & s_\phi s_\psi + c_\phi s_\theta c_\psi \\ c_\theta s_\psi & c_\phi c_\psi + s_\phi s_\theta s_\psi & -s_\phi c_\psi + c_\phi s_\theta s_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix}, \]  

(2.7)

where \( s_\theta = sin(\theta) \) and \( c_\theta = cos(\theta) \).

2.2.2 ECEF to NED Transformation

The transformation between the ECEF to the NED frame is defined according to [6, Section 2.3.2.2] as

\[ \mathbf{x}_t = \mathbf{R}_e^t (\mathbf{x}_e - \mathbf{x}_{e,ref}), \]  

(2.8)
2.3 Sensors

The UA V used in this study is the Skeldar V-200 developed by UMS Skeldar. Detailed specification can be found in [20]. It is a versatile UA V designed for civilian and military missions. It is a remotely piloted aircraft system controlled from a ground station. The payload can be configured to a wide range of sensors depending on the mission. The Skeldar V-200 sensor setup available in this thesis can be seen in Figure 2.3. The sensors relevant here are described in more detail in the following sections.

2.3.1 Phased Array Radio

The PARS consists of two antennas, one at the ground station (CRE2-189) and one mounted on Skeldar V-200 (CRE2-179-UA V). CRE2-189 and CRE2-179-UA V are using phased array technology for high data rate communication over large distances. The advantage of this system is the possibilities to get the relative angles and range to the UAV. To get decent measurements the UAV needs to be in front of the ground station antenna. The angles are calculated using the direction of arrival (DOA) from the incoming radio signal. The DOA is well studied with a variety of different algorithms to solve the problem. This is outside the scope for this thesis but is studied in e.g. [17]. The range

\[
R^t_e = \begin{pmatrix}
-\sin \phi_{ref} \cos \lambda_{ref} & -\sin \phi_{ref} \sin \lambda_{ref} & \cos \phi_{ref} \\
-\sin \lambda_{ref} & \cos \lambda_{ref} & 0 \\
-\cos \phi_{ref} \cos \lambda_{ref} & -\cos \phi_{ref} \sin \lambda_{ref} & -\sin \phi_{ref}
\end{pmatrix},
\]

where \( \phi_{ref} \) and \( \lambda_{ref} \) are the latitude and longitude corresponding to \( x_{e,ref} \).
measurement is calculated by

\[ r = (t_r - t_c)c, \]  

(2.10)

where \( t_r \) is the round trip time measured at the ground station, \( t_c \) is the internal computational time for the system and \( c \) is the speed of light. All calculations are made at the ground station. The setup can be seen in Figure 2.4. The measurements from PARS are non-uniform sampled which means that the time between each sample varies.

**Sensor model**

The measurements from the PARS are in spherical coordinates relative the body coordinate system for the ground panel. The relative angles for the ground panel are defined in Figure 2.5 where the \( x \)-axis is pointing out from the panel, the \( y \)-axis is pointing to the left and the \( z \)-axis is pointing in the down direction of the panel. The measurements are defined as

\[ \rho_b = \rho_u + e_{\rho}, \]  

(2.11)

\[ \theta_b = \theta_u + e_{\theta}, \]  

(2.12)

\[ \phi_b = \phi_u + e_{\phi}. \]  

(2.13)

Where \( \rho_b \) is the measured range from PARS to the UAV, \( \theta_b \) is the measured azimuth (horizontal angle), \( \phi_b \) is the measured elevation (vertical angle). \( \rho_u, \theta_u \) and \( \phi_u \) are the true values for range, azimuth and elevation in the body frame. \( e_{\rho}, e_{\theta} \) and \( e_{\phi} \) are the errors introduced in the measurements. The pose for the ground panel is used to transform the
coordinates to the local NED frame as

\begin{align*}
\rho_t &= \rho_b, \quad (2.14) \\
\theta_t &= \text{mod}(\theta_b + \theta_0, 2\pi), \quad (2.15) \\
\phi_t &= \phi_b + \phi_0, \quad (2.16)
\end{align*}

where \(\theta_0\) and \(\phi_0\) are the azimuth and elevation of the ground panel pose. The modulus operator is used to ensure that \(0 \leq \theta_t < 2\pi\). The translation \(t_b^t\) mentioned in (2.5) is not used because the local NED frame is initialized in the origin of the PARS. The spherical measurements in (2.14) can be transformed to Cartesian positions using

\begin{align*}
\bar{\rho} &= \rho_t \cos \phi_t, \quad (2.17) \\
x_t &= \bar{\rho} \cos(\theta_t), \quad (2.18) \\
y_t &= \bar{\rho} \sin(\theta_t), \quad (2.19) \\
z_t &= -\rho_t \sin(\phi_t), \quad (2.20)
\end{align*}

which gives the final sensor model

\[
y = (\rho_t, \theta_t, \phi_t)^T = \begin{pmatrix}
\sqrt{x_t^2 + y_t^2 + z_t^2} \\
\arctan2(y_t, x_t) \\
\arctan\left(-\frac{z_t}{\sqrt{x_t^2 + y_t^2}}\right)
\end{pmatrix} + e. \quad (2.21)
\]

The \text{atan2} function is the four quadrant arctangens defined in the range \(0 < \text{atan2}(y_t, x_t) \leq \pi/2\).
2\pi as
\[
\text{atan2}(y_t, x_t) = \begin{cases} 
\arctan(y_t/x_t), & x_t > 0 \text{ and } y_t \geq 0 \\
\arctan(y_t/x_t) + 2\pi, & x_t > 0 \text{ and } y_t < 0 \\
\pi + \arctan(y_t/x_t), & x_t < 0 \\
\frac{\pi}{2}, & y_t > 0 \text{ and } x_t = 0 \\
\frac{3\pi}{2}, & y_t < 0 \text{ and } x_t = 0 
\end{cases}
\] (2.22)

\subsection{2.3.2 Accelerometer}

The accelerometer measures specific force in a specific axis. The accelerometer used in Skeldar V-200 is a 3-axis accelerometer that is fixed in the body frame described in Section 2.1.4. The gravitational force is always acting on the accelerometer. This is used together with the gyro measurements to get a good attitude estimate if the UAV is not affected by other accelerations or if they are known. Velocity and position can be estimated by transforming the measurements to the navigation frame and then integrate the acceleration. These estimates tend to drift over time, the error is called random walk error (RWE) and is further explained in the next section.

\subsubsection{Error Characteristics}

As mentioned in [11, Chapter 4], the accelerometer is affected by four types of errors: bias, scale factor and cross-coupling errors, and random measurement noise. The scale factor is the error between input and output from the sensor, which is proportional to the true specific force. The cross-coupling error arises from misalignment of the sensor axes when they are not perfectly orthogonal to each other. According to [11], the scale factor and the cross-coupling error are unitless and often expressed in parts per million (ppm). For most inertial sensors, these errors are in the magnitude of 100 - 1000 ppm. The bias error can be split into two biases, a static and a dynamic bias. The static bias changes on start up but is constant the whole operating time. The dynamic bias changes over time, this is related to e.g. temperature changes or mechanical stress on the sensor. The RWE for the accelerometer is the error that occurs when the acceleration measurements are integrated to get the inertial velocity. The random measurement noise is defined as
\[
e_{acc} \sim \mathcal{N}(0, \sigma_w),
\] (2.23)
which gives the RWE for one sample
\[
\text{RWE} = \sqrt{T_s \sigma_w},
\] (2.24)
where \(T_s\) is the sampling interval and \(\sigma_w\) is the standard deviation of the accelerometer noise.

\subsubsection{Sensor Model}

The sensor model for the accelerometer is
\[
\tilde{f}_b = f_b + b_{acc} + e_{acc},
\] (2.25)
where \(\tilde{f}_b\) is the measured specific force vector, \(f_b\) is the true value vector, \(b_{acc}\) is the bias vector and \(e_{acc}\) is the noise error vector. For simplicity, the scaling error and cross-
coupling error are included in the bias error.

### 2.3.3 Barometer

The barometer is used to measure ambient pressure, which can be used to estimate the altitude of the UAV. The relation between pressure and altitude is calculated, assuming the standard atmospheric model, as

$$h_g = \frac{T_s}{k} \left( 1 - \frac{p}{p_s} \right) + h_s,$$

(2.26)

where $h_g$ is the geodetic height, $R = 287.1$ J kg$^{-1}$ K$^{-1}$ is the gas constant, $k = 0.0065$ K m$^{-1}$ is the atmospheric temperature gradient and $g_0 = 9.80665$ m s$^{-2}$ is the average surface acceleration [11]. $p_s$ and $T_s$ are the surface pressure and temperature, respectively, and $h_s$ is the geodetic height where $p_s$ and $T_s$ are measured. For a differential barometry those values are updated at a reference station but for a stand-alone barometry the ISA standard is used [14]. According to the ISA standard $p_s = 101.325$ kPa, $T_s = 288.15$ K and $h_s = 0$.

#### Sensor Model

The height is modeled as

$$\tilde{h}_g = h_g + b_{baro} + e_{baro},$$

(2.27)

where $\tilde{h}_g$ is the transformed height from the measured ambient pressure with the ISA standard and $h_g$ is the true geodetic height. The bias, $b_{baro}$, arises from the difference in true and modeled atmospheric temperature and pressure. The error, $e_{baro}$, comes from random noise in the pressure measurement.

### 2.4 Bayesian Filtering

A non-stationary parameter $x_k$ affected by noise can be estimated by assuming restrictions described by a dynamic and observation model. The dynamic and observation model can be expressed as the probability density functions

$$p(x_{k+1}|x_k),$$

(2.28)

$$p(y_k|x_k),$$

(2.29)

respectively, where $x_k$ is a state at time $k$ and $y_k$ is a measurement at time $k$. The Bayesian approach to filtering is to estimate the non-stationary parameter $x_k$ by computing the distributions $p(x_k|y_{1:k})$ and $p(x_{k+1}|y_{1:k})$ with the model described in (2.28) [13]. The posterior distribution is obtained using Bayes’ rule, i.e.

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})},$$

(2.30)
Figure 2.6: Posterior and prior distribution for state $x$

given the prior distribution $p(x_k|y_{1:k-1})$ and the likelihood of the obtained measurement $p(y_k|x_k)$. Note that

$$p(y_k|y_{1:k-1}) = \int_{\mathbb{R}} p(y_k|x_k) p(x_k|y_{1:k-1}) \, dx_k$$ (2.31)

is the normalization constant. The prediction $p(x_k|y_{1:k-1})$ can be expressed by the total law of probability as

$$p(x_{k+1}, x_k|y_{1:k}) = p(x_{k+1}|x_k) p(x_k|y_{1:k}),$$ (2.32)

by integrating both sides with respect to $x_k$

$$p(x_{k+1}|y_{1:k}) = \int_{\mathbb{R}} p(x_{k+1}|x_k) p(x_k|y_{1:k}) \, dx_k.$$ (2.33)

Equation (2.30) is called the measurement update and (2.33) is called the time update. The measurement update can be seen as a correction step that reduces the uncertainty in the estimate at time $k$. The time update is the prediction step where the dynamic model in (2.28) is used to predict the estimate at time $k+1$ with increasing uncertainty. Figure 2.6 shows the posterior and prior for the state $x$.

According to [7], two potential criteria for measuring optimality is the minimum mean-squared error (MMSE) and maximum a posterior (MAP). They are defined as

$$\hat{x}^{\text{MMSE}}_{k|k} = \mathbb{E}[x_k|y_{1:k}] = \int_{\mathbb{R}} x_k p(x_k|y_{1:k}) \, dx_k,$$ (2.34)

$$\hat{x}^{\text{MAP}}_{k|k} = \arg \max_{x_k} p(x_k|y_{1:k}),$$ (2.35)

and they are equal if the distributions are linear Gaussian.
2.5 Kalman Filter

In 1960, R.E. Kalman invented the Kalman filter (KF) [16]. The filter computes the posterior (2.30) exactly for a Gaussian linear system [13, Chapter 7]. The Apollo space mission program used the KF in the navigation system and is still a frequently used algorithm [10].

2.5.1 State-Space Model

The discrete state-space model used to compute the KF is

\[
\begin{align*}
    x_{k+1} &= Fx_k + Gu_k + w_k, \\
    y_k &= Hx_k + Ju_k + e_k,
\end{align*}
\]

where \( w_k \) is the process noise and \( e_k \) is the measurement noise, assumed to be Gaussian with zero mean. The state space-model represents the system’s relation from input to output.

2.5.2 Algorithm

The KF algorithm on standard algebraic form is presented in Algorithm 2.1. The standard form is using the Riccati equation to solve the error covariance in the time update [12]. The filter is initialized with \( \hat{x}_{1|0} = \mathbb{E}(x_0) \) and \( P_{1|0} = \text{Cov}(x_0) \). \( \epsilon_k \) is called the innovation.

**Algorithm 2.1 Kalman Filter**

For the discrete state space model mentioned in (2.36), the best linear unbiased filter is given by the following recursions:

**Measurement update.**

\[
\begin{align*}
    S_k &= H_k P_{k|k-1} H_k^T + R_k \\
    K_k &= P_{k|k-1} H_k^T S_k^{-1} \\
    \epsilon_k &= y_k - H_k \hat{x}_{k|k-1} - D_k u_k \\
    \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k \epsilon_k \\
    P_{k|k} &= P_{k|k-1} - P_{k|k-1} H_k^T S_k^{-1} H_k P_{k|k-1}.
\end{align*}
\]

**Time update.**

\[
\begin{align*}
    \hat{x}_{k+1|k} &= F_k \hat{x}_{k|k} + G_k u_k, \\
    P_{k+1|k} &= F_k P_{k|k} F_k^T + Q_k
\end{align*}
\]

\( S_k \) is the innovation covariance and \( K_k \) is the Kalman gain. The process covariance is defined as \( Q_k = \text{Cov}(w_k) \) and the measurement covariance is defined as \( R_k = \text{Cov}(e_k) \).
### 2.5.3 Extended Kalman Filter

The Extended Kalman filter (EKF) was invented to solve the estimation problem for non-linear systems

\[
x_{k+1} = f(x_k, u_k, w_k),
\]
\[
y_k = h(x_k, u_k, e_k).
\]

The EKF is using the Taylor expansion to linearize the nonlinear models \( f \) and \( h \) and then apply the KF algorithm described in Algorithm 2.1. The EKF algorithm is presented in Algorithm 2.2.

**Algorithm 2.2 Extended Kalman Filter**

The EKF using the first order Taylor expansion around the current state to linearize the model described in (2.39) and (2.40) with additive noises \( v_k \) and \( e_k \) given by the following recursions initialized with \( \hat{x}_{1|0} \) and \( P_{1|0} \):

**Measurement update.**

\[
S_k = h'(\hat{x}_{k|k-1})P_{k|k-1}(h'(\hat{x}_{k|k-1}))^T + R_k
\]
\[
K_k = P_{k|k-1}(h'(\hat{x}_{k|k-1}))^T S_k^{-1}
\]
\[
\epsilon_k = y_k - h(\hat{x}_{k|k-1})
\]
\[
\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \epsilon_k
\]
\[
P_{k|k} = P_{k|k-1} - P_{k|k-1}(h'(\hat{x}_{k|k-1}))^T S_k^{-1} h'(\hat{x}_{k|k-1}) P_{k|k-1}
\]

**Time update.**

\[
\hat{x}_{k+1|k} = f(\hat{x}_{k|k})
\]
\[
P_{k+1|k} = f'(\hat{x}_{k|k})P_{k|k}(f'(\hat{x}_{k|k}))^T + Q_k
\]
This chapter evaluates the PARS measurements. First is the experimental setup explained, thereafter is the sensor characteristics evaluated.

### 3.1 Experimental Setup

Two flights are used to determine the quality and characteristics of the PARS measurements. Figure 5.1 and 5.2 show the two flight trajectories from the GNSS solution implemented in Skeldar V-200. The GNSS solution is used as ground truth for this analysis. The GNSS measurements are first transformed from WGS84 to ECEF as in (2.2). Thereafter the transformation given in Section 2.2.2 is used to get the position in the local NED frame with origin in the PARS ground panel. Both flights are performed over water.

#### 3.1.1 Ground Panel Pose

The azimuth and the position for the PARS ground panel is measured with a compass and a GPS. The azimuth for each flight can be seen in Table 3.1. The local NED frame described in Section 2.1.3 is used as reference frame with origin in the PARS ground panel. The measured pose for the PARS ground panel gave small offset compared to the GNSS measurement. Therefore, the azimuth for the ground panel is adjusted to reduce the offset. The ground panel pose can also be estimated if the start of the UAV is in front of the ground panel and with good GNSS coverage, the method is described in [21].

*Table 3.1: Azimuth for the ground panel.*

<table>
<thead>
<tr>
<th>Flight</th>
<th>Measured azimuth</th>
<th>Adjusted azimuth</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>273°</td>
<td>274.25°</td>
</tr>
<tr>
<td>2</td>
<td>234°</td>
<td>234.8°</td>
</tr>
</tbody>
</table>
3.1.2 Ground Truth Distance and Azimuth to UAV

The range measurement from PARS is compared against the calculated distance from the ground truth data. The ground truth distance to the UAV is calculated as

\[ GNSS_{distance} = \sqrt{(p_n)^2 + (p_e)^2 + (p_d)^2}, \quad (3.1) \]

where \( p_n, p_e \) and \( p_d \) are the ground truth positions from the GNSS solution in the local NED frame.

The ground truth azimuth used for validation of the PARS azimuth measurement is defined as

\[ GNSS_{azimuth} = \text{atan2}(p_e, p_n), \quad (3.2) \]

where \( \text{atan2} \) is defined in (2.22).

3.1.3 Approximation to 2D

Figure 3.3 shows the altitude calculated from PARS compared to the GNSS altitude. The sudden drops in altitude are assumed to come from reflections of the radio beam in the ground that the ground panel mistakes for the real signal. The poor altitude estimation from PARS will therefore not be used in this thesis. In this section the problem will be approximated as a two-dimensional problem with \( \rho \approx \bar{\rho} \) in (2.17). The introduced error from this approximation can be seen in Figure 3.4. A barometer is added in Section 4.4 to estimate the altitude of the UAV.
Figure 3.3: Flight 1 Altitude from PARS compared to the GNSS solution

Figure 3.4: Approximation error from $\rho \approx \bar{\rho}$. 
3.2 Sensor Characteristics

The position estimates from PARS are illustrated in Figures 3.5 and 3.6. The PARS estimates in Flight 1 seems to be able to follow the ground truth trajectory except for some peaks that occur at approximately the same distance from the PARS ground station. This can be explained by interference from reflected signals that for some distance are 180° shifted compared to the real signal. This is called multipath propagation and it is hard to avoid [1]. One solution is to use multiple PARS sensors at different heights and distances to the target. Another solution is to accept the multipath propagation and reject the measurements if they are affected by it. The problem with this solution is to determine when the multipath propagation is present. This is done by measuring the signal strength and to track sudden drops in the signal.

The same behavior can be seen in flight 2 but with some bigger peaks when the UAV is returning home. The placement of the PARS antenna on the UAV is assumed to be the problem. The antenna on the UAV is placed in the back of the UAV with a gimbal camera mounted in front of it, which is assumed to effect the signal.

The root mean square error (RMSE) is a common error metric used for position accuracy of a estimate or a measurement. It is defined as

\[
RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^{N} (x_k^0 - \hat{x}_k)^2}
\]

(3.3)

where \(x_k^0\) is the ground truth at time \(k\) and \(\hat{x}_k\) is the estimated state.
3.2 Sensor Characteristics

The RMSE for the norm, east and north positions along with the standard deviation (STD) for the norm positions are presented in Figure 3.7. It can be seen that the performance of the PARS is considerably worse for the longer flight, the large number of outliers seen in Figure 3.6 and the range bias mentioned in the next section are assumed to be the reason for this.

3.2.1 Range

Figure 3.8 presents the error in range from PARS compared to the range calculated from the GNSS solution in (3.1). The figure indicates that it is some bias in the range measurement where the range error goes from positive 5 meters to negative 50 meters when the distance between the ground panel and UAV increases. The reason for this bias is not certain, it may be due to range bias in the PARS sensor or the calculated ground truth distance is not good enough to use in this experiment. The bias can lead to a offset in the position estimates compared to the GNSS positions. The offset is accepted as it is a slow change in position compared to the noise present in the measurement.

Figure 3.9 shows the error distribution for the range measurements from flight 1. It looks like it is almost normal distributed with a standard deviation of $\sigma_\rho \approx 4$ m. The range measurement from PARS is quantized with 3.5 m, this can be seen in Figure 3.10.

3.2.2 Azimuth

Figure 3.11 shows the error in azimuth from the PARS azimuth compared to GNSS $S_{azimuth}$. The error is greater when the UAV is closer to the ground station, this is probably due to the circular motion of the UAV in the start of the flight which seems to be harder for the
**Figure 3.7:** RMSE and STD for flight 1 and 2.

**Figure 3.8:** PARS range error for flight 2.
3.2 Sensor Characteristics

Figure 3.9: Flight 1 range error distribution.

Figure 3.10: Quantization of the range measurements from PARS
PARS to follow. The two distributions in Figures 3.12 and 3.13 show that the error seems to be normal distributed with a small variance. However, the error characteristics for the azimuth is hard to determine when the UAV is moving. A better experiment would be to do a static test where the ground truth azimuth is known, due to time constraints this is not tested. The standard deviation is estimated to be $\sigma_{\theta} \approx 0.31^\circ$ from the results in this thesis along with the results from [4, Section 6.6.4].

**Figure 3.11: Flight 1 azimuth error.**
Figure 3.12: Azimuth error distribution for the time segment $T = 2000 \text{ s} - 2200 \text{ s}$ from Figure 3.11.

Figure 3.13: Azimuth error distribution for the time segment $T = 1400 \text{ s} - 1700 \text{ s}$ from Figure 3.11.
This chapter derives the five filters used in this thesis. Firstly, three EKF filters are presented. The EKF is used because of the nonlinear sensor model for PARS and it is similar to the filter used by the UAV today with GNSS measurements instead of PARS. This is followed by a converted measurements Kalman filter (CMKF) that is transforming the PARS measurements to pseudo-measurements in Cartesian positions before filtering. Lastly, a stationary KF is derived with the same pseudo-measurements.

4.1 Basic Algorithm

Motion models are essential in target tracking where limited observations of a target can be solved with a good motion model [19]. The basic algorithm is using a constant velocity (CV) model to define the system dynamics because of the simplicity and the linear properties. The state vector is

\[
x = \begin{pmatrix} p_n \\ p_e \\ v_n \\ v_e \end{pmatrix},
\]

which according to [13] gives the system dynamics

\[
x_{k+1} = \begin{pmatrix} I_2 & T I_2 \\ 0_2 & I_2 \end{pmatrix} x_k + \begin{pmatrix} T^2 I_2 \\ T I_2 \end{pmatrix} w_k,
\]

where \( T \) is the update time for the filter, \( I_2 \) is the identity matrix with dimension two and \( w_k \) is the process noise.
The process noise is the unknown acceleration for the UAV modeled as $\mathbf{w}_k \sim \mathcal{N}(0, \sigma^2_w)$. The process covariance matrix is defined as

$$Q_k = \text{Cov}(\mathbf{w}_k) = G\sigma^2_wG^T. \quad (4.3)$$

$Q_k$ is time invariant as the sampling time $T$ and $\sigma^2_w$ are constant. The assumption that the process noise is normal distributed is not always the case, e.g. if the UAV is affected by high dynamics during maneuvers. Therefore, $\sigma^2_w$ is replaced by a tuning parameter $q$ that depends mostly on good estimates from the filter rather than the physical interpretation as the acceleration $\sigma^2_w$.

The measurement equation is given by

$$\mathbf{y}_k = \begin{pmatrix} \rho_{bk} \\ \theta_{bk} \end{pmatrix} = h(\mathbf{x}_k) + \mathbf{e}_k, \quad (4.4)$$

where $\mathbf{e}_k$ is the error in range and azimuth modeled as white Gaussian noise. The range and azimuth error assumes to be uncorrelated which gives the noise covariance matrix

$$R = \begin{pmatrix} \sigma^2_r & 0 \\ 0 & \sigma^2_a \end{pmatrix}, \quad (4.5)$$

where $\sigma^2_r$ is the range variance and $\sigma^2_a$ is the azimuth variance.

The sensor model is defined as in (2.21) but with the elevation angle and attitude set to zero. The system dynamics are linear but the sensor model is not, for the measurement update in Algorithm 2.2 the sensor model needs to be linearized around the current estimate as

$$H = \left. \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x} = \hat{\mathbf{x}}_k} = \begin{pmatrix} p_n & p_v \\ \sqrt{p_n^2 + p_z^2} & \sqrt{p_v^2 + p_z^2} \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \quad (4.6)$$

Algorithm 2.2 is used for the time update and measurement update. The filter frequency is 50 Hz and the time update is performed each iteration, however, the measurement update is only performed if a new measurement is available.

An outlier rejection is implemented in the measurement update to take care of the outliers mentioned in Section 3.2. A standard hypothesis test is proposed as in [13, Section 7.6]. The test statistics is defined as

$$T(\mathbf{y}_k) = (\mathbf{y}_k - H_k \hat{\mathbf{x}}_{k|k-1})(H_kP_{k|k-1}H_k^T + R_k)^{-1}(\mathbf{y}_k - H_k \hat{\mathbf{x}}_{k|k-1}) \sim \chi^2_{n_y}, \quad (4.7)$$

where $n_y$ is the degrees of freedom for the signal, in this case $n_y = 2$. An outlier is detected and removed if $T(\mathbf{y}_k) > \chi^2_{n_y,\alpha}$ where $\alpha$ is the significant threshold to incorrectly detect an outlier.

## 4.2 Kinematics

There is no need for a complex motion model if the measurement uncertainties are relative high. Therefore, in the following filters the UAV is considered a particle with kinematics described in this section. Time notation will not be used – the kinematics are described in
continuous time in this section. The relevant states are

\[ \mathbf{x} = \begin{pmatrix} p_t \\ v_t \end{pmatrix} = \begin{pmatrix} p_n \\ p_e \\ p_d \\ v_n \\ v_e \\ v_d \end{pmatrix}. \]

(4.8)

The position dynamics are given by

\[ \dot{p}_t = v_t, \]

(4.9)

and the velocity dynamics are given by

\[ \dot{v}_t = R^t_b(\Phi) \mathbf{a}_b + \mathbf{g}, \]

(4.10)

where

\[ \mathbf{g} = \begin{pmatrix} 0 \\ 0 \\ 9.81 \end{pmatrix}. \]

(4.11)

\( R^t_b \) is the rotation matrix (2.7) and \( \mathbf{a}_b \) is the acceleration in the body frame. The total dynamics are summarized as

\[ \begin{pmatrix} \dot{p}_t \\ \dot{v}_t \end{pmatrix} = \begin{pmatrix} 0_3 & I_3 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0_3 \\ R^t_b(\Phi) \end{pmatrix} \mathbf{a}_b + \begin{pmatrix} \bar{0} \\ \mathbf{g} \end{pmatrix}, \]

(4.12)

where \( \bar{0} \) is the zero vector.

### 4.3 IMU Aided Algorithm

This algorithm is using the accelerometer measurements to estimate the velocity instead of assuming it to be constant as in the basic algorithm. The state vector is the same as the basic algorithm

\[ \mathbf{x} = \begin{pmatrix} p_n \\ p_e \\ v_n \\ v_e \end{pmatrix}. \]

(4.13)

The acceleration measurements are defined as (2.25) where the bias error in the body fixed z-axis is known. The bias in the z-axis can be subtracted from the measurements as

\[ \mathbf{a}_b = \hat{\mathbf{a}}_b - \mathbf{b}_{acc}. \]

(4.14)

The acceleration is transformed to the local NED frame as in (4.10) and used as input to the filter

\[ \mathbf{u} = \begin{pmatrix} a_n \\ a_e \end{pmatrix}. \]

(4.15)
By removing the down direction from (4.12) and discretize the dynamics using Euler forward approximation as in [12, Section 5.5] gives

$$x_{k+1} = \begin{pmatrix} I_2 & T \cdot I_2 \\ 0_2 & I_2 \end{pmatrix} x_k + \begin{pmatrix} 0_2 \\ T \cdot I_2 \end{pmatrix} (u_k + w_k), \quad (4.16)$$

where $w$ is the measurement noise from the accelerometer, assumed to be white Gaussian noise $w \sim \mathcal{N}(0, \sigma^2_w)$. The covariance matrix is defined as

$$Q_k = Cov(w) = G \sigma^2_w G^T. \quad (4.17)$$

The measurement equations and the outlier rejection are the same as in the basic algorithm.

### 4.4 IMU and Barometer Aided Algorithm

A barometer is added to the filter for altitude estimation. The design is similar to the one with the integrated accelerometer. The state vector is augmented, with two more components, into

$$x = \begin{bmatrix} p_n \\ p_e \\ p_d \\ v_n \\ v_e \\ v_d \end{bmatrix} \quad (4.18)$$

where $p_d$ is the down position and $v_d$ is the down velocity. The acceleration in the down direction is added as an input signal. The total dynamics are similar to the dynamics given in Section 4.3, i.e.

$$x_{k+1} = \begin{pmatrix} I_3 & T \cdot I_3 \\ 0_3 & I_3 \end{pmatrix} x_k + \begin{pmatrix} 0_3 \\ T \cdot I_3 \end{pmatrix} (u_k + w_k), \quad (4.19)$$

where

$$u_k = \begin{bmatrix} a_n \\ a_e \\ a_d \end{bmatrix} = R_b^t(\Phi)(\tilde{a}_b - b_{acc}) + g. \quad (4.20)$$

The measurement update and outlier rejection for PARS are the same as in Section 4.1. The barometer is added as a measurement with the sensor model described in Section 2.3.3. The estimate for the altitude and the down velocity is not coupled with the other states. This is done by doing the same approximation as in Section 2.3.1. The reason for this approximation is motivated by the results in Section 5.3.5.
4.5 Converted Measurements Kalman Filter

As mentioned before, the EKF algorithm is used to handle the nonlinear sensor model for PARS. However, the PARS measurements can be converted to Cartesian coordinates making it possible to use the KF algorithm instead of the EKF. The state vector is

\[
x = \begin{pmatrix} p_n \\ p_e \\ v_n \\ v_e \end{pmatrix},
\]

(4.21)

with the same dynamics and input as the algorithm explained in Section 4.3. The process noise covariance matrix is the same as in (4.17).

The PARS measurements can be converted to pseudo-measurements in Cartesian coordinates with (2.14) and (2.17). The new measurement equation is

\[
y_k = \begin{pmatrix} y_n \\ y_e \end{pmatrix} = H x_k + e_k,
\]

(4.22)

where

\[
H = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix},
\]

(4.23)

and \(e_k\) is the measurement noise approximated to be uncorrelated and modeled as white Gaussian noise which gives the following covariance matrix

\[
R = \begin{pmatrix} \sigma_n^2 & 0 \\ 0 & \sigma_e^2 \end{pmatrix},
\]

(4.24)

where \(\sigma_n^2\) and \(\sigma_e^2\) are the variance in north and east position. This is just an approximation as the pseudo-measurements are correlated and biased, the proof can be seen in [18]. The paper also presents a way to account for these errors in the measurement covariance matrix, however, the studied scenarios in [18] involve long distances to target and poor sensor measurements. In Chapter 3, it can be seen that the error in the azimuth for the PARS is small and the distance to the target is shorter than the experiments in [18]. Hence, the error from correlation and bias should be small.

The uncertainty in the position estimates from the filter is not affected by the distance between the UAV and the PARS ground station. The proposed outlier rejection for the Cartesian filter is to verify if the Euclidean distance between the estimates and the measurements are greater than some limit \(d_{\text{max}}\).

4.6 Stationary Kalman Filter

In Section 4.5, the state-space model used in the CMKF algorithm is time-invariant. The stationary Kalman filter described in [12, Algorithm 8.2] is therefore used to simplify the
The steady state Kalman gain is
\[ \bar{K} = \bar{P}H^T(H\bar{P}H^T + R)^{-1}, \]  
(4.25)
where the stationary covariance matrix \( \bar{P} \) is found by solving the stationary Riccati equation
\[ \bar{P} = F\bar{P}F^T - F\bar{P}H^T(H\bar{P}H^T + R)^{-1}H\bar{P}F^T + Q. \]  
(4.26)
The solution is found by iterating the Riccati equation until convergence. The state-space model is discretized with \( T = 0.1 \) s assuming that PARS measurements are sampled at 10 Hz.

The time update and measurement update for the stationary KF are
\[ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K(y_k - H\hat{x}_{k|k-1}), \]  
(4.27)
\[ \hat{x}_{k+1|k} = F\hat{x}_{k|k} + Gu_k. \]  
(4.28)
The filter is updated at 50 Hz and the outlier rejection implemented in Section 4.5 is used in the measurement update.

With the stationary properties the frequency response for the filter can be expressed as a linear filter, mentioned in [12, Section 8.3.3],
\[ \hat{X} = (zI - F + F\bar{K}C)^{-1}F\bar{K}Y(z) \]  
(4.29)
\[ H_{\hat{x}}(z) \]
This is used to characterize the filter in a linear fashion by studying the step response and the frequency response in a bode plot.
In this chapter the algorithms defined in Chapter 4 are evaluated. The chapter begins with a presentation of the experimental setup along with the studied flight scenarios. This is followed by the results from the simulations.

5.1 Experimental Setup

The experiment setup for the filter evaluation is similar to the one in Section 3.1. The setup for each flight can be found in Appendix A.1. The flights in Section 3.1 are used for filter tuning. The filters are evaluated with flights described in Section 5.2.

5.1.1 Time Synchronization and Delay

When fusing measurements from different sensors it is important to know when the measurements are obtained. The PARS ground station measurements are synchronized against a local clock on the UAV. The measurements from the PARS is affected by some delay because of the calculations and pre-processing done at the ground station. This is a real-time implementation problem and is not accounted for or studied further in this thesis.

5.1.2 Ground Truth

The GNSS solution implemented in Skeldar V-200 is used as ground truth for the validation. The drawbacks with this is mentioned in Chapter 3 and especially the bias mentioned in Section 3.2.1.
5.2 Scenarios

It is important to use different data for filter tuning and validation. This shows the consistency of the filters [12]. The data used for filter tuning is presented in Figures 5.1 and 5.2. Both flights are made over water with an altitude of 70 meters over the water for flight 1. The altitude for flight 2 is 70 meters in the start, increasing to a maximum altitude of 800 meters in the end of the flight. The average velocity of the UAV is about 40 km/h for flight 1 and 60 km/h for flight 2. The PARS position and attitude for each flight are presented in Appendix A.1.

Two flights are used to validate the consistency of the filters, they can be seen in Figures 5.3 and 5.4. Both flights are also made over water with an altitude of 40 meters over the water for flight 3 and an altitude of 240 meters for flight 4. The average velocity of the UAV is 70 km/h for flight 3 and 40 km/h for flight 4. Flight 3 is characterized by high dynamics with a lot of turns. The shorter distance from the ground station should give a better result as the range bias mentioned in Section 3.2.1 grow with longer range. Flight 4 is a longer flight with less dynamics than flight 3.

5.3 Results

The barometer aided filter in Section 4.4 is evaluated last in this section as the horizontal states are the same as the IMU aided algorithm. The other algorithms are presented and compared against each other to demonstrate the advantages and disadvantages with each filter. The notation of the filters used in the figures are presented in Table 5.1.
5.3 Results

Figure 5.3: Trajectory for the UAV and ground position for PARS in flight 3.

Figure 5.4: Trajectory for the UAV and ground position for PARS in flight 4.

Table 5.1: Notation for the filters

<table>
<thead>
<tr>
<th>Notation</th>
<th>Filter Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASIC</td>
<td>Basic algorithm</td>
</tr>
<tr>
<td>IMU</td>
<td>IMU aided algorithm</td>
</tr>
<tr>
<td>BARO</td>
<td>IMU and barometer aided algorithm</td>
</tr>
<tr>
<td>CMKF</td>
<td>Converted measurements Kalman filter</td>
</tr>
<tr>
<td>KF</td>
<td>Stationary Kalman filter</td>
</tr>
</tbody>
</table>

5.3.1 RMSE Evaluation

The RMSE mentioned in (3.3) is computed for the Euclidean distance between the ground truth and the estimated position, the result can be seen in Figure 5.5.

The basic algorithm performs better than the estimates from the PARS in three of the four flights. The poor RMSE for the basic filter in flight 3 indicates that the CV model used in the basic algorithm performs bad in flights with high dynamics such as hard turns. The position estimates in Figure 5.11 supports this statement as the filter performs notably worse than the other filters.

The RMSE shows that the IMU aided filter performs better than the other algorithms for all flights. The IMU aided filter is the most complex filter that incorporate most information from the measurements.

The CMKF algorithm performs almost as good as the IMU algorithm which is interesting as information about correlation and biases in the pseudo measurements are not included. The stationary KF is also performing equal with the CMKF, this is the case since the process noise and measurement noise are the same. The difference is that the CMKF account for the time difference between measurements and update the covariance matrix accordingly. This can be seen in Figure 5.6, where the variance is almost stationary with small changes that depend on the time difference between measurements from PARS.

The largest increase in accuracy is in flight 2 compared to the position estimates from
PARS. Flight 2 is highly affected by outliers which contributes to greater error, the outlier rejection implemented in the filters seems to do a good job rejecting those measurements. The number of outliers rejected for each flight and filter are presented in Figure 5.22.

### 5.3.2 Position Estimates

Figures 5.7 - 5.10 are position estimates from flight 2 when the UAV is the furthest away from the ground station. The offset in the estimates compared to the ground truth is assumed to be the bias mentioned in Section 3.2.1. The covariance matrix for the basic and IMU filters increase with the distance from the ground panel, which results in the filters relying more on the time update rather than the measurements. It is clear that the basic model is insufficient when the UAV is performing a maneuver, the IMU filter performs much better as it measures the acceleration.

By studying Figures 5.9 and 5.10 it can be seen that the CMKF and the stationary KF relies more on the measurements and they are not as smooth as the IMU estimates in Figure 5.8. It is hard to tune those filters to perform well on both longer and shorter distances. Figures 5.12 - 5.14 show that the performance of the CMKF and the stationary KF are similar to the IMU algorithm on shorter distances.

The performance of the different filters on a straight trajectory with some disturbances from PARS can be seen in Figures 5.15 - 5.18. The IMU algorithm seems to be the smoothest one, the disturbance from the PARS measurements in the middle does not affect the position estimates that much. The basic filter performs better than the CMKF and the stationary KF, however, they are all affected by the disturbance from the PARS measurements.
5.3 Results

**Figure 5.6:** The CMKF and stationary KF north position variance for flight 2.

**Figure 5.7:** Position estimates from the basic algorithm for flight 2.

**Figure 5.8:** Position estimates from the IMU aided algorithm for flight 2.
**Figure 5.9:** Position estimates from the CMKF algorithm for flight 2.

**Figure 5.10:** Position estimates from the stationary KF for flight 2.

**Figure 5.11:** Position estimates from the basic algorithm for flight 3.

**Figure 5.12:** Position estimates from the IMU algorithm for flight 3.

**Figure 5.13:** Position estimates from the CMKF algorithm for flight 3.

**Figure 5.14:** Position estimates from the stationary KF for flight 3.
5.3 Results

Figure 5.15: Position estimates from the basic algorithm for flight 4.

Figure 5.16: Position estimates from the IMU algorithm for flight 4.

Figure 5.17: Position estimates from the CMKF algorithm for flight 4.

Figure 5.18: Position estimates from the stationary KF for flight 4.
5.3.3 Step Response

The step response for a filter determines how fast the filter converges to the measurements. The step response is achieved by simulating PARS measurements with constant range $\rho_t$ and a step in the azimuth angle $\theta_t$ at time $t_{\text{step}}$. The step is made at $t_{\text{step}} = 500$ s with a step size of $1^\circ$ in the azimuth angle at three different range measurements. The simulated measurements are obtained every $T_s = 0.1$ s with no noise added to the signals. The step in the PARS measurements result in a step in both the north and east position, however, it is only the east position that is presented since the time constant $\tau$ is the same for both positions, where $\tau$ is defined in [8, Section 2.6] as

$$\tau = 0.63(\rho_c(t_{\text{step}}) - \rho_c(t_{\text{step}} - T_s)).$$  \hspace{1cm} (5.1)

The result can be seen in Figures 5.19 - 5.21, where the range $\rho_t$ is in the same range as the position estimates presented in Section 5.3.2. A faster step response indicates that the measurements are trusted more than the model used in the time update step.

When the simulated measurements are obtained at a constant frequency the CMKF time constant is the same for all distances from PARS. The time constant for the stationary KF is always the same since the Kalman gain is constant. The step response can therefore be calculated as mentioned in (4.29).

The time constant for the IMU aided algorithm is much slower than the other filters and is therefore not affected that much from PARS disturbances.

5.3.4 Outlier Rejection

Section 3.2 shows that the PARS measurements are affected by outliers. The result from the outlier rejection algorithms explained in Chapter 4 are presented in Figure 5.22. Flight 2 seems to be affected the most by outliers, it is also the flight with the greatest increase in performance when looking at the RMSE in Section 5.3.1. It seems like the outlier rejection implemented in the basic and IMU aided algorithm detects more outliers than the other one. This is because of the maximum distance tolerated between a measurement
and an estimate for the outlier rejection implemented in the CMKE and stationary KF. The filters diverge at longer distances if the maximum distance limit is too small, a greater limit is therefore chosen. The trade-off is that the outliers at shorter distances are harder to detect, but it is better than divergence of the estimates at longer distances.

5.3.5 IMU and Barometer Aided Algorithm

The results for the barometer aided algorithm are presented in this section. The down direction has been changed to the up direction to give a more intuitive representation of the altitude.

By studying the results in Figures 5.23 and 5.24 it can be seen that an offset of 5-10 meters is present in the altitude estimate. This is due to the standard model used in Section 2.3.3, the pressure varies depending on the weather conditions and deviate from the standard model. However, this is not a great problem in this application since the UAV is flying higher than 10 meters after take-off. Under the landing and take-off is a ranging sensor used instead of the barometer, but this is not included in the thesis.

Figure 5.25 shows that the high frequency noise is almost removed from the barometer measurements, however, the lower frequency noise that depend on local pressure changes are not removed. The error from the barometer is not Gaussian distributed which can be seen in Figure 5.26.

The step response can be seen in Figure 5.27, with a time constant of $\tau = 5$ s when the covariance is set to $R_{\text{baro}} = 1000000 \text{ m}$. 

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure521.png}
\caption{Step response with $\rho_t = 1000\text{ m}$.}
\end{figure}
Experimental Evaluation

Figure 5.22: Outliers rejected from the filters.

Figure 5.23: Altitude from the GNSS solution compared to the barometer aided filter for flight 1.
Figure 5.24: Altitude from the GNSS solution compared to the barometer aided filter for flight 2.

Figure 5.25: Raw barometer altitude compared to the filtered altitude
Figure 5.26: Error distribution for flight 4.

Figure 5.27: Step response for the altitude.
5.3.6 Time Complexity

The computational power for the UAV is limited and it is therefore essential to study the time complexity of each filter. This is either done by studying time complexity with a big-O analysis or by measuring the actual time it takes for a filter loop. The second method is proposed here, as it is enough to show the difference between the filters.

The result is presented in Figure 5.28 and it is the average loop time obtained from flight 1. These results should be used as a comparison between the filters and not to conclude if the filters are fast enough for an implementation.
Conclusion and Future Work

This chapter discusses the questions asked in Section 1.3 with the results from Chapters 3 and 5. This is followed by suggestions for future work.

6.1 Conclusion

This thesis has studied the possibility to use a positioning system for a UAV independent from GNSS with the use of PARS and on board sensors. The first part of the thesis was to compare the PARS against the GNSS solution implemented in the UAV, which gave promising results in the range and azimuth measurements. However, the elevation measurements were full of noise caused by reflections in the water. It was decided to remove it from the problem and to investigate the horizontal position alone. The range measurement was affected by a bias that increased with the distance to target. This was likely due to poor modeling of the earth, a ECEF frame would be preferred in a similar comparison. The azimuth was also affected by the reflections, but not to the same extent as the elevation angle. It was concluded that the PARS can be a decent positioning system on its own, especially since it was designed for communication and not positioning. It can be used to monitor deviations from the GNSS position and detect jamming and spoofing attacks.

The second part of this thesis has studied if a filter can improve the position estimates from the PARS. Five filters were derived and compared against each other. Three filters were using the EKF algorithm to incorporate the nonlinear spherical measurements from the PARS. The other two transformed the spherical coordinates to Cartesian coordinates before using the KF algorithm. Different aspects to the filtering problem were studied, the filter needs to be robust, low computational power and small acceptance of error.

Based on the results in Chapter 5, the filter that performed the lowest error was the IMU
aided EKF algorithm. The estimates from the filter were also smooth and followed the ground truth trajectory well. One major problem with the filter was that it needed a lot of computational power. Hence, the stationary KF can be a better choice since it was less computational demanding and the error was similar to the IMU aided EKF at shorter distances and decent at long distances. The choice of a CMKF algorithm with the updated covariance matrix was not motivated since the result was almost the same as the stationary KF.

The basic algorithm with the constant velocity model has the worst performance regarding the results in Chapter 5, in particular when the UAV performed maneuvers at long distances. However, the implementation was independent from on board sensors which results in a more standalone solution compared to the other filters.

Lastly the barometer aided algorithm was evaluated, as seen from the result in Section 5.3.5, the barometer measurements were affected by local pressure variations that depends on the weather conditions. It was decided to decouple the altitude position with the horizontal positions. This was motivated by a simpler solution and a more robust position in the horizontal plane.

The final conclusion was that the IMU and barometer aided algorithm would be the best solution, the stationary KF algorithm can be motivated to use for a less computational demanding solution. The results showed that it were possible to navigate with both solutions, however, the system should be used as a redundancy system since the GNSS solution was more accurate.

6.2 Future Work

This section includes future work that was outside the scope for this thesis but can be interesting for a future implementation.

In this thesis, the attitude angles were assumed to be known. In the final filter solution, the attitude angles along with the gyro biases need to be estimated. This will make the filter a lot more complex since the dynamic model will be nonlinear and 6 more states need to be estimated. A more complex filter like a unscented Kalman filter (UKF) can be studied to get a better result, this might be to computational heavy to do online but it might be good with a theoretical limit for the estimation error.

There will be some challenges to implement the filter in a real time system. The DOA calculations need to be done in real time at the ground, preferable on a Field-Programmable Gate Array (FPGA) that can perform the calculations fast. The pose of the PARS ground panel needs to be known, this can be done by either measure it with GNSS and an inclinometer or estimate it as in [21]. It will be important to accurately synchronize the measurements from the PARS with the IMU measurements, one example can be to use the SyncBoard in [2] or similar hardware.

It would be interesting to add more ground panels to make the system more robust to the multipath propagation problem mentioned in Section 3.2 and make it possible to use the elevation angle in the estimation problem.
Appendix
A.1 Flight Parameters

Table A.1: Azimuth pose for PARS ground panel.

<table>
<thead>
<tr>
<th>Flight</th>
<th>Flight number</th>
<th>$\theta_0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10118</td>
<td>274.25°</td>
</tr>
<tr>
<td>2</td>
<td>10119</td>
<td>234.80°</td>
</tr>
<tr>
<td>3</td>
<td>10115</td>
<td>255.25°</td>
</tr>
<tr>
<td>4</td>
<td>10117</td>
<td>274.25°</td>
</tr>
</tbody>
</table>

Where $\theta_0$ is the azimuth pose for the PARS ground panel and the flight number is an intern company marking for a flight.

A.2 Filter Parameters

The filters need to be initialized with starting parameters that is the same for all flights. The parameters are

- $n_x$ is the dimension for the state vector.
- $x_0$ is the estimated starting states for the UAV.
- $P_0$ is the error covariance matrix for the starting state.
- $R$ is the measurement noise matrix.
- $Q$ is the process noise matrix.
- $f$ is the update frequency for the filter.
• $d_{\text{max}}$ is the outlier rejection parameter explained in Section 4.5.
• $\chi^2_\alpha$ is the outlier rejection limit explained in Section 4.1.

A.2.1 Basic Filter

• $n_x = 4$
• $x_0$ is initialized from GNSS measurement.
• $P_0 = \text{diag}(10 \text{ m}^2, 10 \text{ m}^2, 10 \text{ m}^2/\text{s}^2, 10 \text{ m}^2/\text{s}^2)$
• $R = \text{diag}(7.5 \text{ m}, 0.1^\circ)$
• $Q = 10^{-4} \begin{pmatrix} 0.04 \text{ m}^2 & 0 & 4 \text{ m}^2/\text{s} & 0 \\ 0 & 0.04 \text{ m}^2 & 0 & 4 \text{ m}^2/\text{s} \\ 4 \text{ m}^2/\text{s} & 0 & 400 \text{ m}^2/\text{s}^2 & 0 \\ 0 & 4 \text{ m}^2/\text{s} & 0 & 400 \text{ m}^2/\text{s}^2 \end{pmatrix}$
• $f = 50 \text{ Hz}$
• $d_{\text{max}}$ not used
• $\chi^2_\alpha = 4$

A.2.2 IMU Filter

• $n_x = 4$
• $x_0$ is initialized from GNSS measurement.
• $P_0 = \text{diag}(10 \text{ m}^2, 10 \text{ m}^2, 10 \text{ m}^2/\text{s}^2, 10 \text{ m}^2/\text{s}^2)$
• $R = \text{diag}(7.5 \text{ m}, 0.1^\circ)$
• $Q = 10^{-4} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 4 \text{ m}^2/\text{s}^2 & 0 \\ 0 & 0 & 0 & 4 \text{ m}^2/\text{s}^2 \end{pmatrix}$
• $f = 50 \text{ Hz}$
• $d_{\text{max}}$ not used
• $\chi^2_\alpha = 4$

A.2.3 Barometer Filter

• $n_x = 6$
• $x_0$ is initialized from GNSS measurement.
• $P_0 = \text{diag}(10 \text{ m}^2, 10 \text{ m}^2, 10 \text{ m}^2, 10 \text{ m}^2/\text{s}^2, 10 \text{ m}^2/\text{s}^2, 10 \text{ m}^2/\text{s}^2)$
• $R_{\text{pars}} = \text{diag}(7.5 \text{ m}, 0.1^\circ)$ and $R_{\text{baro}} = 1000000 \text{ m}$
A.2 Filter Parameters

• $Q = 10^{-4} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 \text{ m}^2/\text{s}^2 & 0 & 0 \\ 0 & 0 & 0 & 4 \text{ m}^2/\text{s}^2 & 0 \\ 0 & 0 & 0 & 0 & 4 \text{ m}^2/\text{s}^2 \end{pmatrix}$

• $f = 50 \text{ Hz}$
• $d_{max}$ not used
• $\chi_{\alpha}^2 = 4$

A.2.4 CMKF and stationary KF

• $n_x = 4$
• $x_0$ is initialized from GNSS measurement.
• $P_0 = \text{diag}(10 \text{ m}^2, 10 \text{ m}^2, 10 \text{ m}^2/\text{s}^2, 10 \text{ m}^2/\text{s}^2)$
• $R_{pars} = \text{diag}(100 \text{ m}, 100 \text{ m})$

• $Q = 10^{-4} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.4 \text{ m}^2/\text{s}^2 & 0 \\ 0 & 0 & 0 & 0.4 \text{ m}^2/\text{s}^2 \end{pmatrix}$

• $f = 50 \text{ Hz}$
• $d_{max} = 300 \text{ m}$
• $\chi_{\alpha}^2$ not used


