

UNMANNED AERIAL VEHICLE TRACKING SYSTEM WITH
OUT-OF-SEQUENCE MEASUREMENT IN A DISCRETE
TIME-DELAYED EXTENDED KALMAN FILTER

by

Roque Lora

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Electrical Engineering

Approved:

Rajnikant Sharma, Ph.D.
Major Professor

Rees Fullmer, Ph.D.
Committee Member

Todd K. Moon, Ph.D.
Committee Member

Mark R. McLellan, Ph.D.
Vice President for Research and
Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2017

Copyright © Roque Lora 2017

All Rights Reserved

ABSTRACT

Unmanned Aerial Vehicle Tracking System with
Out-Of-Sequence Measurement in a Discrete
Time-Delayed Extended Kalman Filter

by

Roque Lora, Master of Science
Utah State University, 2017

Major Professor: Rajnikant Sharma, Ph.D.
Department: Electrical and Computer Engineering

This extends the delayed Kalman filter to a delayed extended Kalman filter that takes into account the delays of the measurement. The filter uses GPS and camera measurements for the estimation of the states of an unmanned aerial system (UAS). The GPS output is delayed by a random number, which produces the out-of-sequence measurement problem. The filter is based on calculating the states with the measurements at the moment, so that when the delayed measurements are available, the optimal filter for the time of the measurement is calculated and propagated to the present time. The estimates provided by the filter are used to point an antenna and a camera towards the UAS.

(61 pages)

PUBLIC ABSTRACT

Unmanned Aerial Vehicle Tracking System with
Out-Of-Sequence Measurement in a Discrete
Time-Delayed Extended Kalman Filter
Roque Lora

The goal of this thesis is to extend the delayed Kalman filter so it can be used with non-linear systems and that it can handle randomized delays on the measurements. In the particular case of this study, the filter is used to estimate the states of an unmanned aerial system. The outputs of the filter are used to point an antenna and a camera towards a UAS. Different scenarios are simulated for the purpose of comparing the efficiency of this technique in various situations.

To God, my wife, Dr. Sharma, my family, AggieAir, and all my friends.

ACKNOWLEDGMENTS

I want to acknowledge God, as the one that gave me the opportunity and strength to reach this milestone in my life. My wife, who was the biggest pillar in my life during the whole master. She always supported me giving me strength during the successes and failures and was the one that comforted me when I felt frustrated. She is the main source of my inspiration. Dr. Sharma, who made this whole research possible. Dr. Sharma took me as his student since he arrived at USU, even when I did not have a clear subject for my research. He dedicated countless hours of his time to polish and guide me through this whole research. I am very grateful for his support, patience and the knowledge that he shared with me. To my family, who gave me support and believed in me since day one. To AggieAir, who taught me so much and helped me developed as a professional. And to all the friends that made this whole experience unforgettable.

Thank you,

Roque Lora

CONTENTS

	Page
ABSTRACT	iii
PUBLIC ABSTRACT	iv
ACKNOWLEDGMENTS	vi
LIST OF FIGURES	ix
ACRONYMS	x
1 INTRODUCTION	1
1.1 Background	1
1.2 Contributions	2
1.3 Literature Review	4
2 SENSORS AND KALMAN FILTERS	6
2.1 Aircraft Dynamics	7
2.2 Global Positioning System	8
2.2.1 GPS Measurement Error Model	8
2.2.2 GPS Smoothing	12
2.3 Camera	16
2.3.1 Gimbal and Camera Frames and Projective Geometry	16
2.3.2 Camera Model	17
2.3.3 Gimbal Pointing	19
2.3.4 Geolocation	21
2.4 Kalman filter	24
2.4.1 Discrete Kalman filter	25
2.4.2 Extended Kalman filter	29
2.4.3 Delayed Extended Kalman filter	30
3 SIMULATION	33
3.1 Simulation Overview	33
3.2 Antenna Pointer	34
3.2.1 GPS	34
3.2.2 Camera	35
3.2.3 EKF/DEKF	35
3.2.4 Pointer	35
3.3 3-D Plot	36

4	RESULTS	37
4.1	Comparison of Estimations Between GPS and GPS Coupled with Camera .	37
4.2	Comparison of Estimation Accuracy for EKF and DEKF with 0.75 s Random Delay	41
4.3	Comparison of Estimation Accuracy Between No delay, 0.25 s, 0.50 s, and 0.75 s Random Delays	44
5	CONCLUSION	47
	REFERENCES	49

LIST OF FIGURES

Figure	Page
1.1 Antenna Pointer tracking an UAS	2
1.2 Out-Of-Sequence Measurement	2
3.1 Antenna Pointer Block	33
3.2 Antenna Pointer Block	34
3.3 Antenna Pointer tracking a UAV.	36
4.1 UAV Trajectory Comparison - GPS vs GPS and Camera	38
4.2 North Position Error Comparison - GPS vs GPS and Camera	38
4.3 East Position Error Comparison - GPS vs GPS and Camera	39
4.4 Azimuth Error Comparison - GPS vs GPS coupled with Camera	39
4.5 Elevation Error Comparison - GPS vs GPS coupled with Camera	40
4.6 UAV Trajectory Comparison - EKF vs DEKF	41
4.7 North Position Error Comparison - EKF vs DEKF	42
4.8 East Position Error Comparison - EKF vs DEKF	42
4.9 Azimuth Error Comparison - EKF vs DEKF	43
4.10 Elevation Error Comparison - EKF vs DEKF	43
4.11 UAV Trajectory Comparison - Different Delays	44
4.12 North Position Error Comparison - Different Delays	45
4.13 East Position Error Comparison - Different Delays	45
4.14 Azimuth Error Comparison - Different Delays	46
4.15 Elevation Error Comparison - Different Delays	46

ACRONYMS

DEKF	Delayed Extended Kalman filter
DOP	Dilution Of Precision
EO	Electro-Optical
EKF	Extended Kalman filter
FAA	Federal Aviation Administration
GCS	Ground Control Station
GPS	Global Positioning System
IMU	Inertial Measurement Unit
IR	Infrared
KF	Kalman filter
LOS	Line-of-Sight
MSE	Mean Squared Error
OOSM	Out Of Sequence Measurement
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle

CHAPTER 1

INTRODUCTION

1.1 Background

Recently the use of unmanned aerial systems (UAS) has increased dramatically in both civilian and military applications. Because of this, the Federal Aviation Administration (FAA) has been modifying the requirements for the operation of UAS. The one that concerns us is the line-of-sight (LOS) requirement, which constrains the aircraft mission to be within the visual field of the operator. The existing method is to keep an eye on the aircraft with binoculars, which is prone to lose the plane frequently. The importance of UAS tracking systems relies on having a good link between the aircraft and the ground control station (GCS), which can be assured using a good antenna pointing system.

Another critical requirement in the reliability of a UAS mission is the communication with the ground station. For reliable communication it is necessary to point the antenna towards the UAS and maintain the LOS with the aircraft. Therefore, the communication link will be strong as long as we have LOS. This is apparent for applications that require real-time feedback, because if the tracking of the UAS is not accurate, the link could be interrupted and no feedback would be sent back. An example is a military mission where an aircraft supports an advancing troop in enemy territory. The decisions made by the leader of the troop are greatly influenced by the feedback of the UAS. If the communication between the UAS and the GCS is lost, there would not be any feedback from the UAS and the safety of the soldiers could be compromised. Moreover, in a civilian application such as disaster relief for an earthquake, the aircraft could be equipped with a thermal imaging camera to detect heat sources. It would identify warm bodies under the rubble in real time so the help could get there as soon as possible. If the communication is lost between the UAS and the GCS, some information might be missed that could lead to the detection of a

survivor.

1.2 Contributions

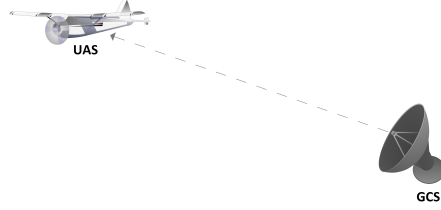


Fig. 1.1: Antenna Pointer tracking an UAS

The main focus of this work is to design an autonomous system to point an antenna mounted with a camera towards a flying UAS to maintain communication and visual LOS to satisfy FAA requirements. In order to point a camera, the precise position of the aircraft is required. The existing approach is for the GCS to use the position obtained from a GPS receiver on board the UAS. There are several limitations of this approach.

- The GPS receives position updates at 1-4 Hz, not fast enough to point the antenna towards an UAS flying at 20 m/s.
- The position updates at the antenna are delayed and have out-of-sequence measurements (OOSM) due to communication delays.

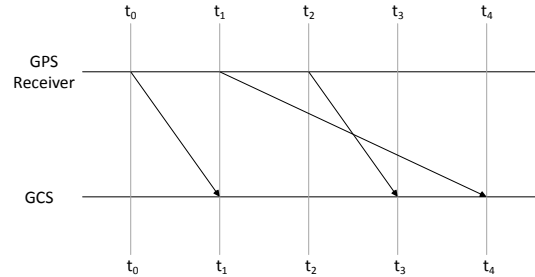


Fig. 1.2: OOSM. In this diagram the generation time of various measurements (GPS receiver line) and the time that they are available to the system (GCS line) are shown. Measurements t_1 and t_2 are out-of-sequence since t_2 is available first than t_1 .

A measurement is called out-of-sequence if it is available to a processing unit first than another measurement taken in an earlier time step. This situation is shown in figure 1.2.

The first problem is addressed by adding a camera which provides location measurements from images of the aircraft at 30 Hz, compensating the slow update rate of GPS. However, the camera's accuracy deteriorates with distance, since at higher distances the aircraft would only be a couple of pixels in the camera image, making the recognition of the UAS impossible. On the other hand, GPS measurements' errors are independent of the distance between the fixed wing and the antenna pointer, making the relative error smaller at higher distances. Hence, we can say that the camera and GPS are complementary sensors for pointing the antenna. The second problem of receiving out of sequence measurements is addressed by using an extended Kalman filter which uses out of sequence GPS measurements with camera measurements to estimate the position of the UAS. This estimation is then used to reliably point the antenna and camera towards the aircraft.

1.3 Literature Review

For more than two decades, the problem of OOSM have been addressed in several different ways. The initial studies about the OOSM are presented in [1] in 1993, where a negative-time update technique is developed using the criterion of minimum mean-square error. The first optimal algorithm for the general multi-step lag problem appeared in [2] in 2001. It consisted in buffering all the measurements received and running the filter behind real-time. Consequently, this algorithm is not able to run in real time. In [3], a solution of this problem is presented, in which a new filter is developed where the measurements are extrapolated forward in time using past and present estimates to calculate an optimal gain for the Kalman filter. The work of [3] is expanded in [4], where a covariance union method is proposed to solve the multi-lag OOSM when the delay is not precisely known. In [5] a linear minimum variance estimation algorithm was extended to handle multiple lags and dynamics models. This exposed that as the number of lags increases, the state estimation accuracy decreases. The same problem is addressed in [6]. The one-step lag OOSM solution is generalized to solve problems when multiple lags arise. This extension comes with a significantly lower storage requirement and a small (1%) degradation of MSE performance. A two-step method was presented in [7] which requires one more step to update the state at the OOSM time. Two optimal algorithms were presented in [8] to approach the multi-step OOSM problem using fixed-point smoothing. It was shown that both of these algorithms are flexible and relatively simple but highly computationally demanding. In [9] the exact state update equation for the OOSM problem is presented and compared to two suboptimal algorithms. Each method is tied to the situations where they should be used comparing simplicity against optimality. A smoothing-based algorithm was proposed in [10] in the case of multi-sensor multi-target ground moving target tracking problem. It concluded that discarding OOSM can lead to severe degradation in the state estimation. Also these two methods have better accuracy than the previous buffering method approach. A unified sub-optimal Bayesian approach is proposed in [11] for the multi-lag OOSM problem. This algorithm, developed on the basis of a fixed-lag smoothing framework and its solution

reduces to an augmented state Kalman filter. An extension of the particle filter approach can be found in [12] where it compares its results with the extended Kalman filter, showing that both have similar performance. The work of [13] develops an out-of-order sigma-point Kalman filter for solving the multi-lag OOSM in a single measurement vector and using the batch-form update. This method is equivalently complex per iteration as the sigma-point Kalman filter but it requires more iterations. An extended Kalman filter was used to estimate the position of a UAS using a camera and then coupled with an inertial measurement unit (IMU). The results shown that coupling the camera with an IMU reduces the position estimation error by almost an order of magnitude, compared to using only the camera [14]. In [15] three different approaches were proposed to achieve optimality. The study yielded that the complete in-sequence information fixed point smoothing method was the optimal approach, but it lacks simplicity.

The work of this thesis is developed by combining the delayed Kalman filter of [3] and the extended Kalman filter explained in [16]. Both techniques are combined in a filter with a nonlinear model of the system that can handle delays. Moreover, random delays and the OOSM problem will be handled by the modified filter. The estimation errors will be compared for different GPS measurement delays, fixed and randomized, as well as using GPS alone or coupled with a camera. Also, for comparison purposes, the estimation is done by an extended Kalman filter which will not take into account the delays or the OOSM problem. Therefore, the major contribution of this study is to clearly show the increase in accuracy that comes from using this delayed extended Kalman filter.

CHAPTER 2

SENSORS AND KALMAN FILTERS

In this chapter we explain the required elements for the antenna pointer system. The first requirement to track a UAS is some information about its position. This data is provided by a GPS mounted in the aircraft and a camera mounted in the antenna pointer. These sensors are low cost and are very common in unmanned aircraft systems. The combination of these sensors is due to the characteristics of them. Since the GPS measurement errors are independent of the distance between the aircraft and the antenna pointer, it is relatively accurate at medium and long distances, but not at short range. By contrast, the camera can estimate correctly the position of an object within short range, but at medium and long distances it can not detect the target since it is just a few pixels in the image, therefore at long distances it is practically unusable. Furthermore, the GPS has an update frequency of 1 Hz to 4 Hz, while cameras can sample from 5 Hz to more than 20 Hz. In section 2.2 we explain how GPS works and provide an algorithm to estimate the states of a UAS. Then in section 2.3 we present the camera model used for this simulation, and some coordinates frames that we use to geolocate a target based on camera images. Also a simple algorithm for pointing the antenna towards the UAS is shown.

The second requirement of our system is the ability to process the information received from both sensors to fuse the data and improve the estimation of the states of the UAS. For this purpose, a Kalman filter is used. In section 2.4 we explain briefly the basic Kalman filter. Later on, the extended Kalman filter is presented. Different reasons can cause delays on the measurements. Therefore, an algorithm called delayed extended Kalman filter is explained, which takes in consideration the time lag on the measurements and compensates for it to produce a better estimation of the states.

2.1 Aircraft Dynamics

12-state equations of motion

$$\begin{aligned}
\dot{p}_n &= (\cos \theta \cos \psi)u + (\sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi)v \\
&\quad + (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)w \\
\dot{p}_e &= (\cos \theta \sin \psi)u + (\sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi)v \\
&\quad + (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi)w \\
\dot{h} &= u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta \\
\dot{u} &= rv - qw - g \sin \theta \\
&\quad + \frac{\rho V_a^2 S}{2m} [C_X(\alpha) + C_{X_q}(\alpha) \frac{cq}{2V_a} + C_{X_{\delta_e}}(\alpha) \delta_e] \\
&\quad + \frac{\rho S_{prop} C_{prop}}{2m} [(k_{motor \delta_t})^2 - V_a^2] \\
\dot{v} &= pw - ru + g \cos \theta \sin \phi + \frac{\rho V_a^2 S}{2m} \\
&\quad \times [C_{Y_0} + C_{Y_\beta} \beta + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a + C_{Y_{\delta_r}} \delta_r] \\
\dot{w} &= qu - pv + g \cos \theta \cos \phi \\
&\quad + \frac{\rho V_a^2 S}{2m} [C_Z(\alpha) + C_{Z_q}(\alpha) \frac{cq}{2V_a} + C_{Z_{\delta_e}}(\alpha) \delta_e] \\
\dot{\phi} &= p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\
\dot{\theta} &= q \cos \phi - r \sin \phi \\
\dot{\psi} &= q \sin \phi \sec \theta + r \cos \phi \sec \theta \\
\dot{p} &= \Gamma_1 pq - \Gamma_2 qr + \frac{1}{2} \rho V_a^2 S b \\
&\quad \times [C_{p_0} + C_{p_\beta} \beta + C_{p_p} \frac{bp}{2V_a} + C_{p_r} \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r] \\
\dot{q} &= \Gamma_5 pr - \Gamma_6 (p^2 - r^2) + \frac{\rho V_a^2 S c}{2\mathcal{J}_y} \\
&\quad \times [C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \frac{cq}{2V_a} + C_{m_{\delta_e}} \delta_e] \\
\dot{r} &= \Gamma_7 pq - \Gamma_1 qr + \frac{1}{2} \rho V_a^2 S b \\
&\quad \times [C_{r_0} + C_{r_\beta} \beta + C_{r_p} \frac{bp}{2V_a} + C_{r_r} \frac{br}{2V_a} + C_{r_{\delta_a}} \delta_a + C_{r_{\delta_r}} \delta_r]
\end{aligned}$$

2.2 Global Positioning System

The global positioning system (GPS) is a navigation system supported by satellites that provides location information for objects on or near the surface of the earth. The navigation satellite timing and ranging (NAVSTAR) GPS has been functioning since 1993 and was developed by the United States Department of Defense. The GPS is one of the most important technologies that allows the functioning of UAS, giving them spatial localization. A more detailed description of the global navigation satellite systems can be found in many texts such as [17], [18], [19], [20], and [21]. In this section we explain briefly how the sensing works in GPS systems and we present a GPS model appropriate for simulation.

The most important element in the GPS system is the 24 satellites that orbit continuously around the earth at an altitude of 20,180 km [17]. The constellation of the satellites orbits is configured so that any location on the surface of the earth can be observed by at least four satellites all the time. It is common knowledge that with one range measurement is possible to locate a point in a line. With two range measurements a point on a plane and with three range measurements, a point on a 3-D surface. This explains the need for three of the four satellites. But the last one comes from clock synchronization errors between the satellites and the receiver. Therefore, with four independent pseudo-range measurements, there exists a system of four nonlinear equations which yield to the solution of the four variables of interest: latitude, longitude, altitude and receiver clock time offset [17]. This section is based on the work of [22].

2.2.1 GPS Measurement Error Model

The two factors that affect the precision of a GPS position measurement are the accuracy of the satellite pseudo-range measurements and the configuration of the satellites from which these pseudo-measurements are taken. The first factor is impacted by errors in the time of flight for each satellite. Where the second one is considered in a term called dilution of precision (DOP).

Standard Pseudo-Range Error Sources

Ephemeris Data

The satellite ephemeris gives the position of the satellite in the sky at a given time. The position of the GPS receiver is calculated using measurement ranges with respect to satellites. To determine this position, we are required to know the locations of the satellites in the first place. Ephemeris errors in the measurements exist because of the inaccuracies in the transmitted orbital satellite location. Common errors vary from 1 to 5 m.

Satellite Clock

The atomic clocks used in GPS satellites are made of cesium and rubidium. These clocks can drift about 10 ns a day. This translates to an error of 3.5 m. But their clocks are synchronized twice a day, therefore leading to an approximately 1.75 m error.

Ionosphere

This is the topmost layer of the atmosphere of the earth and consists of a shell of electrons and electrically charged atoms and molecules that surround the earth. These particles can delay the transmission of GPS signals. Even though the GPS receivers compensate for this delay, variations in the speed of light along this atmosphere's layer, is the biggest cause of GPS measurement errors, which contributes to approximately 2 to 5 m.

Troposphere

The troposphere is the lowest portion of the atmosphere containing approximately 80% of the atmosphere's mass and is located at an altitude of 7 to 20 km. Most of the weather phenomena occurs in this region. This produces variations in the speed of light and consequently in the time of flight and pseudo-range estimates. These variations introduce errors in the GPS measurements of about 1 m.

Multipath Reception

Multipath errors occur when reflected signals are received by the GPS receiver, mixing with

the real signal. Big surfaces as large buildings or structures increases this error. Normally this error source contributes to 1 m of variation in the GPS measurement.

Receiver Measurement This error is caused by the implicit limits with which the time of the satellite signal can be calculated. Modern receivers errors for this source is less than 0.5 m.

The pseudo-range error sources described previously are considered statistically uncorrelated and we can use the root sum of squares to add them. The addition of all of these error sources is called the user-equivalent range error. Parkinson, et al. [18] characterized these errors as a combination of slowly varying biases and random noise.

Transient Characteristic of GPS Positioning Error

For simulation purposes we are required to know the dynamic characteristics of the GPS error. According to [23], the dynamic model that describes the GPS error is a Gauss-Markov process which is modeled by

$$\nu[n+1] = e^{-k_{GPS}T_s}\nu[n] + \eta_{GPS}[n], \quad (2.1)$$

where $\nu[n]$ is the error being modeled, $\eta_{GPS}[n]$ is zero-mean Gaussian white noise, $1/k_{GPS}$ is the time constant of the process, and T_s is the sample time. An appropriate model for simulation of GPS measurements is given by

$$y_{GPS,n}[n] = p_n[n] + v_n[n] \quad (2.2)$$

$$y_{GPS,e}[n] = p_e[n] + v_e[n] \quad (2.3)$$

$$y_{GPS,d}[n] = p_d[n] + v_d[n], \quad (2.4)$$

where p_n , p_e , and p_d are the position coordinates in the local tangent plane or the North East Down coordinate system, and n is the time step of the measurement. Usually the GPS

receivers used by small UAV have a measurement frequency of 1 to 4 Hz.

GPS Velocity Measurements

Using the Doppler effect, which is the changes in frequency of a wave for an observer moving relative to its source, the speed of the GPS receiver can be estimated. This approximation has a standard deviation of 0.01 to 0.05 m/s. Modern GPS receivers usually provide more than just the position and also include the velocity of the receiver, horizontal ground speed and course angle [22]. The last two are calculated from the north and east velocity components as

$$V_g = \sqrt{V_n^2 + V_e^2} \quad (2.5)$$

$$\chi = \tan^{-1} \left(\frac{V_n}{V_e} \right), \quad (2.6)$$

where $V_n = V_a \cos \psi + w_n$, and $V_e = V_a \sin \psi + w_e$.

Using uncertainty analysis as in [24], the uncertainty in ground speed and course measurements can be approximated as

$$\sigma_{V_g} = \sqrt{\frac{V_n^2 \sigma_{V_n}^2 + V_e^2 \sigma_{V_e}^2}{V_n^2 + V_e^2}}$$

$$\sigma_\chi = \sqrt{\frac{V_n^2 \sigma_{V_e}^2 + V_e^2 \sigma_{V_n}^2}{(V_n^2 + V_e^2)^2}}.$$

If the uncertainty in the north and east axis have the same magnitude (i.e., $\sigma_{V_n} = \sigma_{V_e} = \sigma_V$), these formulas simplify as

$$\sigma_{V_g} = \sigma_V \quad (2.7)$$

$$\sigma_\chi = \frac{\sigma_V}{V_g}. \quad (2.8)$$

In these expressions we can observe that the uncertainty in the course measurements increases with the inverse of the ground speed with high speeds yielding small errors and low velocity creating big errors. The reason for this behavior is because the course angle is

only defined for moving objects. The ground speed and the course angle measurements of the GPS receiver can be modeled parting from (2.5)-(2.6) and (2.7)-(2.8)

$$y_{GPS,V_g} = \sqrt{(V_a \cos \psi + w_n)^2 + (V_a \sin \psi + w_e)^2} + \eta_V \quad (2.9)$$

$$y_{GPS,\chi} = \arctan^2(V_a \sin \psi + w_e, V_a \cos \psi + w_n) + \eta_\chi, \quad (2.10)$$

where η_V and η_χ are zero-mean Gaussian processes with variances $\sigma_{V_g}^2$ and σ_χ^2 .

2.2.2 GPS Smoothing

In this section we present an algorithm that estimates the position, ground speed, course, wind speed and heading of a fixed wing aircraft. Assuming that the flight path angle (γ) is zero, the position can be expressed as

$$p_n = V_g \cos \chi$$

$$p_e = V_g \sin \chi$$

where χ is the course angle of the UAS. To get the variation of the ground speed we must differentiate (2.5) getting

$$\begin{aligned} \dot{V}_g &= \frac{d}{dt} \sqrt{(V_a \cos \psi + w_n)^2 + (V_a \sin \psi + w_e)^2} \\ &= \frac{1}{V_g} [(V_a \cos \psi + w_n)(\dot{V}_a \cos \psi - V_a \dot{\psi} \sin \psi + \dot{w}_n) \\ &\quad + (V_a \sin \psi + w_e)(\dot{V}_a \sin \psi + V_a \dot{\psi} \cos \psi + \dot{w}_e)], \end{aligned}$$

where $\psi + w_n$ and $\psi + w_e$ are the wind speed in the north and the east direction respectively, and V_a is the air velocity. If we assume that the wind speed and airspeed are constant the evolution of the ground speed simplifies to

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

On the other hand, the evolution of χ is described as

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

and the dynamic of ψ is given by

$$\dot{\psi} = q \frac{\sin \phi}{\cos \phi} + r \frac{\cos \phi}{\cos \phi}.$$

The nonlinear propagation model is given by $\dot{x} = f(x, u)$, where x is the state and is defined as $x = (p_n, p_e, V_g, \chi, w_n, w_e, \psi)^\top$, and u is the input expressed as $u = (V_a, q, r, \phi, \theta)^\top$, and \dot{x} is defined as

$$\begin{pmatrix} \dot{p}_n \\ \dot{p}_e \\ \dot{V}_g \\ \dot{\chi} \\ \dot{w}_n \\ \dot{w}_e \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} V_g \cos \chi \\ V_g \sin \chi \\ \frac{(V_a \cos \psi + w_n)(-V_a \dot{\psi} \sin \psi) + (V_a \sin \psi + w_e)(V_a \dot{\psi} \cos \psi)}{V_g} \\ \frac{g}{V_g} \tan \phi \cos(\chi - \psi) \\ 0 \\ 0 \\ q \frac{\sin \phi}{\cos \phi} + r \frac{\cos \phi}{\cos \phi} \end{pmatrix}. \quad (2.11)$$

The Jacobian matrix, obtained by taking partial derivatives of the propagation model $\dot{x} = f(x, u)$ with respect to x , is expressed as

$$\frac{\partial f(x, u)}{\partial x} = \begin{pmatrix} 0 & 0 & \cos \chi & -V_g \sin \chi & 0 & 0 & 0 \\ 0 & 0 & \sin \chi & V_g \cos \chi & 0 & 0 & 0 \\ 0 & 0 & -\frac{\dot{V}_g}{V_g} & 0 & -\dot{\psi} V_a \sin \psi & \dot{\psi} V_a \cos \psi & \frac{\partial \dot{V}_g}{\partial \psi} \\ 0 & 0 & \frac{\partial \dot{\chi}}{\partial V_g} & \frac{\partial \dot{\chi}}{\partial \chi} & 0 & 0 & \frac{\partial \dot{\chi}}{\partial \psi} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (2.12)$$

where

$$\begin{aligned}\frac{\partial \dot{V}_g}{\partial \psi} &= \frac{-\left(q \frac{\sin \phi}{\cos \phi} + r \frac{\cos \phi}{\cos \phi}\right) V_a (w_n \cos \psi + w_e \sin \psi)}{V_g} \\ \frac{\partial \dot{\chi}}{\partial V_g} &= -\frac{g}{V_g^2} \tan \phi \cos(\chi - \psi) \\ \frac{\partial \dot{\chi}}{\partial \chi} &= -\frac{g}{V_g} \tan \phi \sin(\chi - \psi) \\ \frac{\partial \chi}{\partial \psi} &= \frac{g}{V_g} \tan \phi \sin(\chi - \psi).\end{aligned}$$

The input u is composed by the north position, east position, ground speed, and course supplied by the GPS receiver.

For measurements, we use the GPS signals for north and east position, ground speed and course. Due to the coupling of the states, we require the triangle relationship given by

$$V_g \begin{pmatrix} \cos \psi \cos \gamma \\ \sin \psi \cos \gamma \\ -\sin \gamma \end{pmatrix} - \begin{pmatrix} w_n \\ w_e \\ w_d \end{pmatrix} = V_a \begin{pmatrix} \cos \psi \cos \gamma_a \\ \sin \psi \cos \gamma_a \\ -\sin \gamma_a \end{pmatrix}.$$

If we assume that $\gamma = \gamma_a = 0$, the down component is discarded, and we are left with

$$V_a \cos \psi + w_n = V_g \cos \chi$$

$$V_a \sin \psi + w_e = V_g \sin \chi.$$

Using these equations we can define the pseudo measurements as

$$y_{wind,n} = V_a \cos \psi + w_n - V_g \cos \chi$$

$$y_{wind,e} = V_a \sin \psi + w_e - V_g \sin \chi,$$

The consequent measurement model is expressed as

$$y_{GPS} = h(x, u) + \eta_{GPS},$$

where $y_{GPS} = (y_{GPS,n}, y_{GPS,e}, y_{GPS,V_g}, y_{GPS,\chi}, y_{wind,n}, y_{wind,e})$, $u = \hat{V}_a$, and

$$h(x, u) = \begin{pmatrix} p_n \\ p_e \\ V_g \\ \chi \\ V_a \cos \psi + w_n - V_g \cos \chi \\ V_a \sin \psi + w_e - V_g \sin \chi \end{pmatrix}, \quad (2.13)$$

And the Jacobian, obtained by taking partial derivatives of the measurement model with respect to x , is given by

$$\frac{\partial h}{\partial x}(\hat{x}, u) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\cos \chi & V_g \sin \chi & 1 & 0 & -V_a \sin \psi \\ 0 & 0 & -\sin \chi & -V_g \cos \chi & 0 & 1 & V_a \cos \psi \end{pmatrix}. \quad (2.14)$$

The estimation of $p_n, p_e, V_g, \chi, w_n, w_e$ and ψ , are going to be calculated using two algorithms called the extended Kalman filter (EKF) and the delayed extended Kalman filter (DEKF), which are going to be explained in section 2.4.

2.3 Camera

This section introduces state estimation using camera data and the control of a gimbal. First, we address the coordinate frame geometry and describe the gimbal and camera frames. Then, the projection of a 3-D object into the 2-D image plane is presented. We also present a simple gimbal pointing algorithm which is implemented in a pan-tilt gimbal to point an antenna towards the UAS. Moreover, we discuss a geolocation algorithm used to estimate the position of an aircraft within its field of view. This study assumes that an image recognition algorithm exists which detects the UAS in the image data. This section is based on the work of [22].

2.3.1 Gimbal and Camera Frames and Projective Geometry

Assuming that the center of mass of the antenna pointer is the origin of the gimbal and camera frames, there are three frames of interest. The camera frame denoted by

$$\mathcal{F}^c = (\mathbf{i}^c, \mathbf{j}^c, \mathbf{k}^c).$$

The second frame of interest is obtained by rotating the body frame an angle of α_{az} , the azimuth angle, about the \mathbf{k}^b axis, resulting in the gimbal frame ($g1$). This frame is denoted by

$$\mathcal{F}^{g1} = (\mathbf{i}^{g1}, \mathbf{j}^{g1}, \mathbf{k}^{g1}).$$

The rotation matrix of this coordinate frame transformation is given by

$$\mathcal{R}_b^{g1}(\alpha_{az}) \triangleq \begin{pmatrix} \cos \alpha_{az} & \sin \alpha_{az} & 0 \\ -\sin \alpha_{az} & \cos \alpha_{az} & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (2.15)$$

Finally, if the gimbal-1 frame is rotated an angle of α_{el} , the elevation angle, in the \mathbf{j}^{g1} axis, the third frame of interest is obtained, which is the gimbal frame. This transformation is

described by

$$\mathcal{R}_{g1}^g(\alpha_{el}) \triangleq \begin{pmatrix} \cos \alpha_{el} & 0 & -\sin \alpha_{el} \\ 0 & 1 & 0 \\ \sin \alpha_{el} & 0 & \cos \alpha_{el} \end{pmatrix}. \quad (2.16)$$

To obtain the rotation matrix from the body frame to the gimbal frame we just need to multiply the matrices in 2.15 and 2.16, resulting in

$$\mathcal{R}_b^g = \mathcal{R}_{g1}^g \mathcal{R}_b^{g1} = \begin{pmatrix} \cos \alpha_{el} \cos \alpha_{az} & \cos \alpha_{el} \sin \alpha_{az} & -\sin \alpha_{el} \\ -\sin \alpha_{az} & \cos \alpha_{az} & 0 \\ \sin \alpha_{el} \cos \alpha_{az} & \sin \alpha_{el} \sin \alpha_{az} & \cos \alpha_{el} \end{pmatrix}. \quad (2.17)$$

The convention in computer vision and image processing for the camera frame is that the \mathbf{i}^c axis points to the right of the image, \mathbf{j}^c axis points down of the image, and \mathbf{k}^c axis points along the optical axis. Therefore, the transformation from the gimbal frame to the camera frame is given by

$$\mathcal{R}_g^c = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}. \quad (2.18)$$

2.3.2 Camera Model

Lets assume that the pixels and the pixel array are square. Defining M as the width of the square pixel array and v as the field-of-view of the camera, then f which is the focal length, is given by

$$f = \frac{M}{2 \tan \frac{v}{2}}. \quad (2.19)$$

The position of the UAS projected into the camera frame is given by $(P_{\epsilon_x}, P_{\epsilon_y}, P_f)$, where ϵ_x and ϵ_y are the pixel location of the aircraft in units of pixels. The distance between the

origin of the camera frame and the location of the plane (ϵ_x, ϵ_y) , is PF where

$$F = \sqrt{f^2 + \epsilon_x^2 + \epsilon_y^2}. \quad (2.20)$$

Using basic trigonometry we get

$$\frac{l_x^c}{\mathbb{L}} = \frac{P_{\epsilon_x}}{PF} = \frac{\epsilon_x}{F} \quad (2.21)$$

$$\frac{l_y^c}{\mathbb{L}} = \frac{\epsilon_y}{F} \quad (2.22)$$

$$\frac{l_f^c}{\mathbb{L}} = \frac{f}{F}. \quad (2.23)$$

Defining \mathbf{l}^c as the vector to the UAS and $\mathbb{L} = \|\mathbf{l}\|$ we can express (2.21) through (2.23) as

$$\mathbf{l}^c = \frac{\mathbb{L}}{F} \begin{pmatrix} \epsilon_x \\ \epsilon_y \\ f \end{pmatrix}, \quad (2.24)$$

since \mathbb{L} is unknown, \mathbf{l}^c cannot be calculated only with camera data. Even so, the unit vector that points towards the aircraft can be obtained as

$$\frac{\mathbf{l}^c}{\mathbb{L}} = \frac{1}{F} \begin{pmatrix} \epsilon_x \\ \epsilon_y \\ f \end{pmatrix} = \frac{1}{\sqrt{\epsilon_x^2 + \epsilon_y^2 + f^2}} \begin{pmatrix} \epsilon_x \\ \epsilon_y \\ f \end{pmatrix}. \quad (2.25)$$

Since this unit vector is used multiple times throughout this section, we use the notation

$$\check{\mathbf{l}} \triangleq \begin{pmatrix} \check{l}_x \\ \check{l}_y \\ \check{l}_z \end{pmatrix} \triangleq \frac{\mathbf{l}^c}{\mathbb{L}}.$$

2.3.3 Gimbal Pointing

In this section we present a basic gimbal-pointing algorithm, which is used by our antenna pointer. The tracking system is assumed to pan and tilt with the respective angles azimuth and elevation. The dynamic model of the gimbal is assumed as

$$\dot{\alpha}_{az} = u_{az}$$

$$\dot{\alpha}_{el} = u_{el}.$$

The purpose of this algorithm is to point an antenna mounted in the gimbal to a given position, in our case the aircraft location. Let \mathbf{p}_{obj}^i be the position of the UAV in the inertial frame. Our goal is to center the aircraft in the image plane, therefore aligning the optical axis of the camera with the desired relative position vector

$$\mathbf{l}_d^i \triangleq \mathbf{p}_{UAV}^i - \mathbf{p}_{AP}^i,$$

where $\mathbf{p}_{AP}^i = (p_n, p_e, p_d)^\top$ is the inertial position of the antenna pointer and the subscript $_d$ indicates a desired quantity. The unit vector that targets the UAS in the body-frame is

$$\check{\mathbf{l}}_d^b = \frac{1}{\|\mathbf{l}_d^i\|} \mathcal{R}_i^b \mathbf{l}_d^i.$$

Furthermore, we need to calculate the desired azimuth and elevation angles so that the UAS is located in the origin of the image plane, this is, to align the optical axis with $\check{\mathbf{l}}_d^b$. The next step is to determine the desired azimuth and elevation angles that aligns the optical axis with $\check{\mathbf{l}}_d^b$. Since the optical axis is given by $(0, 0, 1)^c$, the c denoting that is in the camera

frame, the commanded gimbal angles α_{az}^c and α_{el}^c are

$$\tilde{\mathbf{l}}_d^b \triangleq \begin{pmatrix} \tilde{l}_{xd}^b \\ \tilde{l}_{yd}^b \\ \tilde{l}_{zd}^b \end{pmatrix} = \mathcal{R}_g^b(\alpha_{az}^c, \alpha_{el}^c) \mathcal{R}_c^g \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad (2.26)$$

$$= \begin{pmatrix} \cos \alpha_{el}^c \cos \alpha_{az}^c & -\sin \alpha_{el}^c & -\sin \alpha_{el}^c \cos \alpha_{az}^c \\ \cos \alpha_{el}^c \sin \alpha_{az}^c & \cos \alpha_{az}^c & -\sin \alpha_{el}^c \sin \alpha_{az}^c \\ \sin \alpha_{el}^c & 0 & \cos \alpha_{el}^c \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad (2.27)$$

$$= \begin{pmatrix} \cos \alpha_{el}^c \cos \alpha_{az}^c \\ \cos \alpha_{el}^c \sin \alpha_{az}^c \\ \sin \alpha_{el}^c \end{pmatrix}. \quad (2.28)$$

Therefore the desired azimuth and elevation angles are

$$\alpha_{az}^c = \tan^{-1} \left(\frac{\tilde{l}_{yd}^b}{\tilde{l}_{xd}^b} \right) \quad (2.29)$$

$$\alpha_{el}^c = \sin^{-1} \left(\tilde{l}_{zd}^b \right). \quad (2.30)$$

Then, choosing the gimbal servo commands we have

$$u_{az} = k_{az}(\alpha_{az}^c - \alpha_{az}) \quad (2.31)$$

$$u_{el} = k_{el}(\alpha_{el}^c - \alpha_{el}), \quad (2.32)$$

where k_{az} and k_{el} are positive control gains.

2.3.4 Geolocation

The objective of this section is to describe an algorithm that can determine the position of an aircraft in inertial coordinates using a EO/IR camera mounted in the antenna pointer. For this study we assume that the antenna pointer can measure its own position. Continuing the discussion of section 2.3.1, lets define $\mathbf{l} = \mathbf{p}_{UAV} - \mathbf{p}_{AP}$ as the relative position vector between the aircraft and the antenna pointer, $\mathbb{L} = \|\mathbf{l}\|$ as the range to the aircraft, and $\check{\mathbf{l}} = \mathbf{l}/\mathbb{L}$ as the unit vector that points towards the UAV. Using simple geometry, we have that

$$\begin{aligned}\mathbf{p}_{UAV}^i &= \mathbf{p}_{AP}^i + \mathcal{R}_b^i \mathcal{R}_g^b \mathcal{R}_c^g \mathbf{l}^c \\ &= \mathbf{p}_{AP}^i + \mathbb{L}(\mathcal{R}_b^i \mathcal{R}_g^b \mathcal{R}_c^g \check{\mathbf{l}}^c),\end{aligned}\tag{2.33}$$

where $\mathbf{p}_{AP}^i = (p_n, p_e, p_d)^\top$, $\mathcal{R}_b^i = \mathcal{R}_b^i(\phi, \theta, \psi)$, and $\mathcal{R}_g^b = \mathcal{R}_g^b(\alpha_{az}, \alpha_{el})$. Note that all the elements on the right-hand side of (2.33) are known, except \mathbb{L} . Hence, the geolocation of a target is simply as estimating the range to the target \mathbb{L} .

Range to Target Using the Flat-Earth Model

Since we assumed that the antenna pointer can determine its position, we have available the height-above-ground of the pointer, h . Also since UAS missions are not usually long range we can neglect the curvature of the earth, and assume a flat-earth model. Then using simple trigonometry we have that

$$\mathbb{L} = \frac{h}{\cos \varphi},$$

where

$$\begin{aligned}\cos \varphi &= \mathbf{k}^i \cdot \check{\mathbf{l}}^i \\ &= \mathbf{k}^i \cdot \mathcal{R}_b^i \mathcal{R}_g^b \mathcal{R}_c^g \check{\mathbf{l}}^c.\end{aligned}$$

Subsequently, the estimation of the range to the target is given by

$$\mathbb{L} = \frac{h}{\mathbf{k}^i \cdot \mathcal{R}_b^i \mathcal{R}_g^b \mathcal{R}_c^g \check{\mathbf{l}}^c}. \quad (2.34)$$

And finally combining (2.33) and (2.34) the geolocation estimate can be expressed as

$$\mathbf{p}_{UAV}^i = \begin{pmatrix} p_n \\ p_e \\ p_d \end{pmatrix} + h \frac{\mathcal{R}_b^i \mathcal{R}_g^b \mathcal{R}_c^g \check{\mathbf{l}}^c}{\mathbf{k}^i \cdot \mathcal{R}_b^i \mathcal{R}_g^b \mathcal{R}_c^g \check{\mathbf{l}}^c} \quad (2.35)$$

Geolocation Using an Extended Kalman filter

Since the geolocation estimation made by (2.35) provides an approximation of the location of the UAS, and does not take into account the past measurements, it is prone to errors. Therefore, in this section we present a method to solve the geolocation problem iteratively, the extended Kalman filter(EKF). This algorithm and its derivation are explained in section 2.4.

If we assume that the antenna pointer is stationary, we have

$$\dot{\mathbf{p}}_{AP}^i = 0.$$

Since $\mathbb{L} = \|\mathbf{p}_{UAV}^i - \mathbf{p}_{AP}^i\|$, we have

$$\begin{aligned} \dot{\mathbb{L}} &= \frac{d}{dt} \sqrt{(\mathbf{p}_{UAV}^i - \mathbf{p}_{AP}^i)^\top (\mathbf{p}_{UAV}^i - \mathbf{p}_{AP}^i)} \\ &= \frac{(\dot{\mathbf{p}}_{UAV}^i - \dot{\mathbf{p}}_{AP}^i)^\top (\mathbf{p}_{UAV}^i - \mathbf{p}_{AP}^i)}{\mathbb{L}} \\ &= \frac{(\mathbf{p}_{UAV}^i - \mathbf{p}_{AP}^i)^\top \dot{\mathbf{p}}_{UAV}^i}{\mathbb{L}}, \end{aligned}$$

If a constant-altitude flight is assumed, we can approximate $\dot{\mathbf{p}}_{UAV}^i$ as

$$\dot{\mathbf{p}}_{UAV}^i = \begin{pmatrix} \hat{V}_g \cos \hat{\chi} \\ \hat{V}_g \sin \hat{\chi} \\ 0 \end{pmatrix},$$

and where \hat{V}_g and $\hat{\chi}$ are calculated using the EKF and the DEKF discussed in the next section. The input to the geolocation algorithm is the position of the antenna pointer in the inertial frame and the estimate of the normalized line-of-sight vector as given in (2.25).

To implement the geolocation problem with an Extended Kalman filter, we define the state as $\hat{x} = (\hat{\mathbf{p}}_{UAV}^{i\top}, \hat{\mathbb{L}})^\top$ and prediction equations given by

$$\begin{pmatrix} \dot{\mathbf{p}}_{UAV}^i \\ \dot{\mathbb{L}} \end{pmatrix} = \begin{pmatrix} 0 \\ \frac{(\mathbf{p}_{UAV}^i - \mathbf{p}_{AP}^i)^\top \dot{\mathbf{p}}_{UAV}^i}{\mathbb{L}^2} \end{pmatrix}.$$

Taking partial derivatives of f with respect to x , we have the Jacobian which is described by

$$\frac{\partial f}{\partial x} = \begin{pmatrix} 0 & 0 \\ \frac{\dot{\mathbf{p}}_{UAV}^i}{\mathbb{L}} & -\frac{(\mathbf{p}_{UAV}^i - \mathbf{p}_{AP}^i)^\top \dot{\mathbf{p}}_{UAV}^i}{\mathbb{L}^2} \end{pmatrix}.$$

The output equation is given by (2.33), and its Jacobian is given by

$$\frac{\partial h}{\partial x} = \begin{pmatrix} 0 & \mathcal{R}_b^i \mathcal{R}_g^b \mathcal{R}_c^g \check{\mathbf{l}}^c \end{pmatrix}.$$

2.4 Kalman filter

A Kalman filter is an optimal recursive data processing algorithm. One of the aspects of this optimality is that the Kalman filter incorporates all the information that can be provided to it. It processes all available measurements, regardless of their precision, to estimate the current value of the variables of interest, with use of knowledge of the system and measurement device dynamics, the statistical description of the system noises, measurement errors and uncertainty in the dynamics models, and any available information about initial conditions of the variables of interest. One of the interesting characteristics of the Kalman filter is that it does not require all previous data to be kept in memory and reprocessed every time a new measurement is taken, making the filter implementation more practical. The filter is actually a data processing algorithm, and therefore it incorporates discrete-time measurement samples rather than continuous time inputs. Often the variables of interest, some finite number of quantities to describe the state of the system, cannot be measured directly, and some means of inferring these values from the available data must be generated, thus the need of the filter. This inference is complicated by the facts that the system is typically driven by inputs other than our own known controls and that the relationships among the various state variables and measured outputs are known only with some degree of uncertainty. Furthermore, any measurement is corrupted to some degree by noise, biases, and device inaccuracies, and so a means of extracting valuable information from a noisy signal must be provided as well. There may also be a number of different measuring devices, each with its own particular dynamics and error characteristics, that provide some information about a particular variable, and it would be desirable to combine their outputs in a systematic and optimal manner. A Kalman filter combines all available measurement data, plus prior knowledge about the system and measuring devices, to produce an estimate of the desired variables in such a manner that the error is minimized statistically. In other words, if we were to run a number of candidate filters many times for some application, then the average results of the Kalman filter would be better than the average results of any other [25].

2.4.1 Discrete Kalman filter

This section is based on the work of [22]. There are several different forms of the Kalman filter, but the form particularly useful for Small UAS applications is the continuous-propagation, discrete-measurement Kalman filter. Lets assume that the linear system dynamics are given by

$$\begin{aligned}\dot{x} &= Ax + Bu + \xi \\ y[n] &= Cx[n] + \eta[n],\end{aligned}\tag{2.36}$$

where $y[n] = y(t_n)$ is the n^{th} sample of y , $x[n] = x(t_n)$ is the n^{th} sample of x , η is the measurement noise at time t_n , ξ is a zero-mean Gaussian random process with covariance Q , and $\eta[n]$ is a zero-mean Gaussian random variable with covariance R . The random process ξ is called the process noise and represents modeling error and disturbances on the system. The random variable η is called the measurement noise and represents noise on the sensors. The covariance R can usually be estimated from sensor calibration, but the covariance Q is generally unknown and therefore becomes a system parameter that can be tuned to improve the performance of the observer. Note that the sample rate does not need to be fixed. The continuous-discrete Kalman filter has the form

$$\dot{\hat{x}} = A\hat{x} + Bu\tag{2.37}$$

$$\hat{x}^+ = \hat{x}^- + L(y(t_n) - C\hat{x}^-),\tag{2.38}$$

where L is the Kalman gain. Define the estimation error as $\tilde{x} = x - \hat{x}$. The covariance of the estimation error at time t is given by

$$P(t) \triangleq E[\tilde{x}(t)\tilde{x}(t)^\top].\tag{2.39}$$

Note that $P(t)$ is symmetric and positive semi-definite, therefore, its eigenvalues are real and non-negative. Also, small eigenvalues of $P(t)$ imply small variance, which implies low average estimation error. Therefore, we would like to choose $L(t)$ to minimize the

eigenvalues of $P(t)$. Recall that

$$\text{tr}(P) = \sum_{i=1}^n \lambda_i,$$

where $\text{tr}(P)$ is the trace of P , and λ_i are the eigenvalues of P . Therefore, minimizing $\text{tr}(P)$ minimizes the estimation error covariance. The Kalman filter is derived by finding L to minimize $\text{tr}(P)$. The equations of the Kalman filter can be categorized into two groups: *time update equations* and *measurement update equations*. The first is responsible for projecting forward in time the current state and error covariance estimates to obtain a *priori* estimate for the next time step. The measurement update equations are responsible for incorporating a new measurement into the *a priori* estimate to obtain an improved *a posteriori* estimate.

Time Update Equations

Differentiating \tilde{x} we get

$$\begin{aligned} \dot{\tilde{x}} &= \dot{x} - \dot{\hat{x}} \\ &= Ax + Bu + \xi - A\hat{x} - Bu \\ &= A\tilde{x} + \xi. \end{aligned}$$

Solving the differential equation with initial condition \tilde{x}_0 we obtain

$$\tilde{x}(t) = e^{At}\tilde{x}_0 + \int_0^t e^{A(t-\tau)}\xi(\tau)d\tau.$$

Computing the evolution of P , the error covariance we get

$$\begin{aligned}
 \dot{P} &= \frac{d}{dt} E \tilde{x} \dot{\tilde{x}}^\top \\
 &= E(\dot{\tilde{x}} \tilde{x}^\top + \tilde{x} \dot{\tilde{x}}^\top) \\
 &= E(A \tilde{x} \tilde{x}^\top + \xi \tilde{x}^\top + \tilde{x} \tilde{x}^\top A^\top + \tilde{x} \xi^\top) \\
 &= AP + PA^\top + E(\xi \tilde{x}^\top) + E(\tilde{x} \xi^\top)
 \end{aligned}$$

Calculating $E(\tilde{x} \xi^\top)$ as

$$\begin{aligned}
 E(\tilde{x} \xi^\top) &= E(e^{At} \tilde{x}_0 \xi^\top(t) + \int_0^t e^{A(t-\tau)} \xi(\tau) \xi^\top(\tau) d\tau) \\
 &= \int_0^t e^{A(t-\tau)} Q \delta(t-\tau) d\tau \\
 &= \frac{1}{2} Q,
 \end{aligned}$$

where the $\frac{1}{2}$ is because we only use half of the area inside the delta function. Therefore, since Q is symmetric we have that P evolves between measurements as

$$\dot{P} = AP + PA^\top + Q.$$

Measurement Update Equations

When a measurement is received, we have that

$$\begin{aligned}
 \tilde{x}^+ &= x - \hat{x}^+ \\
 &= x - \hat{x}^- - L(Cx + \eta - C\hat{x}^-) \\
 &= \tilde{x}^- - LC\tilde{x}^- - L\eta.
 \end{aligned}$$

On the other hand we have that

$$\begin{aligned}
P^+ &= E(\tilde{x}^+ \tilde{x}^{+\top}) \\
&= E[(\tilde{x}^- - LC\tilde{x}^- - L\eta)(\tilde{x}^- - LC\tilde{x}^- - L\eta)^\top] \\
&= E[\tilde{x}^- \tilde{x}^{-\top} - \tilde{x}^- \tilde{x}^{-\top} C^\top L^\top - \tilde{x}^- \eta^\top L^\top \\
&\quad - LC\tilde{x}^- \tilde{x}^{-\top} + LC\tilde{x}^- \tilde{x}^{-\top} C^\top L^\top + LC\tilde{x}^- \eta^\top L^\top \\
&\quad - L\eta \tilde{x}^{-\top} + L\eta \tilde{x}^{-\top} C^\top L^\top + L\eta \eta^\top L^\top] \\
&= P^- - P^- C^\top L^\top - LCP^- + LCP^- C^\top L^\top + LRL^\top.
\end{aligned} \tag{2.40}$$

Since η and \tilde{x}^- are independent, $E(\tilde{x}^- \eta^\top L^\top) = E(L\eta \tilde{x}^{-\top}) = 0$.

To continue our derivation, the following matrix relationship is required:

$$\begin{aligned}
\frac{\partial}{\partial A} \text{tr}(BAD) &= B^\top D^\top \\
\frac{\partial}{\partial A} \text{tr}(BAA^\top) &= 2AB, \text{ if } B = B^\top.
\end{aligned}$$

Our goal is to minimize $\text{tr}(P^+)$ by choosing L . A required condition is that

$$\begin{aligned}
\frac{\partial}{\partial L} \text{tr}(P^+) &= -P^- C^\top - P^- C^\top + 2LCP^- C^\top + 2LR = 0 \\
&\Rightarrow 2L(R + CP^- C^\top) = 2P^- C^\top \\
&\Rightarrow L = P^- C^\top (R + CP^- C^\top)^{-1}.
\end{aligned}$$

If we substitute into (2.40) we get

$$\begin{aligned}
P^+ &= P^- + P^- C^\top (R + CP^- C^\top)^{-1} CP^- - P^- C^\top (R + CP^- C^\top)^{-1} CP^- \\
&\quad + P^- C^\top (R + CP^- C^\top)^{-1} (CP^- C^\top + R) (R + CP^- C^\top)^{-1} CP^- \\
&= P^- - P^- C^\top (R + CP^- C^\top)^{-1} CP^- \\
&= (I - P^- C^\top (R + CP^- C^\top)^{-1} C) P^- \\
&= (I - LC) P^-.
\end{aligned}$$

Summarizing the Kalman filter we have two sets of equations. The time update equations, which propagate the state estimates

$$\dot{\hat{x}} = A\hat{x} + Bu \quad (2.41)$$

$$\dot{P} = AP + PA^\top + Q, \quad (2.42)$$

where \hat{x} is the estimate of the state, and P is the symmetric covariance matrix of the estimation error. The second set of equations, the measurement update equations, are used when a measurement is received from the i^{th} sensor, which updates the state estimates and error covariance with the following equations

$$L_i = P^- C_i^\top (R_i + C_i P^- C_i^\top)^{-1} \quad (2.43)$$

$$P^+ = (I - L_i C_i) P^- \quad (2.44)$$

$$\hat{x}^+ = \hat{x}^- + L_i (y_i(t_n) - C_i \hat{x}^-), \quad (2.45)$$

where L_i is called the Kalman gain for the i^{th} sensor.

2.4.2 Extended Kalman filter

This section is based on the work of [22]. In the previous section we assumed that the system propagation model and measurement model are linear. However, for many applications, including the one used in this thesis, the system propagation model and the measurement model are nonlinear. Therefore the model in (2.36) becomes

$$\hat{x} = f(x, u) + \xi \quad (2.46)$$

$$y[n] = h(x[n], u[n]) + \eta[n], \quad (2.47)$$

This case is called the extended Kalman filter. For the EKF, the state propagation and update equations use the nonlinear model, but the propagation and update of the error covariance use the Jacobian of f for A , and the Jacobian of h for C .

Pseudo-code for the continuous discrete extended Kalman filter is given below

- Initialize: $\hat{x} = x_0$.
- Pick an output sample rate T_{out} that is less than the sample rates of the sensors.
- At each sample time T_{out} :
- **for** $i = 1$ to N **do** [Time Update Equation]
 - $\hat{x} = \hat{x} + \frac{T_{out}}{N} f(\hat{x}, u)$
 - $A = \frac{\partial f}{\partial x}(\hat{x}, u)$
 - $P = P + \frac{T_{out}}{N}(AP + PA^\top + Q)$
- **end for**
- **if** Measurement has been received from sensor i **then** [Measurement Update Equation]
 - $C_i = \frac{\partial h_i}{\partial x}(\hat{x}, u[n])$
 - $L_i = PC_i^\top (R_i + C_i PC_i^\top)^{-1}$
 - $P = (I - L_i C_i)P$
 - $\hat{x} = \hat{x} + L_i(y_i[n] - h(\hat{x}, u[n]))$
- **end if**

2.4.3 Delayed Extended Kalman filter

In real world applications, measurements usually have processing and communication delays which cannot be ignored. These delays can have big variations, consequently some measurements can arrive in an out of sequence fashion, which worsen the estimations if not handled properly.

In this section we show a modification of the extended Kalman filter, which enables it to handle delayed measurements and the OOSM problem. This filter works by introducing the measurement in the estimation as soon as they arrive, but using it to estimate the states at the time that the measurement was made. After that, this estimation is propagated forward in time. If another measurement more recent is available, then it is incorporated into the estimation and the states are propagated again in time. This is done until the filter is on the current time. Therefore, opposed to how the filter works in [2], this filter can run in real time. For this, we need to store the state estimations, the error covariance ma-

trices and the delayed measurements for as many time steps as the maximum probable delay.

Pseudo-code for the continuous discrete delayed extended Kalman filter is given below

- Initialize: $\hat{x} = x_0$.
- Pick an output sample rate T_{out} that is less than the sample rates of the sensors.
- At each sample time T_{out} :
- $[\hat{x}, P] = propagate(\hat{x}, P, N)$ [Time Update Equation]
- **if** Measurement has been received from GPS sensor **then**
 - Organize measurements (newest first)
 - $[\hat{x}, P] = (\hat{x}_{t_{meas_{new}}}, P_{t_{meas_{new}}})$
 - **for** $j = \text{position of new measurement to } 0$ **do**
 - $[\hat{x}, P] = update(\hat{x}, P, meas_j)$ [Measurement Update Equation]
 - **if** $j > 1$ **then**
 - $N = 100 (t_{meas_j} - t_{meas_{j-1}})$
 - **else**
 - $N = 100 (t_{current} - t_{meas_j})$
 - **end if**
 - $[\hat{x}, P] = propagate(\hat{x}, P, N)$
 - **end for**
- **end if**
- **if** Measurement has been received from camera **then**
 - $[\hat{x}, P] = update(\hat{x}, P, meas_{camera})$
- **end if**

Where the *propagate* function represents the Time Update Equation and is described as

- **function** $[\hat{x}, P] = propagate(\hat{x}, P, N)$
- **for** $i = 1$ to N **do**
 - $\hat{x} = \hat{x} + \frac{T_{out}}{N} f(\hat{x}, u)$

```

·       $A = \frac{\partial f}{\partial x}(\hat{x}, u)$ 

·       $P = P + \frac{T_{out}}{N}(AP + PA^\top + Q)$ 

·      end for

· end function

```

And the *update* function, which is the Measurement Update Equation, is represented by

```

· function  $[\hat{x}, P] = \text{update}(\hat{x}, P, meas)$ 

·       $C = \frac{\partial h}{\partial x}(\hat{x}, u[n])$ 

·       $L = PC^\top(R + CPC^\top)^{-1}$ 

·       $P = (I - LC)P$ 

·       $\hat{x} = \hat{x} + L(y[n] - h(\hat{x}, u[n]))$ 

· end function

```

CHAPTER 3

SIMULATION

3.1 Simulation Overview

This simulation was developed by [26] using MATLAB and Simulink, for the purpose of tracking several ground targets. This was achieved by using a camera mounted in an aircraft and pointing it in an optimal fashion to improve the estimation of the states of all the objectives. A block was added to provide the means of pointing an antenna and a camera towards a UAV, estimating the states with a Delayed Extended Kalman filter. In this study we are not concerned about the behavior of the plane, since we will point at it as long as it is within the line-of-sight. The simulation has four main blocks: UAV Sensor Block, Target Selection Step, Antenna Pointer block, and 3-D Plot block. We will only focus on the block that we developed, the Antenna Pointer block.

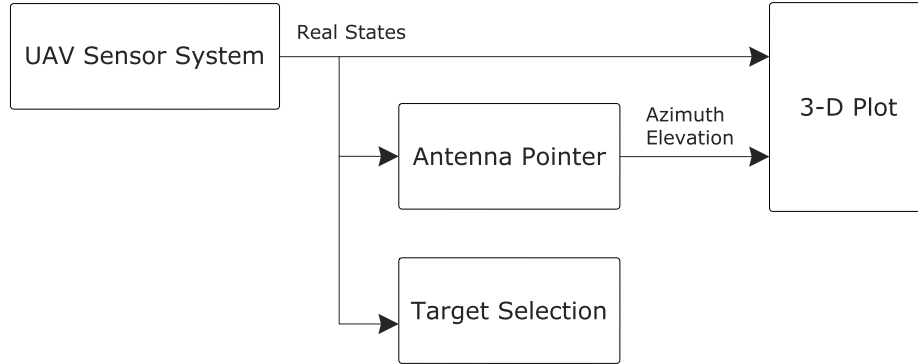


Fig. 3.1: Antenna Pointer Block

3.2 Antenna Pointer

The antenna pointer block precisely points an antenna and a camera towards the aircraft. It consists of four blocks: GPS, camera, EKF/DEKF and a pointer block as shown below.

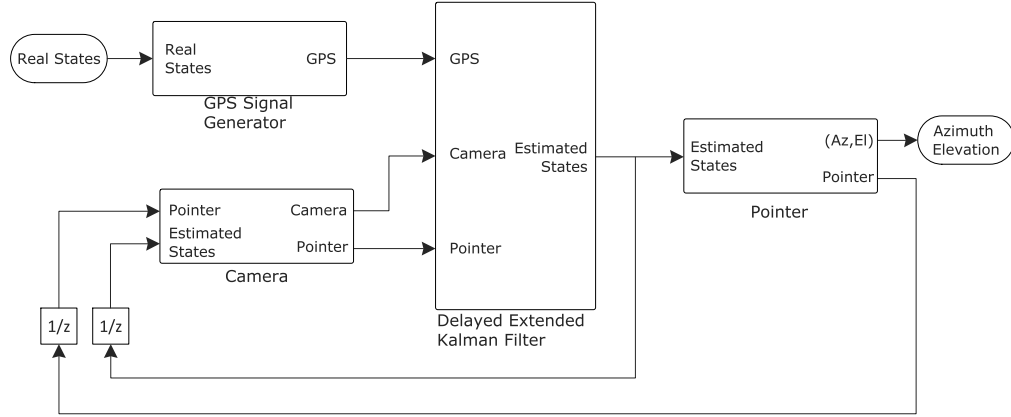


Fig. 3.2: Antenna Pointer Block. The interaction of each sub-block of the Antenna Pointer Block is shown here. The GPS receives the real measurements, adds noise and delays them. The camera receives the estimated states of the plane and if in the field of view send the position to the DEKF. The filter process all the data and outputs the estimation of the states. Finally the Pointer calculates the azimuth and elevation angles required to point towards the UAV.

3.2.1 GPS

This block generates the GPS measurements based in the real states received from the UAV sensor block. First, it reduced the frequency of the data from 100 Hz to 4 Hz, which is the common frequency found in modern GPS. Afterwards, it adds noise to the measurements as described in section 2.2.1. Finally, it waits a random time to send each measurement to the EKF/DEKF block where they are implemented in the estimation.

3.2.2 Camera

The camera block simulates a camera mounted in the antenna pointer working at 10 Hz and 20 Hz frequency. If the aircraft is in the field of view of the camera, the position in the camera frame is sent to the EKF/DEKF to introduce this data to the estimation of the states. In this simulation we assume that there is an algorithm which detects the UAV as long as it is in the field of view of the camera. The model of this camera is explained in section 2.3.2.

3.2.3 EKF/DEKF

This block process the information sent by the GPS and camera blocks to estimate the states of the aircraft. The processing of this data is done by the Delayed Extended Kalman filter, defined in section 2.4.3, which compensates for the delays in the GPS measurements. For comparison purposes the Extended Kalman filter defined in section 2.4.2 is used, to see how the estimates are impacted if we disregards the delays in the measurements. In this work, we assume that the delays in the camera measurements can be neglected since the processing unit and the camera are connected directly.

3.2.4 Pointer

This block calculates the angles to command and the controller of the antenna pointer. First, the pointer uses the estimated states sent by the EKF/DEKF block to calculate the azimuth and elevation angles required to point towards the aircraft. Next, the angles are sent to a simple proportional controller, defined by equations (2.31)-(2.31), which actuates the servos in the antenna pointer and feedbacks the actual angles to the camera and the EKF/DEKF blocks.

3.3 3-D Plot

The purpose of this block is to collect all the data and plot the aircraft, the antenna pointer and the field of view of the camera mounted in the antenna pointer in real time.

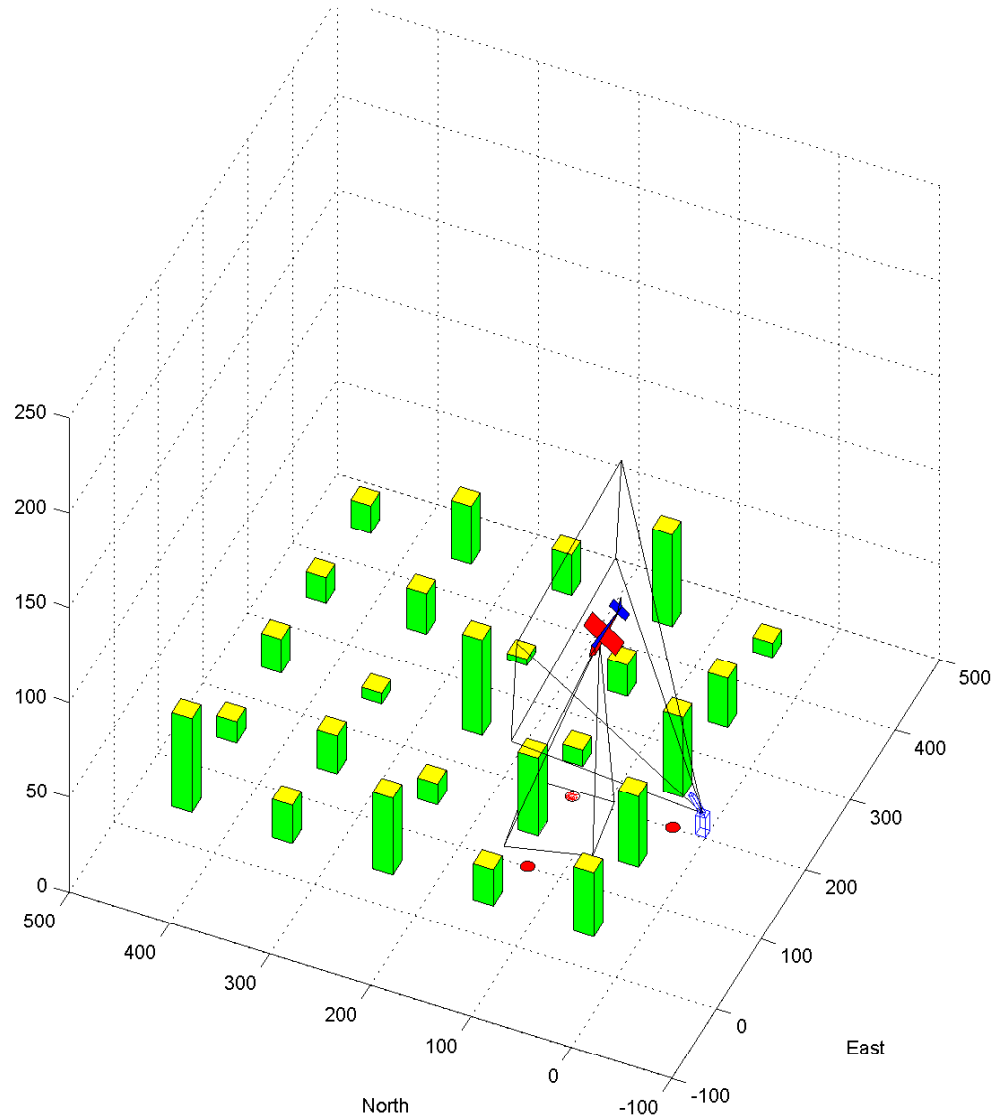


Fig. 3.3: Antenna Pointer tracking a UAV. This figure shows a UAV tracking different ground targets using a camera. At the same time, our antenna pointer system, follows the aircraft pointing an antenna and a camera towards it. This graphic was generated using Matlab and Simulink.

CHAPTER 4

RESULTS

This chapter presents the results gathered by the simulations explained in chapter 3. Three different sets of comparisons are presented. The first comparison shows the difference in accuracy of using a GPS sensor and a GPS coupled with a camera. The second case is to show how the delays affect the accuracy of the estimations. And the last case compares the estimations of the Extended Kalman filter against the Delayed Extended Kalman filter with delayed measurements, to see the impact of disregarding the delays.

4.1 Comparison of Estimations Between GPS and GPS Coupled with Camera

In this scenario, we focus on the comparison between using a GPS at 1 Hz coupled with a camera at 20 Hz, and the GPS alone. The introduction of the camera measurements are expected to improve the estimation. In this case, the GPS measurements have a random delay of 0.75 seconds.

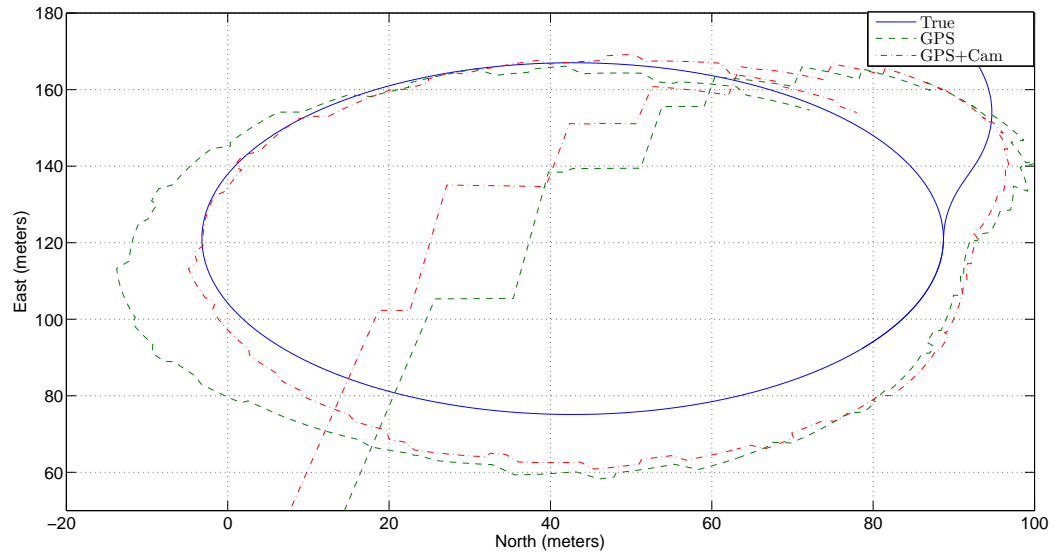


Fig. 4.1: UAV Trajectory Comparison - GPS vs GPS and Camera. The real trajectory of the aircraft is shown in this plot, against the estimated trajectory calculated by using a GPS coupled with a camera, and just a GPS.

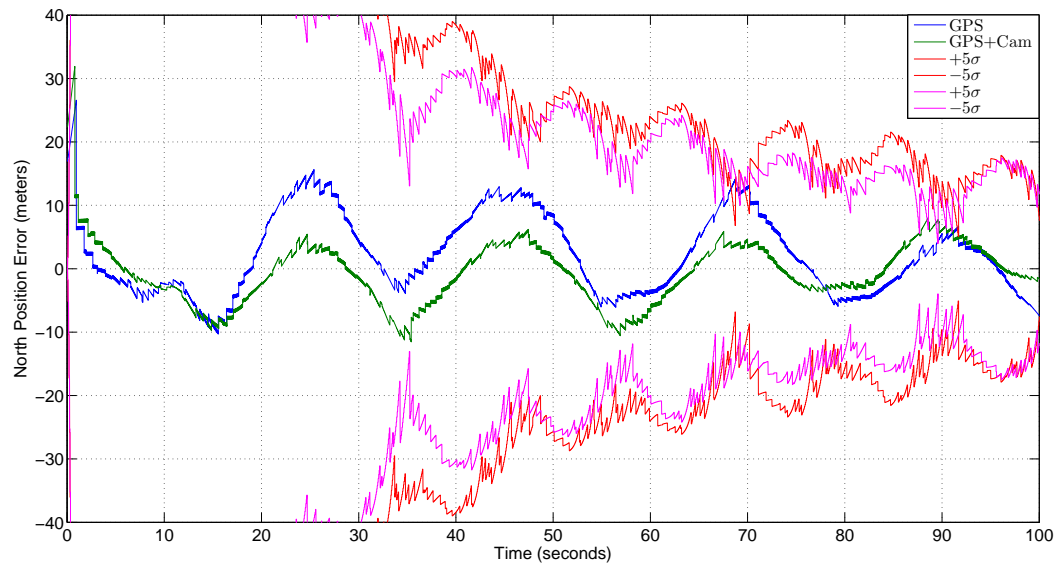


Fig. 4.2: North Position Error Comparison - GPS vs GPS and Camera. The error in the north position is shown in this plot. The states estimations are compared for GPS and GPS coupled with a camera.

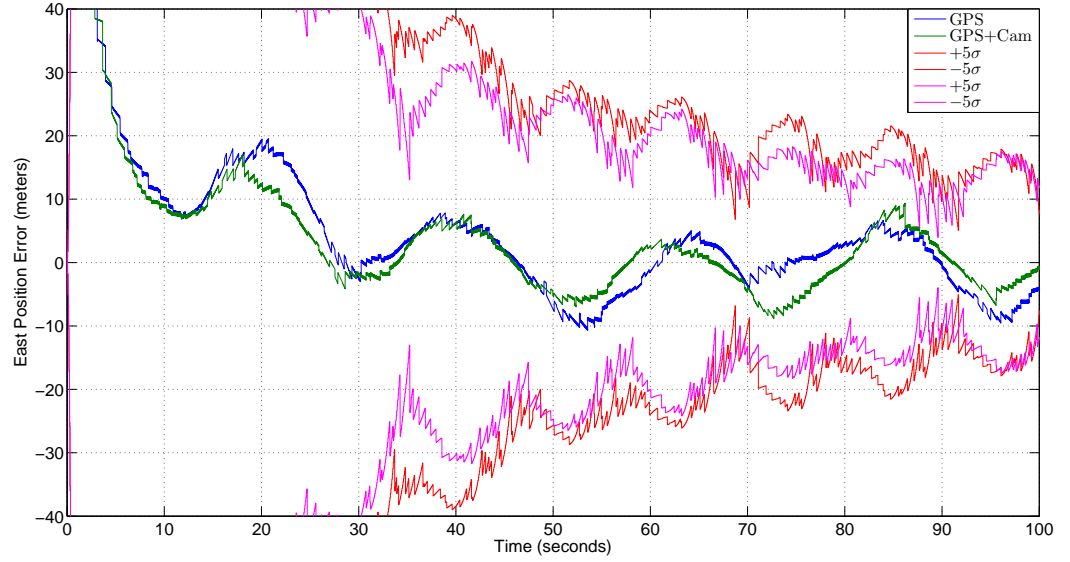


Fig. 4.3: East Position Error Comparison - GPS vs GPS and Camera. The error in the east position is shown in this plot. The states estimations are compared for GPS and GPS coupled with a camera.

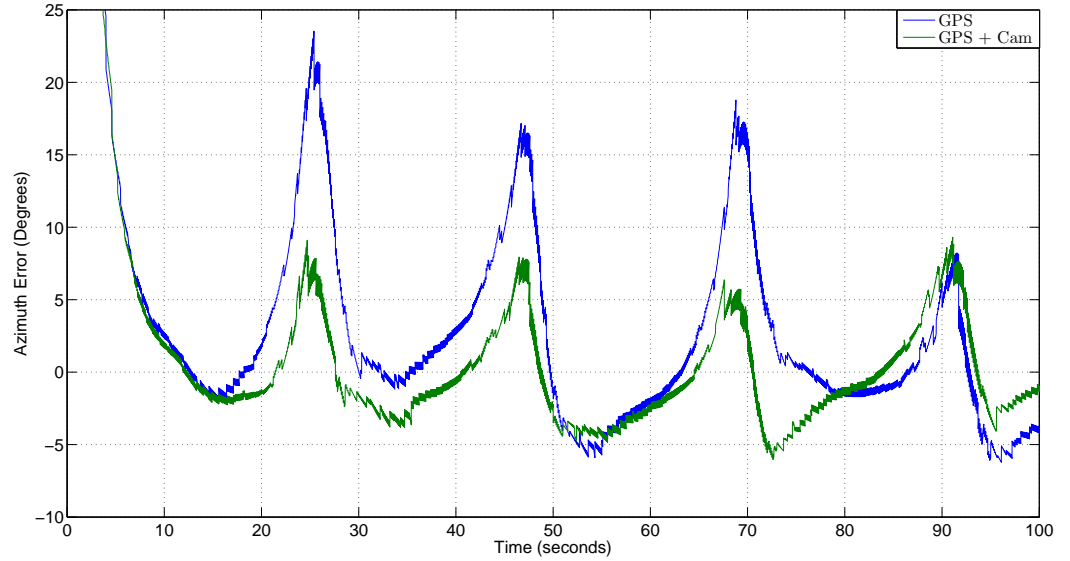


Fig. 4.4: Azimuth Error Comparison - GPS vs GPS coupled with Camera. The error in the azimuth angle calculated by the estimations are presented. It can be seen that coupling a GPS with a camera improves the estimation.

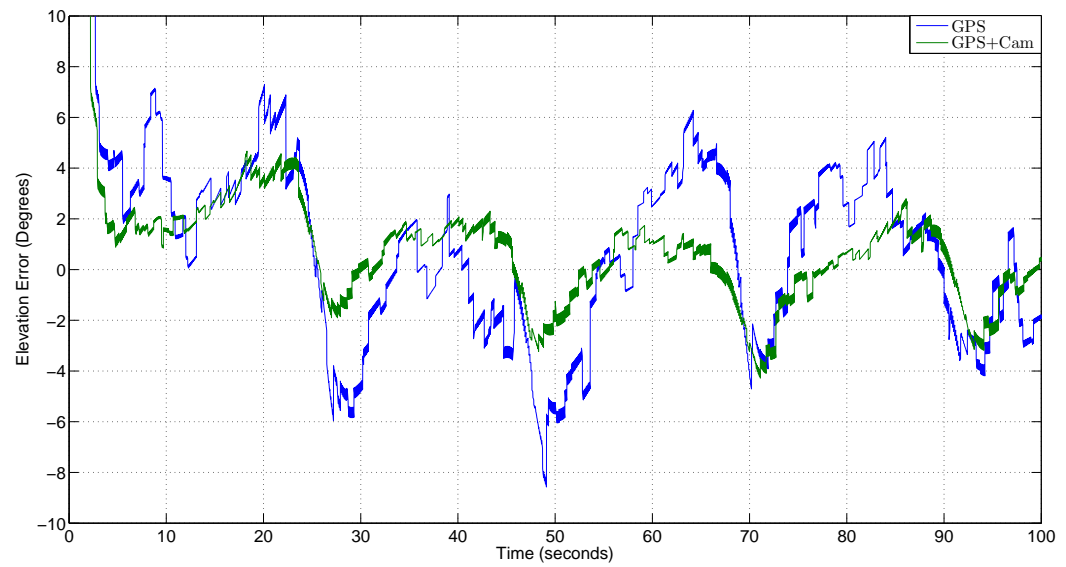


Fig. 4.5: Elevation Error Comparison - GPS vs GPS coupled with Camera. The error in the elevation angle calculated by the estimations are presented. It can be seen that coupling a GPS with a camera improves the estimation.

4.2 Comparison of Estimation Accuracy for EKF and DEKF with 0.75 s Random Delay

In this scenario the precision of the state estimations from an EKF and a DEKF are considered. Again, a GPS coupled with a camera serves as the inputs for the filters. Here we show how the estimations are impacted if the measurements have delays and a filter that does not take them into account is used. An improvement in the estimations is expected using the DEKF.

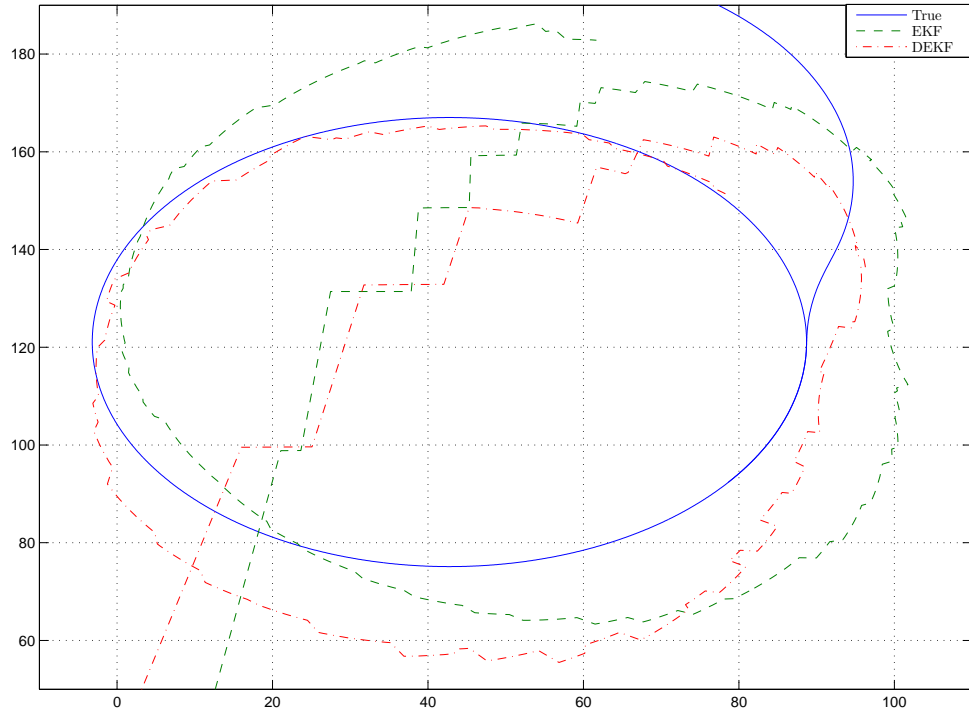


Fig. 4.6: UAV Trajectory Comparison - EKF vs DEKF. The trajectory of the aircraft is shown in this plot, against the estimated trajectory calculated by an EKF and by a DEKF.

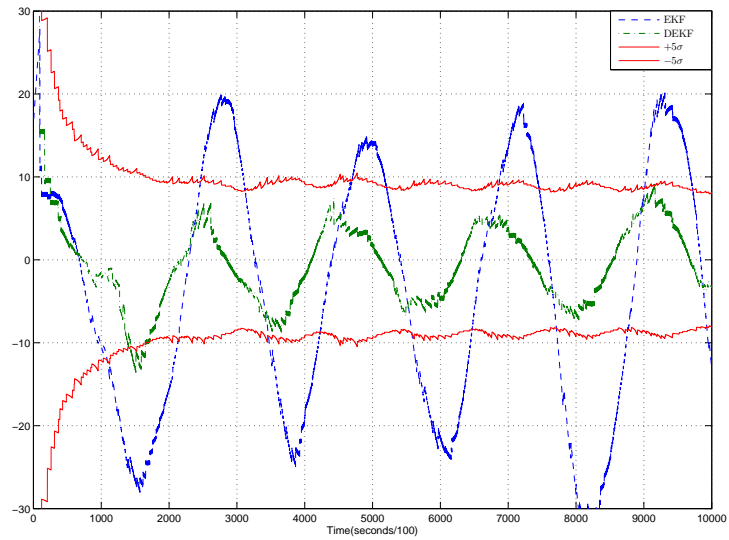


Fig. 4.7: North Position Error Comparison - EKF vs DEKF. The error in the north position is shown in this plot. The states estimations are compared for an EKF and a DEKF.

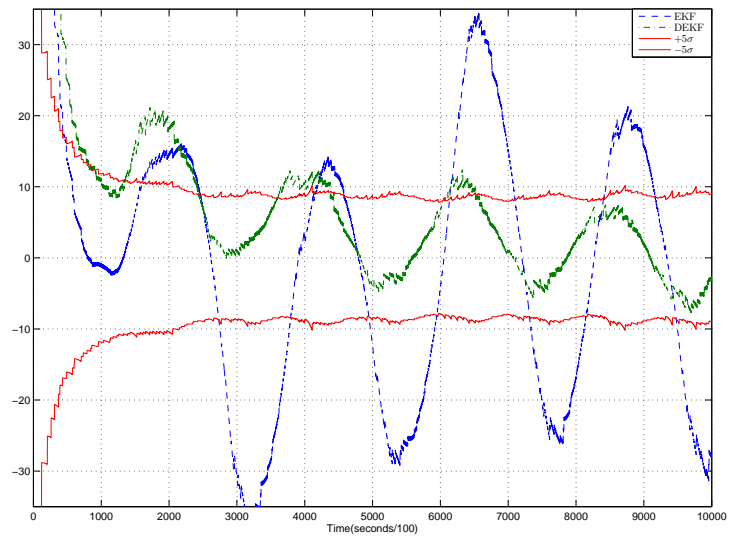


Fig. 4.8: East Position Error Comparison - EKF vs DEKF. The error in the east position is shown in this plot. The states estimations are compared for an EKF and a DEKF.

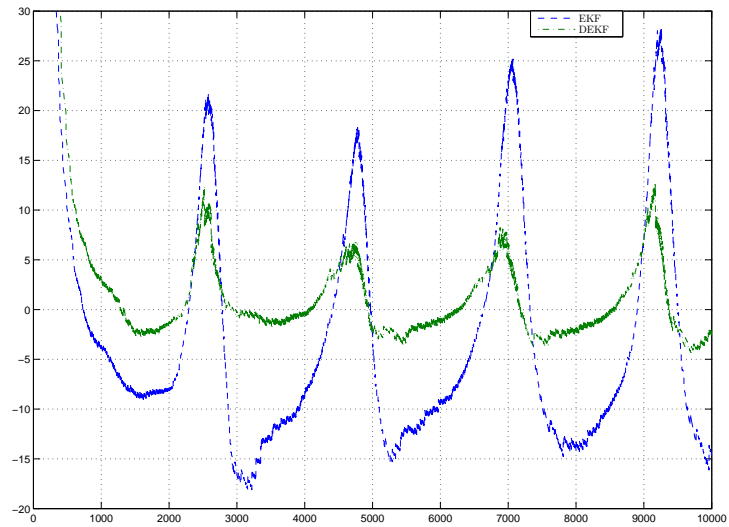


Fig. 4.9: Azimuth Error Comparison - EKF vs DEKF. The error in the azimuth angle calculated by the estimations is shown in this plot. Azimuth for the EKF and the DEKF are compared.

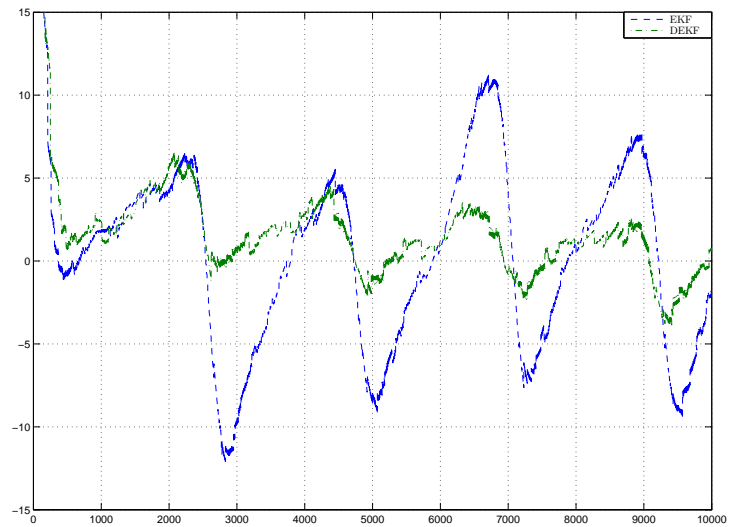


Fig. 4.10: Elevation Error Comparison - EKF vs DEKF. The error in the elevation angle calculated by the estimations is shown in this plot. Elevation for the EKF and the DEKF are compared.

4.3 Comparison of Estimation Accuracy Between No delay, 0.25 s, 0.50 s, and 0.75 s Random Delays

For this scenario we will use a GPS coupled with a camera. This scenario compares the estimation made by a DEKF for no delay, random delay of up to 0.25 s, 0.50 s, and 0.75 s. Since the delays are randomly generated, some of the measurements arrive in an out-of-sequence fashion. The Delayed Extended Kalman filter also compensates for this.

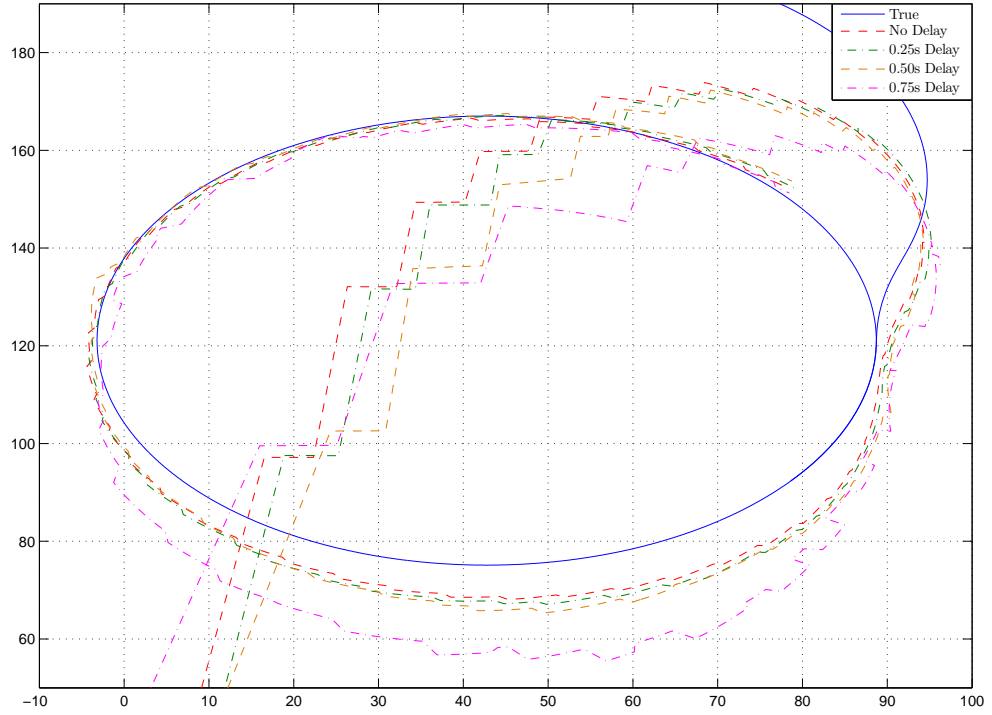


Fig. 4.11: UAV Trajectory Comparison - Different Delays. The trajectory of the aircraft is shown in this plot, against the estimated trajectory calculated by a DEKF for no delay, 0.25 s, 0.50 s, and 0.75 s of random delay.

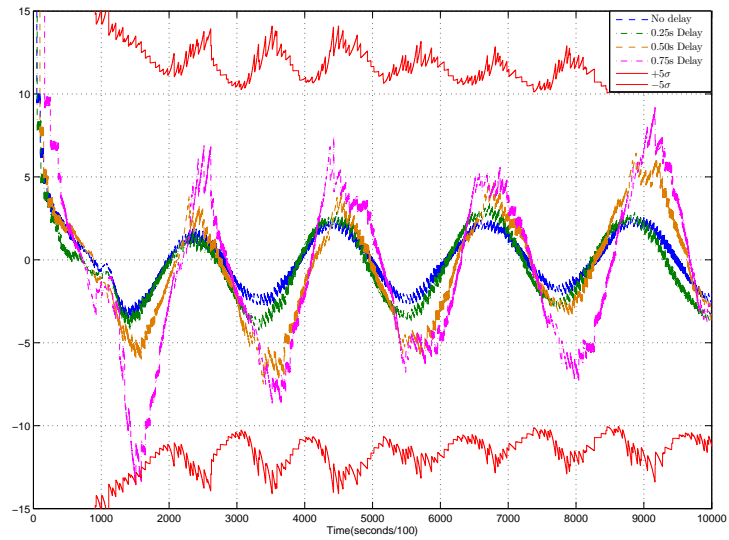


Fig. 4.12: North Position Error Comparison - Different Delays. The error in the north position is shown in this plot. The states estimations are compared for no delay, 0.25 s, 0.50 s, and 0.75 s of random delay.

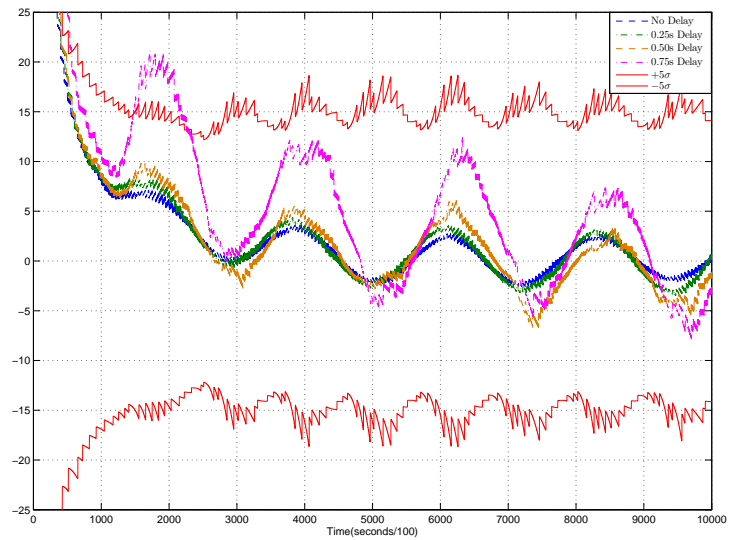


Fig. 4.13: East Position Error Comparison - Different Delays. The error in the east position is shown in this plot. The states estimations are compared for no delay, 0.25 s, 0.50 s, and 0.75 s of random delay.

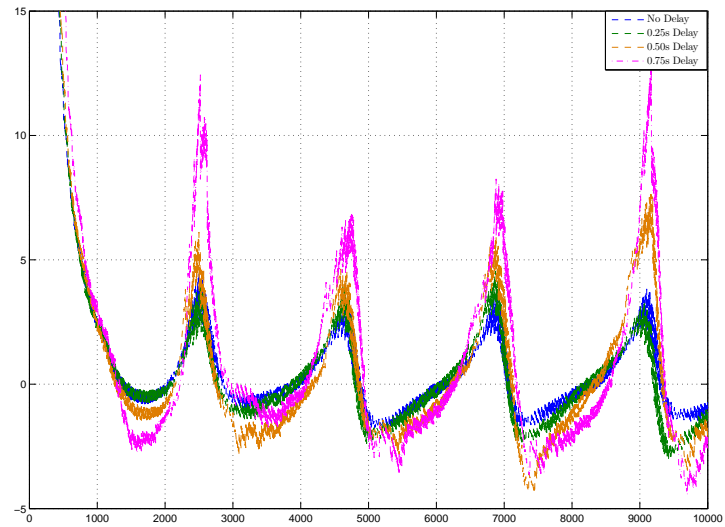


Fig. 4.14: Azimuth Error Comparison - Different Delays. The error in the azimuth angle calculated by the estimations is shown in this plot. Azimuth for no delay, 0.25 s, 0.50 s, and 0.75 s of random delay, are compared.

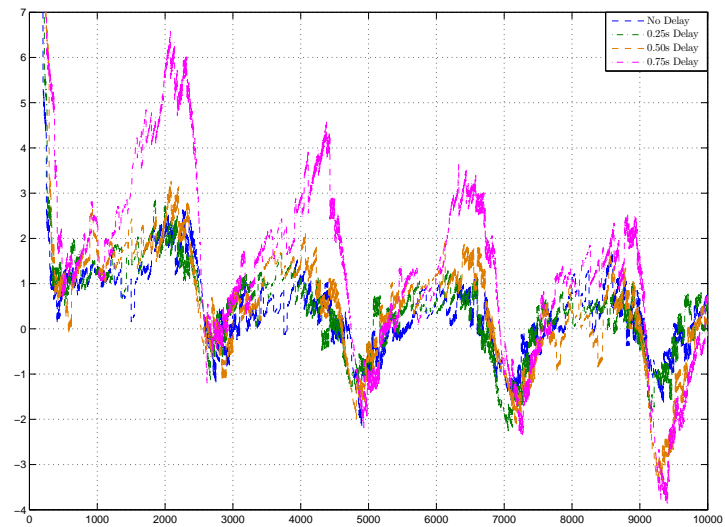


Fig. 4.15: Elevation Error Comparison - Different Delays. The error in the elevation angle calculated by the estimations is shown in this plot. Elevation for no delay, 0.25 s, 0.50 s, and 0.75 s of random delay, are compared.

CHAPTER 5

CONCLUSION

The purpose of this study is to develop an algorithm which could point an antenna and a camera towards a UAS. The reasons and motivations of this problem are primarily two. The first motivation was the Line-Of-Sight requirement of the Federal Aviation Administration which limits UAV missions to operate if and only if the aircraft is in the field of view of the operator. The existing method, is to keep an eye on the UAV with binoculars, which is prone to miss the plane frequently. Likewise, the other reason for the study was to improve the reliability of the mission by assuring the communication link with the aircraft, which can be achieved with a trustworthy tracking system.

The main hypotheses that we had were the following three.

1. The coupling of a camera with a GPS improves the state estimations of an aircraft.
2. Delayed and out of sequence measurements affect the state estimations in an inversely proportional way. A special Kalman filter needs to be used to address this scenario.

With the empirical findings we were able to conclude that:

1. If a camera is coupled with a GPS for state estimation in a Kalman filter, the errors will be less than those of using a GPS alone. This is more evident when the GPS measurements are delayed.
2. State estimations are adversely affected by an increase of delays in the measurements. Errors were significantly decreased when a special filter called Delayed Extended Kalman filter was used, instead of using the regular Extended Kalman filter.

One of the impacts of this study is the possibility of a modification of the Line-Of-Sight requirement of the FAA, allowing tracking systems to comply with this regulation as long as they are equipped with a camera. Therefore, freeing the operator of this task. Furthermore, tracking systems could implement these findings to improve their robustness, making more reliable the mission as a whole.

This research could be extended with the implementation of the antenna pointer system, to run more extensively tests. On the other hand, the same principle could be analyzed where the only sensors available were a camera and a signal strength receiver. Moreover, a scenario where communication or measurements are lost could be an interesting case of study.

This thesis has offered an approach to UAV tracking systems to improve the existing process, and was conducted in a simulated platform, MATLAB and Simulink, to have a strictly controlled environment. For simplifying purposes we made two main assumptions that need to be mentioned. One of the premises was that there was an image recognition algorithm which always recognized the aircraft as long as it were in the field-of-view of the camera. The second assumption was that the delays in the camera measurements were so small that they could be neglected without a big impact of the estimations.

As a support of what is often reasoned, an algorithm which consider the delays in the measurements and the OOSM problem, such as the Delayed Extended Kalman filter, reduces the estimates errors. Hence, a system which regards these conditions, can efficiently track a UAS increasing the reliability of the whole mission.

REFERENCES

- [1] R. Hilton, D. Martin, and W. Blair, "Tracking with time-delayed data in multisensor systems," 1993. [Online]. Available: <http://oai.dtic.mil/oai/oai?verb=getRecord{%&metadataPrefix=html{%&identifier=ADA271334}}>
- [2] E. Nettleton and H. Durrant-Whyte, "Delayed and asequent data in decentralized sensing networks," vol. 4571, no. 2001, pp. 1–9, 2001. [Online]. Available: http://spie.org/x648.html?product{%_id=444148http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=901100}}
- [3] T. Larsen and N. Andersen, "Incorporation of time delayed measurements in a discrete-time Kalman filter," vol. 4, no. December, pp. 3972–3977, 1998. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs{%_all.jsp?arnumber=761918}}
- [4] S. Julier and J. Uhlmann, "Fusion of time delayed measurements with uncertain time delays," *Proceedings of the 2005, American Control Conference, 2005.*, pp. 4028–4033, 2005.
- [5] M. Mallick, S. Coraluppi, and C. Carthel, "Advances in asynchronous and decentralized estimation," vol. 4, pp. 1873–1888, 2001. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs{%_all.jsp?arnumber=931505}}
- [6] Y. Bar-Shalom, M. Mallick, H. Chen, and R. Washburn, "One-step solution for the general out-of-sequence-measurement problem in tracking," *IEEE Aerospace Conference Proceedings*, vol. 4, pp. 1551–1559, 2002.
- [7] P. Lanzkron and Y. Bar-Shalom, "A two-step method for out-of-sequence measurements," *Aerospace Conference, 2004.*, pp. 2036–2041, 2004. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs{%_all.jsp?arnumber=1367986}}
- [8] K. Zhang and X. Li, "Optimal update with out-of-sequence measurements for distributed filtering," *Information Fusion, 2002.*, vol. 2, no. 1, pp. 1519–1526, 2002. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs{%_all.jsp?arnumber=1020997}}
- [9] Y. Bar-Shalom, "Update with out-of-sequence measurements in tracking: Exact solution," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 3, pp. 769–778, 2002.
- [10] M. Mallick, J. Krant, and Y. Bar-Shalom, "Multi-sensor multi-target tracking using out-of-sequence measurements," *Information Fusion, 2002.*, no. Fusion, pp. 135–142, 2002. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs{%_all.jsp?arnumber=1021142}}
- [11] S. Challa, R. J. Evans, X. Wang, and J. Legg, "A fixed-lag smoothing solution to out-of-sequence information fusion problems ," vol. 2, no. 4, pp. 325–348, 2002.

- [12] M. Orton and A. Marrs, "Particle filters for tracking with out-of-sequence measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 2, pp. 693–702, 2005.
- [13] G. L. Plett, D. Zarzhitsky, and D. J. Pack, "Out-of-order sigma-point Kalman filtering for target localization using cooperating unmanned aerial vehicles," *Lecture Notes in Control and Information Sciences*, vol. 369, pp. 21–43, 2007.
- [14] J. Kelly, S. Saripalli, and G. Sukhatme, "Combined visual and inertial navigation for an unmanned aerial vehicle," *Field and Service Robotics*, vol. 42, pp. 255–264, 2008. [Online]. Available: [http://link.springer.com/chapter/10.1007/978-3-540-75404-6_{ }](http://link.springer.com/chapter/10.1007/978-3-540-75404-6_{ }24)
- [15] S. Zhang and Y. Bar-Shalom, "Optimal Update with Multiple Out-of-Sequence Measurements with Arbitrary Arriving Order," *Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3116–3132, 2012. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_{ }all.jsp?arnumber=6324681
- [16] R. W. Beard and T. W. McLan, "State Estimation," in *Small Unmanned Aircraft: Theory and Practice*, 2010.
- [17] J. Zogg, "GPS: Essentials of satellite navigation," *Compendium*, p. 174, 2009. [Online]. Available: <http://scholar.google.com/scholar?hl=en{&}btnG=Search{&}q=intitle:GPS:+Essentials+of+Satellite+Navigation{#{ }0http://scholar.google.com/scholar?hl=en{&}btnG=Search{&}q=intitle:Gpsessentials+of+satellite+navigation{#{ }0>
- [18] B. W. Parkinson, J. J. Spilker, P. Axelrad, and P. Enge, *Global Positioning System: Theory & Applications, Volume I*, 1st ed. American Institute of Aeronautics and Astronautics, 1996.
- [19] —, *Global Positioning System: Theory & Applications, Volume II*. American Institute of Aeronautics and Astronautics, 1996.
- [20] E. D. Kaplan and C. Hegarty, *Understanding GPS: Principles and Applications*, 2nd ed. Artech House, 2005.
- [21] M. S. Grewal, L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, 2nd ed. Wiley-Interscience, 2007.
- [22] R. W. Beard and T. W. McLan, "Sensors for MAVs," in *Small Unmanned Aircraft: Theory and Practice*, 2010.
- [23] J. Rankin, "GPS and Differential GPS: An Error Model for Sensor Simulation," *Position, Location, and Navigation Symposium*, pp. 260–266, 1994.
- [24] R. Figliola and D. Beasley, *Theory and Design for Mechanical Measurements*. New York: John Wiley & Sons, Inc., 2006.
- [25] P. S. Maybeck, *Stochastic Models, Estimation and Control: Volume 1*, 1979.

- [26] R. Sharma and D. Pack, “Cooperative Sensor Resource Management for Multi Target Geolocalization using Small Fixed-wing Unmanned Aerial Vehicles,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*, 2013, pp. 1–11.