

UNIVERSITEIT • STELLENBOSCH • UNIVERSITY

Correlation and Tracking using Multiple Radar Sensors

by

Hendrik Barney de Villiers

*Thesis presented at the University of Stellenbosch in partial
fulfilment of the requirements for the degree of*

Master of Engineering Science

Department of Applied Mathematics
Stellenbosch University
Private Bag X1, 7602 Matieland, South Africa

Study leaders:

Prof. Ben Herbst Prof. Johan du Preez

December 2005

Copyright © 2005 University of Stellenbosch
All rights reserved.



Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and that I have not previously in its entirety or in part submitted it at any university for a degree.

Signature:

H.B. de Villiers

Date:



Abstract

Correlation and Tracking using Multiple Radar Sensors

H.B. de Villiers

Department of Applied Mathematics

Stellenbosch University

Private Bag X1, 7602 Matieland, South Africa

Thesis: MScEng (Applied Mathematics)

December 2005

Tracking manoeuvring military airborne targets with radar is problematic due to the low scan rates and the high levels of measurement noise. Surveillance systems using multiple radars have the benefit of an increased rate of observation and noise reduction but also have the problem of correlating observations from multiple sensors. Methods are discussed to correlate single observations from multiple radar sensors, as well as assigning observations to existing tracks. Filtering methods to reduce measurement noise of the target tracks and methods to extrapolate the predicted position of targets are also explored.

Uittreksel

Correlation and Tracking using Multiple Radar Sensors

H.B. de Villiers

Department of Applied Mathematics

Stellenbosch University

Private Bag X1, 7602 Matieland, South Africa

Tesis: MScEng (Applied Mathematics)

Desember 2005

Die teikenspoor op 'n radarstelsel van 'n maneuerende vegvliegtuig is problematies om te voorspel as gevolg van die lae waarnemingstempo en die hoë vlakke van waarnemingsruis. Lugwaarnemingstelsels wat veelvuldige radars gebruik het die voordeel van 'n verhoogde waarnemingstempo en geruisvermindering. Die probleem is egter hoe om hierdie veelvuldige radarobservasies te korreleer. Metodes word bespreek om enkele waarnemings van meervoudige radars met mekaar te korreleer en observasies met 'n bestaande teikenspoor te assosieer. Filtermetodes word ondersoek om die waarnemingsruis van die teikenspoor te verminder en om die verwagte posisies van die teikens met groter akuraatheid te ekstrapoleer.

Acknowledgements

My study leader, Prof Ben Herbst, for the guidance and ideas for this project. Thank you for always shedding light on the mathematical side of the problem.

Pieter-Jan Wolfaardt, for his time, for sharing his operational knowledge of the radar systems and having the perseverance to repeatedly dive into the statistical depths of the project.

Reutech Radar Systems, for the opportunity and the funding to pursue this project.

Marna, for positive criticism of my grammatical errors.

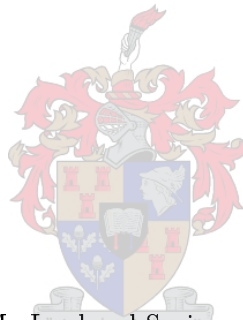
My parents, for the continuing support and love without which I would not have been able to reach the goals I've set for myself.

My flat mates, Tleng, Lara and Annaat who had to put up with my very random office hours and who cared enough to send smses sometimes, to know if I would be coming home that night.

MJ, for your friendship and wisdom and for keeping this impulsive man on the steady track of discipline and friendship in God. You are more than a champion.



Dedication

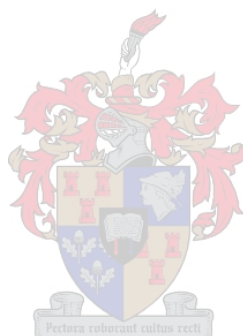


My Lord and Saviour,

Jesus Christ.

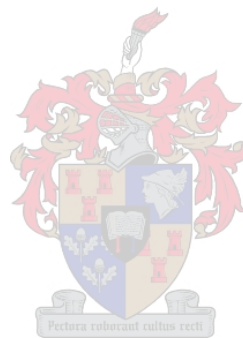
Contents

Declaration	ii
Abstract	iii
Uittreksel	iv
Acknowledgements	v
Dedication	vi
Contents	vii
List of Figures	x
List of Tables	xii
List of Abbreviations	xiii
List of Symbols	xiv
1 Introduction	1
1.1 Overview	1
1.2 Observation characteristics of radars	2
1.3 IFF information	5
1.4 Thesis Overview	6
2 Static Correlation	7
2.1 Preliminaries	7
2.2 Elevation estimation from multiple 2-D radars	8
2.2.1 Overview	8
2.3 The Range and Azimuth method	13
2.3.1 Overview	13
2.3.2 Window definition	13
2.3.3 Method	14
2.4 Gaussian Difference Vectors	16
2.4.1 Overview	16



2.4.2	Algorithm	17
2.4.3	Implementation Remarks	22
2.5	Comparison of Methods	23
2.6	Conclusions	28
3	Data Association (or Gating) for Radar Tracking	29
3.1	2-D vs. 3-D Modelling with only 2-D measurements	29
3.2	Range Azimuth Window	32
3.2.1	Overview	32
3.2.2	Algorithm	32
3.2.3	Illustration	33
3.3	Propagated Particle Covariance Estimation	35
3.3.1	Motivation	35
3.3.2	Position prediction of accelerating particles	36
3.3.3	Propagated Particles Covariance Correlation Algorithm	37
3.3.4	Illustration	39
3.4	Comparison of Methods	43
3.5	Conclusion	47
4	Target Tracking	48
4.1	The Track Manager	48
4.1.1	Correlation with existing tracks	49
4.1.2	Management of tracks	50
4.2	Target Models	51
4.2.1	Constant Velocity Target Model	51
4.2.2	Manoeuvring Target Model	52
4.2.3	Polar vs Cartesian coordinate system models	53
4.3	Filter Methods	56
4.3.1	Alpha-Beta ($\alpha\beta$) Filter	57
4.3.2	Kalman Filter	60
4.3.3	Interactive Manoeuvre Model (IMM) Filter	68
4.3.4	Performance evaluation of filters	70
4.4	Manoeuvre Detection	76
4.4.1	Overview	76
4.4.2	The second derivative of the filtered target position	77
4.4.3	The first derivative of the observed Doppler information	77
4.4.4	Comparison	78
4.5	Observation Covariance calculation	79
4.5.1	Overview	79
4.5.2	Algorithm	79
4.5.3	Illustration of observation noise	80
4.6	Conclusion	80

<i>Contents</i>	ix
5 Conclusions and Future Work	81
5.1 Conclusions	81
5.2 Future Work	82
Bibliography	83
A Radar sensor noise parameters	A-1



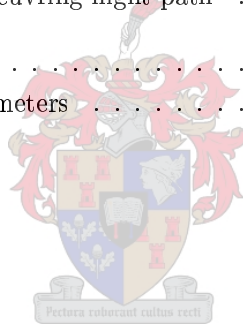
List of Figures

1.1	Radar tracking system architecture	3
1.2	Reflection of electromagnetic waves	3
1.3	Azimuth measurements	5
1.4	Combined range and azimuth uncertainty	5
2.1	Polar vs Cartesian coordinate systems	8
2.2	The effect of elevation on range measurement	8
2.3	Comparison error when ignoring target elevation	9
2.4	Method performance with no observation noise	11
2.5	Realistic method performance with target 10 km from sensor's centre .	12
2.6	Realistic method performance with target 40 km from sensor's centre .	13
2.7	Correlation of two radar observations	14
2.8	Illustration of range and azimuth-defined correlation windows	14
2.9	Zero Crossings problem	16
2.10	Realistic noisy observations of two radar sensors	17
2.11	Sensor noise covariances of radar sensors from different locations . . .	18
2.12	Randomly selected observation pairs	19
2.13	Estimated covariance from difference vectors	21
2.14	Scoring the original difference vector against the statistics	22
2.15	Sigma point placement in covariance	23
2.16	Results for Test Case 1	25
2.17	Results for Test Case 2	26
2.18	All test cases compared	27
2.19	Comparative results to highlight $R\Theta$ problem	27
3.1	Effect of target elevation on target track observations	30
3.2	Maximum elevation detection envelope for the Kameelperd radar system	31
3.3	Effect of target elevation on measured range	31
3.4	Correlation window definition A	33
3.5	Correlation window definition B	34
3.6	Correlation window operation with the $\alpha\beta$ filter	35
3.7	Geometric illustration of the particle manoeuvre formulation	37
3.8	Effect of manoeuvre/banking acceleration on particles	40

3.9	Effect of heading on particles	40
3.10	Effect of velocity on particles	41
3.11	Effect of process position covariance on particles (1)	41
3.12	Effect of process position covariance on particles (2)	42
3.13	Effect of cumulative process and observation covariances on particles	42
3.14	Particle positions after taking all unknowns into consideration	43
3.15	Correlation test for a straight path	44
3.16	Window size results for a straight path	44
3.17	Correlation test for a 4 G Manoeuvring path	45
3.18	Window size results for a 4 G Manoeuvring path	46
3.19	Correlation test for path with straight and manoeuvring sections	46
3.20	Window size results for a straight and manoeuvring paths	47
4.1	Track Management Flow Diagram	49
4.2	Correlation windows for track updating	50
4.3	Correlation window size increase	51
4.4	Velocity projection to the Range plane	55
4.5	Velocity projection perpendicular to the target heading	55
4.6	Velocity projection directly in the range plane	56
4.7	Effects of $\alpha\beta$ parameters on filter performance	58
4.8	Correlation window operation with the $\alpha\beta$ filter	59
4.9	Parameter adjustment methods with the $\alpha\beta$ filter	60
4.10	Z flight test simulation setup	64
4.11	Process Covariance Adjustment	65
4.12	Effect of initial process covariance selection	67
4.13	Effect of process covariance noise selection	68
4.14	IMM filter construction	69
4.15	Illustration of IMM performance	69
4.16	Z flight test simulation setup	73
4.17	Z flight simulation close-up at the first manoeuvre	73
4.18	Straight flight simulation setup	74
4.19	Manoeuvre simulation setup	75
4.20	Results for manoeuvre detection	78
4.21	Illustration of typical radar observation noise	80

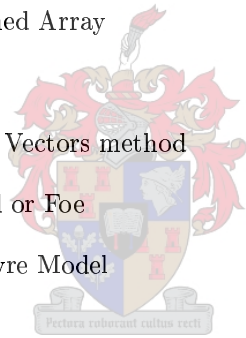
List of Tables

2.1	Correlation results from the various test locations	26
4.1	Parameter setup for $\alpha\beta$ filters	58
4.2	$\alpha\beta$ Filter and correlation window parameters	59
4.3	Filters setup	70
4.4	Filter results from Z flight simulation	74
4.5	Filter results for a straight flight path	75
4.6	Filter results for a manoeuvring flight path	76
A.1	Page sensor parameters	A-1
A.2	Kameelperd sensor parameters	A-1



List of Abbreviations

2-D	Two-Dimensional
3-D	Three-Dimensional
ATC	Air Traffic Control
CM	Correlation Miss
EKF	Extended Kalman Filter
ESA	Electronically Scanned Array
FC	False Correlation
GDV	Gaussian Difference Vectors method
IFF	Identification Friend or Foe
IMM	Interactive Manoeuvre Model
m	Metres
KF	Kalman Filter
km	Kilometres
$R\theta$	Range Azimuth method
RADAR	Radio Detection And Ranging
RF	Radio Frequency
RMS	Root Mean Square
RRS	Reutech Radar Systems
PPI	Plan Position Indicator



List of Symbols

Vectors and Matrices:

I	Identity matrix
C	Covariance matrix
C^t	Transposed matrix C
C^{-1}	Inverse of matrix C
R	Observed Target Range
\hat{R}	Estimated Target Range
\tilde{R}	Target Range with added expected observation noise
Θ	Observed Target Azimuth
$\hat{\Theta}$	Estimated Target Azimuth
$\tilde{\Theta}$	Target Azimuth with added expected observation noise
ϕ	Observed Target Elevation Angle
$\hat{\phi}$	Estimated Target Elevation Angle
$\tilde{\phi}$	Target Elevation Angle with added expected observation noise
x	Observed Target position on the x axis
\hat{x}	Estimated Target position on the x axis
\tilde{x}	Target position on the x axis with added expected observation noise
σ_R	Range noise variance
σ_Θ	Azimuth noise variance
σ_ϕ	Elevation noise variance

Chapter 1

Introduction

1.1 Overview

The detection of aircraft with radar can be obstructed by the limitations of the radar itself, the characteristics of the aircraft, clutter from atmospheric conditions or electronic countermeasures, to mention only a few. Having multiple radars in the field will increase the probability of detection and accuracy, but the integration of multiple sensors is not trivial. The work for this thesis was funded by Reutech Radar Systems (RRS) in order to answer core questions of multiple sensor integration.

Suppose there are two radars and each radar registers/observes a target, how can one determine whether both register the same target? This is the central question of this thesis. Although the answer appears to be straightforward: if the observations appear at the same position at the same time, surely they are of the same target—it turns out that it is not that simple.

What can go wrong? Firstly, the characteristics of any radar system has inherent uncertainties that are described in Section 1.2. Secondly, a radar with a detection envelope that should have observed the target, might not detect it due to obstructions like mountainous terrain. Thirdly, other aircraft in close proximity might easily be confused with the target.

Static correlation refers to the process of correlating single observations from two radars. It is assumed that the observation of both radars occurred at exactly the same instant. Ideally the correlation window should be as small as possible to minimise false correlations with other aircraft but also large enough to account for observation noise. The method currently used by Reutech Radar Systems is simply to define a window around an observation and see if there are any other observations within this window. Although the attempt is made to take the noise characteristics of both radars into account for in the window definition, it does not take the combined uncertainty of both radars into account. A new method was developed to deal with this combined uncertainty and compared with the existing method.

The next step was to take the dynamics into account. Dynamic correlation refers

to the process of correlating observations to an existing target track. Given the dynamics of the target, the position of the target can be predicted via extrapolation at the time of the observation. Now the question becomes: Is this new observation a valid update for the existing target track? Again the problems described earlier like aircraft in close formation, false detection from clutter or lack of detection require the definition of the correlation windows to be as small as possible but also large enough to account for inaccuracies in the observation. The method used by Reutech Radar Systems is much like the static correlation version, but the size of the window depends on the tracking filter parameters. However, the method must be fine-tuned manually for the expected aircraft velocities and radar characteristics. A new method was developed to automate the correlation window definition based on the derived target dynamics from the tracking filter. This resulted in a much smaller and accurate representation of the target track uncertainty in the correlation window.

As a dynamic correlation window relies heavily on the accuracy of the derived tracking information, a closer look was given at the filter algorithms. Reutech Radar Systems uses an alternating parameter, Alpha-Beta filter in their systems. Kalman filters were compared to the Alpha-Beta filters. Although Alpha-Beta filters are extremely simple and cost-effective, Kalman filters are much more accurate due to the estimation of the target state and the target process uncertainty. Another advantage of using Kalman filters is that the process uncertainty can be incorporated into the dynamic correlation methods.

The radar system with the correlation and tracking interaction is illustrated in Figure 1.1.



1.2 Observation characteristics of radars

The term radar is an acronym for the term **R**adio **D**etection and **R**anging. Electromagnetic waves transmitted from the source are scattered by changes of atomic density. Therefore objects constructed from metals or carbon fibre, like aircraft and ships, will scatter the waves. The small portion of waves scattered back to the source (the reflected waves) are used to detect aircraft, as illustrated in Figure 1.2.

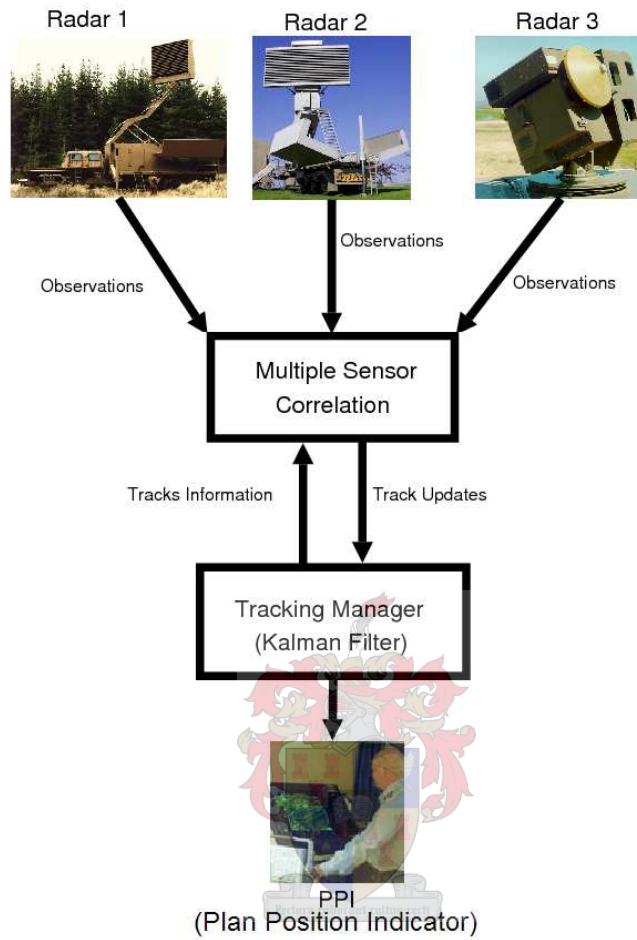


Figure 1.1: Radar tracking system architecture

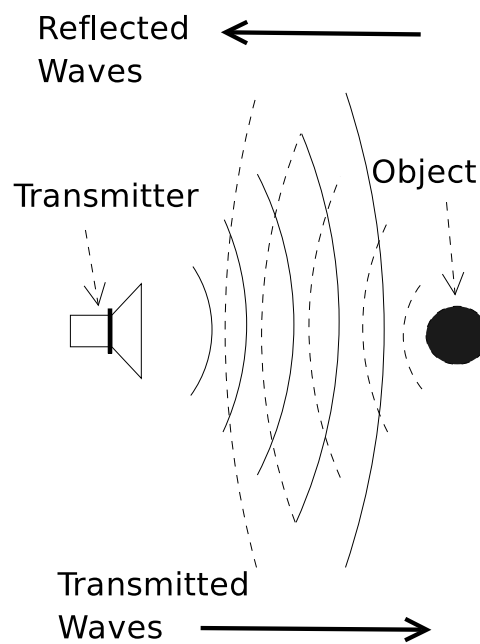


Figure 1.2: Reflection of electromagnetic waves

The distance (or range) of an object that reflects the signal is measured as the delay between the transmission of the signal and the reception of its echo (or reflection) times the speed of the wave,

$$r = \frac{c\lambda}{2}.$$

c is the speed of light and λ is the travel time of the reflected wave. The intensity of the reflection is very low, $\frac{1}{r^4}$ where r is the range, thus the detected signal will be very noisy. Rather than looking for the reflection directly in the received signal, the reflection is found in the cross-correlation between a copy of the original transmitted signal and the received signal. This cross-correlation will be large at the lag time associated with the time duration of the signal to the reflecting object and back. If there is no reflection, the uncorrelated nature of the noise in the signal should make the cross-correlation small. A threshold is used to separate detections.

There are mainly two types of radars: Tracking radars or Search radars. Tracking radars lock onto one target and attempt to keep the lock on the target by moving the antenna to face the target at all times. This provides very accurate position measurements at very high update rates.

The search radar has an antenna that is rotated continuously to observe multiple targets in its detection envelope. Reflections from specific directions or sectors can be detected by using a directional antenna for the transmission and reception of the waves. By rotating this antenna through 360 degrees, a range and azimuth measurement can be associated with every detected object, as illustrated in Figure 1.3. The azimuth measurement is simply the angle of the antenna, relative to north, during rotation at the instant of object detection. Due to the rotation and width of the transmitted beam, the azimuth measurement, relative to the range measurement, is noisy and escalates as the range of the target increases. The rotation of the transmitter also implies a scan rate of the sensor - the antenna only updates the target positions in the detection envelope after every rotation. Search radar antenna rotations typically take between 4 and 10 seconds to complete. In aeronautical terms, this is a long time as aircraft flying at an average velocity of Mach 0.7 or 250m/s will travel about 1.5 km in a scan update interval of six seconds.

By observing the frequency shift between the transmitted and received signal, the rate at which the range is changing can be calculated. This is more commonly known as the range-rate or Doppler measurement. Tracking and Search radar systems can be quipped with this technology, in which case the range-rate is also available.

Typically, search radar systems are 2-D sensors. Therefore they only measure the range and azimuth of the target relative to the radar. The elevation angle measurement can be obtained by means of either fixed multiple beams in elevation or scanned pencil beams. The cost and limited benefit of this feature makes the simpler 2-D radar systems much more popular than 3-D systems in military environments with limited resources. This report assumes 2-D radars are used, but the extra

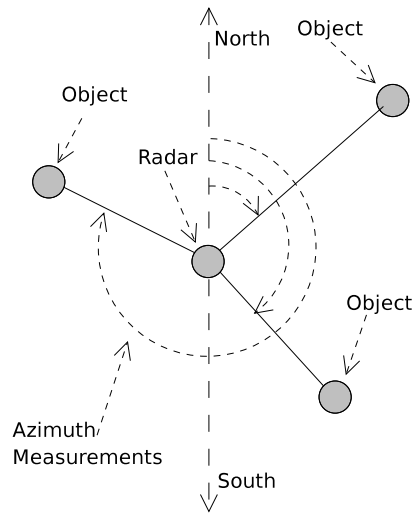


Figure 1.3: Azimuth measurements

elevation information from a 3-D radar can effortlessly be included in the correlation and tracking methods.

Taking into account the combined range and azimuth uncertainties, Figure 1.4 illustrates the positional uncertainty of detected targets. Note that the azimuth measurement is responsible for most of the uncertainty.

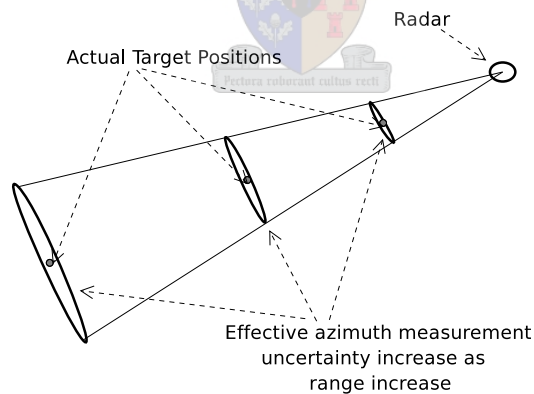


Figure 1.4: Combined range and azimuth uncertainty

1.3 IFF information

Identification Friend or Foe (IFF) is used in the military environment to identify targets on the Plan Position Indicator (PPI) or radar screen as friendly or not. The interrogation occurs by sending an encrypted message to the suspect aircraft

and requesting a pass-code answer based on the sent message. If the answer is correct, the aircraft is designated friendly on the PPI. These interrogations must be kept to a minimum as enemy code breakers will attempt to extract the pass-codes from the transmissions. Therefore, the operator or the automatic tracking programme must keep a record of aircraft positions to minimise interrogations. This becomes especially difficult as friendly and enemy aircraft may fly very close at times and perform high G manoeuvres during aerial combat. An understanding of the radar observation uncertainty whilst designing correlation and tracking methods is therefore imperative.

1.4 Thesis Overview

Chapter 2 discusses two static correlation methods. The method used by Reutech Radar Systems and a new method that attempts to improve on it are illustrated and compared.

Accurate data association is crucial in the military environment, as explained in Section 1.3. A simple distance measurement between the predicted target position and the observation with a threshold is not advisable. All methods define some type of correlation window around the predicted position. To repeat: the method currently used by Reutech Radar Systems and another method with different window definitions are explored and compared in Chapter 3.

The above mentioned correlation methods require the speed and the heading estimates. A target tracking algorithm is required to obtain these estimates of the target. In Chapter 4 the well-known $\alpha\beta$ -, Kalman- and IMM filters are implemented and compared, to provide these estimates.

Chapter 2

Static Correlation

Situations arise when airborne targets are within the detection envelopes of multiple radar sensors. Detections of one target from more than one sensor is very beneficial for the position tracking of the target, but it is necessary to first establish that it is the same target. This chapter discusses possible methods for correlating observations from two radar sensors.

2.1 Preliminaries

Updates on the position of the target is not possible at any time instant due to the rotating nature of the search radar antenna. The target updates are available only once during each rotation. The instants at which the sensors observe the target will almost never coincide - thus it is not feasible to rely purely on the observed positions from search radars to correlate the observations. Information on the dynamics of the target from the tracking system is required to extrapolate the measured position of the target from both radars to obtain estimates at the same instant. This will be covered in the next chapter. The methods discussed in this chapter assume that the observations from the two sensors are gathered at exactly the same time.

The following methods discussed in this chapter require translations between the polar and Cartesian coordinate systems. The translation is simple and reversible as illustrated in the following equations,

$$\begin{bmatrix} R \\ \Theta \end{bmatrix} = \begin{bmatrix} \sqrt{x^2 + y^2} \\ \arctan\left(\frac{y}{x}\right) \end{bmatrix}$$
$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} R \cos(\Theta) \\ R \sin(\Theta) \end{bmatrix}.$$

Figure 2.1 concisely illustrates the differences between the two coordinate systems.

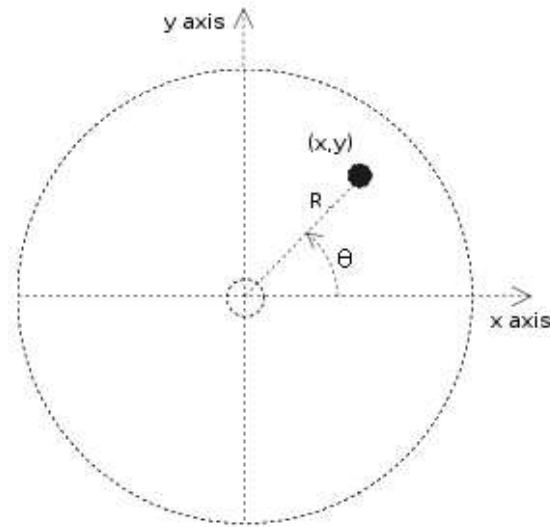


Figure 2.1: Polar vs Cartesian coordinate systems

2.2 Elevation estimation from multiple 2-D radars

2.2.1 Overview

It is especially necessary for static correlation to estimate the elevation and actual ground range of the target to the radar site as given by the observed range and azimuth. Figure 2.2 illustrates the effect of elevation on the observed range.

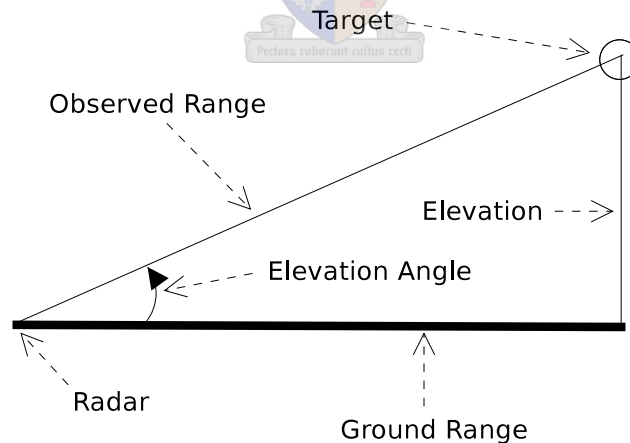


Figure 2.2: The effect of elevation on range measurement

This effect is very important to take into account when comparing observations from two radars. If two radars are observing the same target and target elevation is ignored, the distance between the two observations will increase as the target comes

closer to the radars. This effect is illustrated in figure 2.3 where a target with an elevation of 4000 meters is observed by two Kameelperd radars.

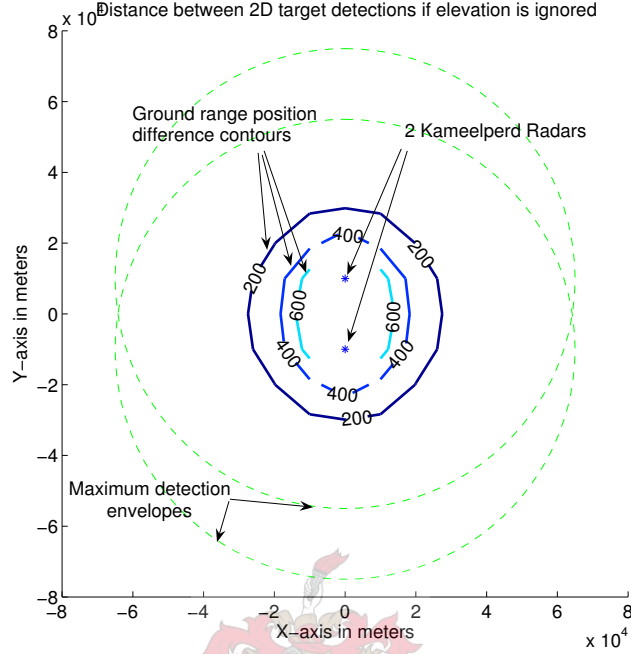


Figure 2.3: Comparison when ignoring target elevation

The elevation is estimated using an iterative procedure. The elevation runs through all possible values, 0 to h_{max} . h_{max} is the maximum elevation observable with the radar. The projected ground range values for radar are computed for each elevation value. Theoretically the difference should be zero for the actual elevation, i.e. the minimum distance gives the best elevation estimate. The algorithm proceeds as follows, given R_1 , Θ_1 (observed from radar 1), R_2 , Θ_2 (observed from radar 2) and \underline{s} is the position of radar 2 in terms of radar 1.

Stepwise algorithm

1. For $h = 0$ to h_{max} in steps of Δh
2. Calculate the ground range coordinates $C_{k,1,(x,y)}$ and $C_{k,2,(x,y)}$ from the range and azimuth observation of both radars assuming the target elevation is equal to the test elevation h_k , given the coordinates $s_{(x,y)}$ of the second radar in terms of the first,

$$C_{k,1,z} = C_{k,2,z} = h_k$$

$$C_{k,1,x} = \sqrt{R_1^2 - h_k^2} \cos(\Theta_1)$$

$$C_{k,1,y} = \sqrt{R_1^2 - h_k^2} \sin(\Theta_1)$$

$$C_{k,2,x} = \sqrt{R_2^2 - (h_k - s_x)^2} \cos(\Theta_2)$$

$$C_{k,2,y} = \sqrt{R_2^2 - (h_k - s_y)^2} \sin(\Theta_2).$$

3. Calculate the distance e_k between the ground plane Cartesian coordinates from the two radar sensors (indices 1 and 2) for this assumed elevation,

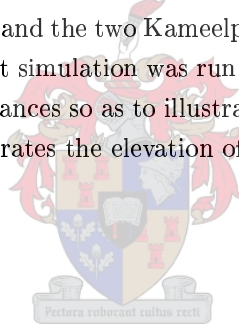
$$e_k = |\underline{C}_{k,1} - \underline{C}_{k,2}|.$$

4. Increment the test elevation h_k with the search interval Δh and repeat steps 2 to 4 until h_{max} is reached.
5. The elevation estimate is the value of h where e is at a minimum,

$$\text{elevation estimate} = h_k \text{ where } e_k = \min(e).$$

Results of estimation

For testing purposes, a target was first simulated to be 10 km from the sensor array centre at an elevation of 5000 m and the two Kameelperd radar sensors were located 10 km from each other. The first simulation was run without any real noise, to test the system under ideal circumstances so as to illustrate the method. The minimum point in Figure 2.4 clearly illustrates the elevation of the target.



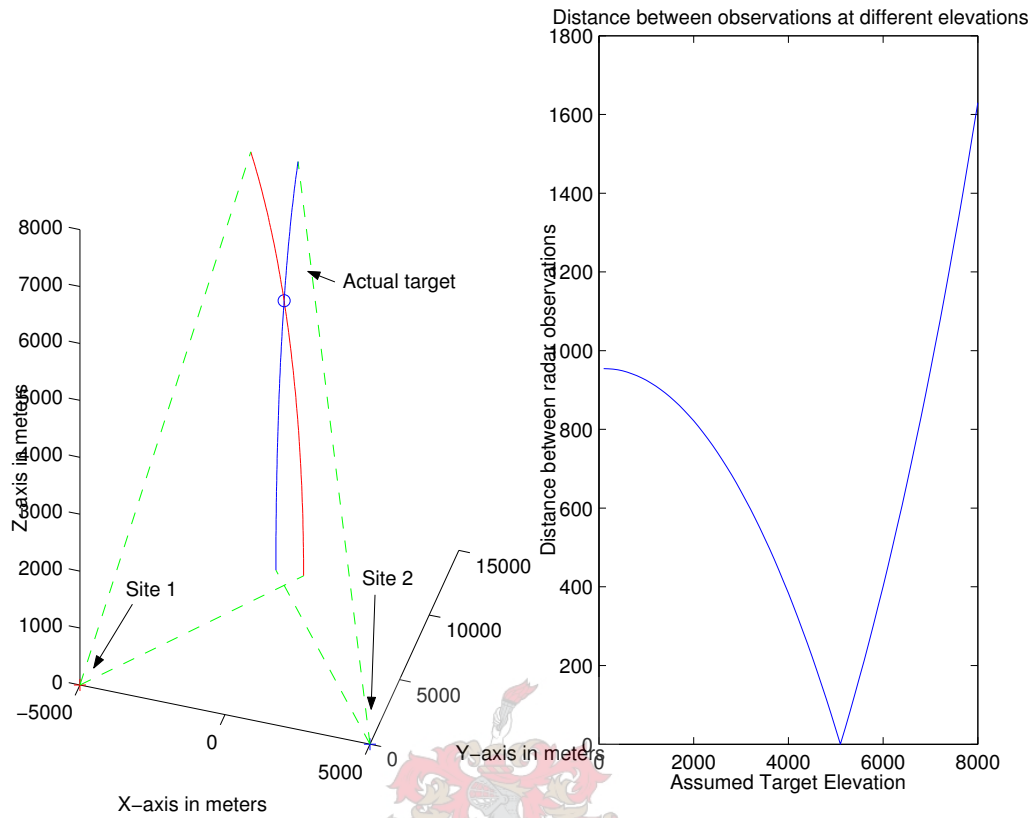


Figure 2.4: Method performance with no observation noise

Figure 2.5 illustrates the performance of the method with the expected observation noise of two Kameelperd radar sensors. The best elevation estimation differs from the actual target elevation by 500 m. The introduction of noise completely changes the situation. In this case there is a well-defined minimum but it is at the wrong elevation.

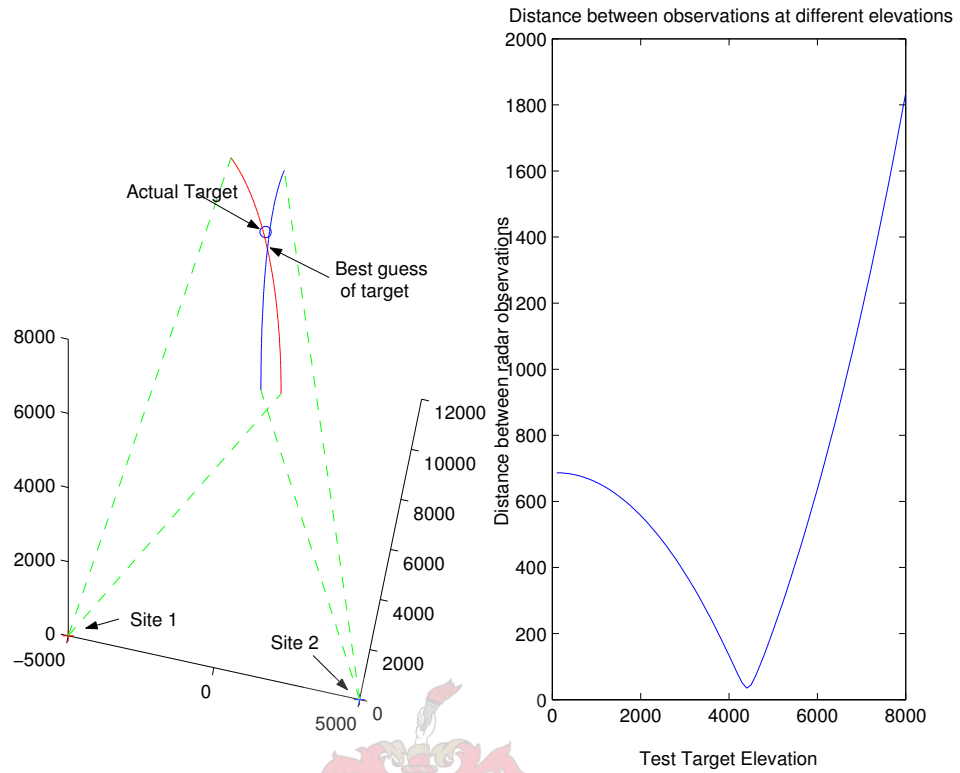


Figure 2.5: Realistic method performance with target 10 km from sensor's centre

Figure 2.6 illustrates how the method completely fails to estimate the target elevation when the target is located 40 km, with a very noisy observation, from the sensor array centre with the expected observation noise from a Kameelperd radar.

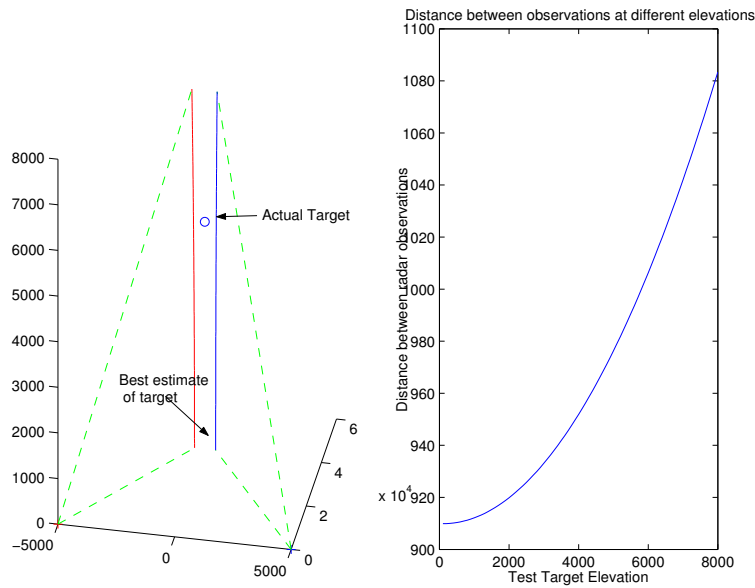


Figure 2.6: Realistic method performance with target 40 km from sensor's centre

2.3 The Range and Azimuth method

2.3.1 Overview

This method attempts to correlate the observations from multiple sensors using sensor noise characteristics without taking the noise characteristics of the other sensor into account. This is accomplished by defining a polar rectangular window around the expected position of the target that is defined in range and azimuth from the sensor centre. Implementations of this method, currently used by RRS, store a number of predefined window sizes and an operator selects one based on the application requirements.

2.3.2 Window definition

The radar measures in the polar coordinate system. Therefore the measurement uncertainties will also be defined in the polar coordinate system and so it will be only natural to define the windows in the polar coordinate system. Unfortunately there is no direct way to compare the range and azimuth measurements of two radars at separate locations. Therefore the observations have to be translated, first to a global coordinate system and then to the polar coordinate system of the radar in question, for comparison, as illustrated in Figure 2.7.

The size of the window is defined in range and azimuth terminology to account for the increasing noise as the distance from the radar to the target increases. Figure 2.8 illustrates two correlation windows with the same azimuth and range size parameters, but at different distances from the radar.

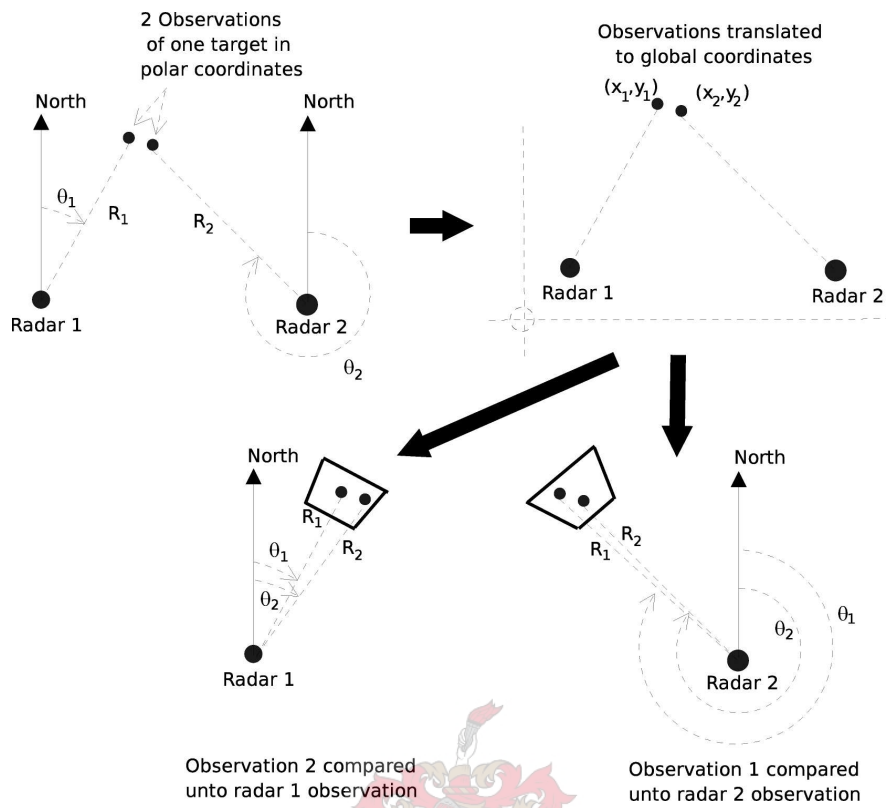


Figure 2.7: Correlation of two radar observations

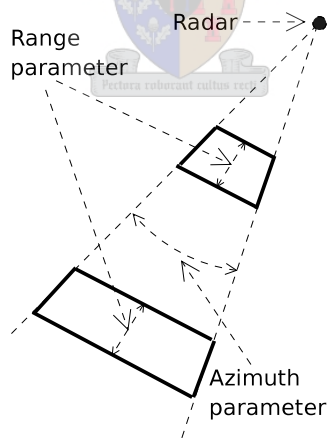


Figure 2.8: Illustration of range and azimuth-defined correlation windows

2.3.3 Method

This method is described visually in Figure 2.7. Mathematically the translations and comparisons are as follows:

1. Translate both observations into the global coordinate system as follows,

where (x, y) are the global coordinates, range R , azimuth Θ and the radar position global coordinates $C_{1,(x,y)}$,

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} R_1 \cos(\Theta_1) + C_{1,x} \\ R_1 \sin(\Theta_1) + C_{1,y} \end{bmatrix}$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} R_2 \cos(\Theta_2) + C_{2,x} \\ R_2 \sin(\Theta_2) + C_{2,y} \end{bmatrix}.$$

2. Translate both observations into the polar coordinate system, using the first radar as origin,

$$\begin{bmatrix} R_1 \\ \Theta_1 \end{bmatrix} = \begin{bmatrix} \sqrt{(x_1 - C_{1,x})^2 + (y_1 - C_{1,y})^2} \\ \arctan\left(\frac{y_1 - C_{1,y}}{x_1 - C_{1,x}}\right) \end{bmatrix}$$

$$\begin{bmatrix} R_2 \\ \Theta_2 \end{bmatrix} = \begin{bmatrix} \sqrt{(x_2 - C_{1,x})^2 + (y_2 - C_{1,y})^2} \\ \arctan\left(\frac{y_2 - C_{1,y}}{x_2 - C_{1,x}}\right) \end{bmatrix}.$$

3. Compare the two observations, using the window size parameters as thresholds,

$$[|\Theta_1 - \Theta_2| < \Theta_t] \text{ and } [|R_1 - R_2| < R_t].$$

4. Repeat steps 2 and 3, using the second radar as origin.
5. If the comparison finds the observations within the windows of either the first or the second radar, the observations correlate.

The algorithm is very simple and easy to implement. There is only one trivial issue - the Zero Crossings problem. Zero Crossings occur when comparing two azimuth observations that lie on the west and east side of the north line. As illustrated in Figure 2.9, it causes a radial difference of $(2\pi - \Delta\Theta)$ instead of the actual radial difference. This is easily corrected but requires some attention.

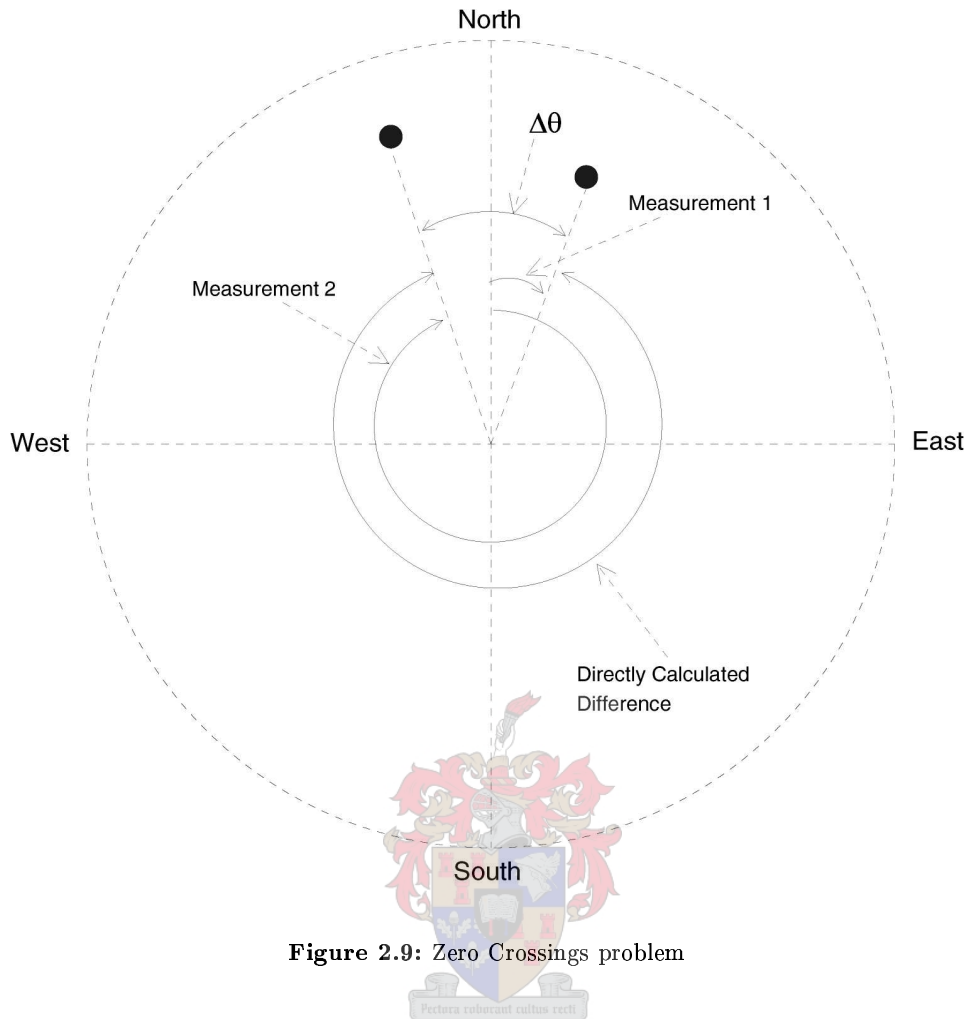


Figure 2.9: Zero Crossings problem

2.4 Gaussian Difference Vectors

2.4.1 Overview

The method previously discussed in Section 2.3 correlates an observation from the first radar with the observation from the second radar by defining a correlation window, based on the characteristics of the second radar. This process is then repeated by switching the roles of the sensors. If either attempt correlates, the observations correlate. The illustration in figure 2.10 shows how the observations from two radar sensors will differ if they are observing the same target. The simulated figure shows a hundred random detections by each sensor given the covariance of the sensor noise. This clearly shows that, if the observation azimuth deviations are large and the correlation windows are too small, the correlation might fail, even though the same target was detected. Increasing the rectangular window in size to accommodate the need for a larger azimuth correlation will be a mistake: it will result in a window that will be too large in the range axis.

The *Gaussian Difference Vectors* (GDV) method is an attempt to compute an

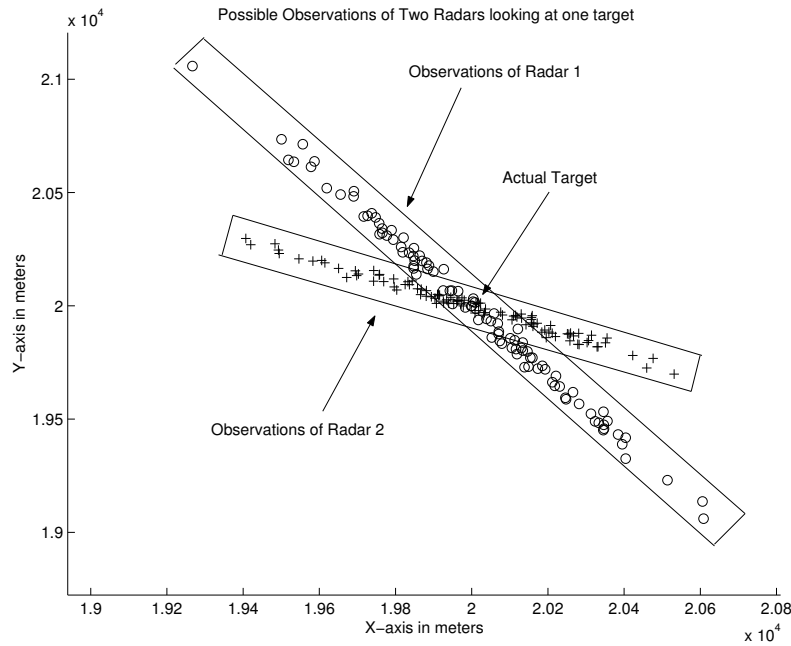


Figure 2.10: Realistic noisy observations of two radar sensors

optimal correlation window using the combined statistics from two independent observing radar sensors. Traditionally a rectangular correlation window is obtained on the basis of the characteristics of the one radar. The position of the second observation relative to the window determines whether a correlation is obtained. However, the GDV method attempts to take into account the Gaussian uncertainty characteristics of *both* radar sites in the decision-making process.

Note that two observations that have independent noise need to be correlated. In other words, this method attempts to match two variables which are both inaccurate and have different noise covariances. Therefore the approach to this problem has to take into account the uncertainty characteristics of both radar sites, not only independently but also cumulatively.

The method basically has three steps. Firstly, it estimates the covariance matrix of how the observations from the two radars will relate to each other at a particular location. Secondly, it calculates the confidence that the two observations do correlate, given the covariance matrix. Thirdly, a threshold is used to determine whether the confidence is high enough for correlation.

2.4.2 Algorithm

Determine the measurement uncertainty of the sensors at the location

In the latter part of the method, it would be advantageous to work in the Cartesian coordinate system, but the uncertainty of the radar sensors is defined in the polar coordinate system. The method use the polar coordinate system to compute the

observations with Gaussian noise, but then use the Cartesian coordinate system for the rest.

Bear in mind that the elevation of the target cannot be measured by 2-D radars, which affects the observed range. Section 2.2 explains a technique for estimating this elevation, but this may not be known due to measurement inaccuracies. In addition, this elevation calculation assumes that the same target is observed but this is exactly what needs to be established. Using this bootstrap argument of assuming that the two observations are of the same target, the elevation can be estimated for more accurate ground range locations of the observations. After the elevation is estimated, the means of the two elevation corrected observations are used as the location where the uncertainty distributions of the radar observations are computed.

Generation of sample observation pairs

Given the uncertainties of the radars, an arbitrary number of observation points are calculated from a Gaussian distribution for both radar sensors, as illustrated in Figures 2.11 and 2.12. Pairs of the observations from the two sources are created and elevation estimation and correction are performed on these pairs. K difference vectors between sample k of the one radar with sample k of the other radar are calculated. The radar indexes are labelled 1 and 2.

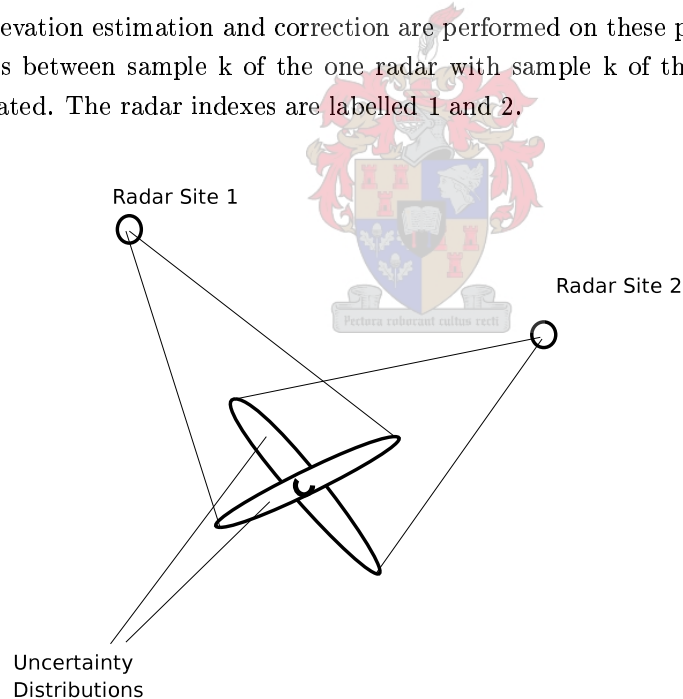


Figure 2.11: Sensor noise covariances of radar sensors from different locations

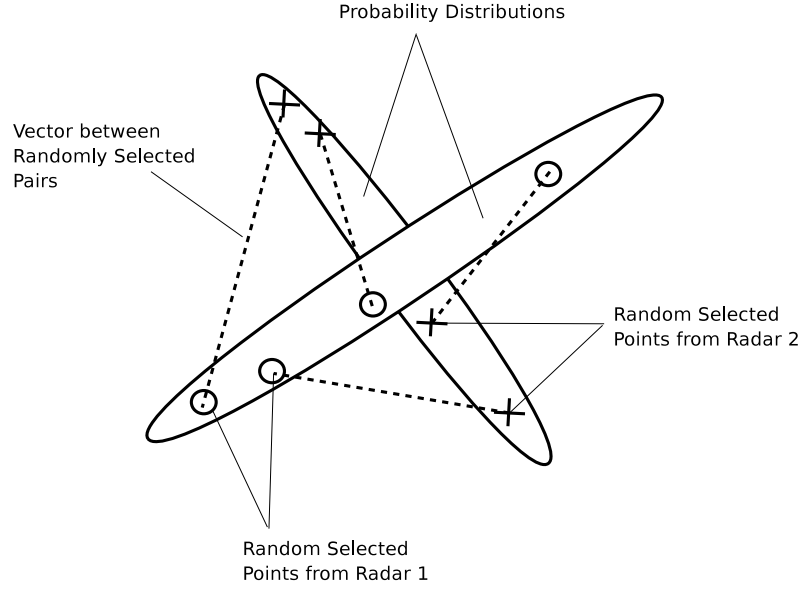


Figure 2.12: Randomly selected observation pairs

- Adjust the observations $\begin{bmatrix} R \\ \Theta \end{bmatrix}_1$ and $\begin{bmatrix} R \\ \Theta \end{bmatrix}_2$ from the two radars with the elevation estimate as illustrated in Section 2.2, where $\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_1$ and $\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_2$ are the elevation corrected ground range observations and h the estimated elevation,

$$h = \text{elevation estimate} \left(\begin{bmatrix} R \\ \Theta \end{bmatrix}_1, \begin{bmatrix} R \\ \Theta \end{bmatrix}_2 \right)$$

$$\left\{ \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_1, \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_2 \right\} = \text{ground range projection} \left[\begin{bmatrix} R \\ \Theta \end{bmatrix}_1, \begin{bmatrix} R \\ \Theta \end{bmatrix}_2, h \right].$$

- Calculate the mean $\begin{bmatrix} \hat{x}_m \\ \hat{y}_m \end{bmatrix}$ of the two elevation corrected ground range observations,

$$\begin{bmatrix} \hat{x}_m \\ \hat{y}_m \end{bmatrix} = \frac{1}{2} \left(\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_1 + \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_2 \right).$$

- Translate the mean $\begin{bmatrix} \hat{x}_m \\ \hat{y}_m \end{bmatrix}$ to the polar coordinate system as $\begin{bmatrix} \hat{R}_m \\ \hat{\Theta}_m \end{bmatrix}$ (Working in polar coordinate system makes the next step much easier),

$$\begin{bmatrix} \hat{R}_m \\ \hat{\Theta}_m \end{bmatrix} = \begin{bmatrix} \sqrt{\hat{x}_m^2 + \hat{y}_m^2} \\ \arctan\left(\frac{\hat{y}_m}{\hat{x}_m}\right) \end{bmatrix}.$$

4. Generate an arbitrary number of samples, K in total, with Gaussian distributed random variables. Noise matrix X is a K by 2 matrix with the random variables multiplied with one standard deviation of the measurement noises. The range and azimuth standard deviations are σ_R and σ_Θ respectively. Two independent noise matrices are generated for both radars. The vector $\begin{bmatrix} \tilde{R}_k \\ \tilde{\Theta}_k \end{bmatrix}$ is the sample generated by adding the noise to the sample which is calculated as follows,

$$X = \begin{bmatrix} X_{1,1}\sigma_R & \cdots & X_{k,1}\sigma_R \\ X_{1,2}\sigma_\Theta & \cdots & X_{k,2}\sigma_\Theta \end{bmatrix}$$

$$\begin{bmatrix} \tilde{R}_k \\ \tilde{\Theta}_k \end{bmatrix} = \begin{bmatrix} \hat{R}_m \\ \hat{\Theta}_m \end{bmatrix} + \begin{bmatrix} X_{k,1} \\ X_{k,2} \end{bmatrix}.$$

5. Convert all the samples $\begin{bmatrix} \tilde{R}_k \\ \tilde{\Theta}_k \end{bmatrix}$ from *both* radars to the global Cartesian coordinate system where $\begin{bmatrix} x_c \\ y_c \end{bmatrix}$ are the coordinates of the radar centre in the global Cartesian plain denoted as $\begin{bmatrix} \tilde{x}_k \\ \tilde{y}_k \end{bmatrix}$,

$$\begin{bmatrix} \tilde{x}_k \\ \tilde{y}_k \end{bmatrix} = \begin{bmatrix} \tilde{R}_k \cos(\tilde{\Theta}_k) \cos(\tilde{\phi}_k) + x_c \\ \tilde{R}_k \sin(\tilde{\Theta}_k) \cos(\tilde{\phi}_k) + y_c \end{bmatrix}.$$

Analysing sample observation pairs

Calculate the K difference vectors $\underline{\mathbf{e}}_k$ between sample $\begin{bmatrix} \tilde{x}_k \\ \tilde{y}_k \end{bmatrix}_1$ of the one radar with sample $\begin{bmatrix} \tilde{x}_k \\ \tilde{y}_k \end{bmatrix}_2$ of the other radar. The radar indices are labelled 1 and 2,

$$\underline{\mathbf{e}}_k = \begin{bmatrix} \tilde{x}_k \\ \tilde{y}_k \end{bmatrix}_1 - \begin{bmatrix} \tilde{x}_k \\ \tilde{y}_k \end{bmatrix}_2.$$

The goal of the modelling process is to create a covariance to score the difference vector between the two observations as a measure of correlation. Now the covariance is calculated from the difference vectors, as illustrated in Figure 2.13,

$$C = E(\mathbf{e}_1, \dots, \mathbf{e}_k).$$

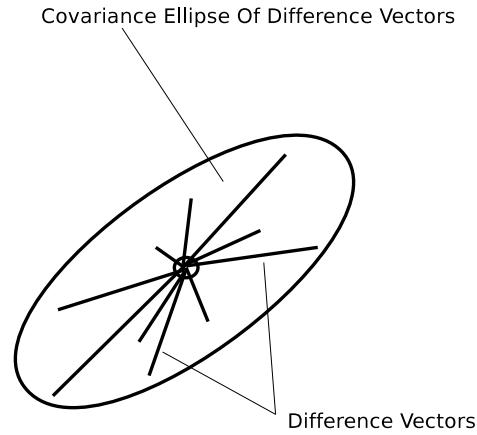


Figure 2.13: Estimated covariance from difference vectors

Scoring the original observation pair

Calculate the difference vector of the original observation pair $\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_1, \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_2$ and score it to the covariance C estimated in the previous section, as illustrated in Figure 2.14. The confidence P is thus calculated to be the integral of the Gaussian density function where A is the area outside the ellipse contour on which $\underline{\mathbf{y}}$ lies,

$$\underline{\mathbf{y}} = \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_1 - \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}_2$$

$$P = \int_A \frac{1}{2\pi |C|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\underline{\mathbf{y}}^t C^{-1} \underline{\mathbf{y}}\right).$$

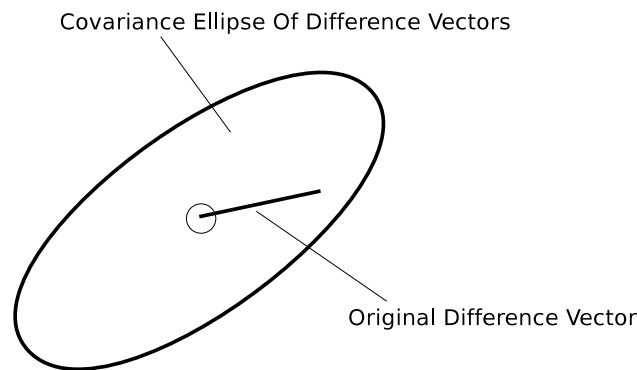


Figure 2.14: Scoring the original difference vector against the statistics

2.4.3 Implementation Remarks

The time-consuming part of this method is the creation of the covariance matrix. Theoretically the best covariance matrix will be calculated from an infinite number of points chosen from a Gaussian distribution. However, to keep the computational requirements reasonable, as few points as possible are used. This is achieved by using the so-called Sigma points as described in [10].

Sigma points are useful for non-linear transformations. The idea is simple: Instead of generating an arbitrarily large number of random points using the noise characteristics to capture the structure of the covariance, a very limited number of strategically chosen points are used. As demonstrated in [10], Sigma points provide a much better estimate of a distribution and with less processing requirements than a large amount of randomly selected points within the distribution. These $2N+1$ points, for a N dimensional distribution, are placed at the mean and one standard deviation from the mean on both sides of the principal axes to create the same distribution of the covariance as illustrated in figure 2.15. The transformed covariances are then estimated from the transformed Sigma points.

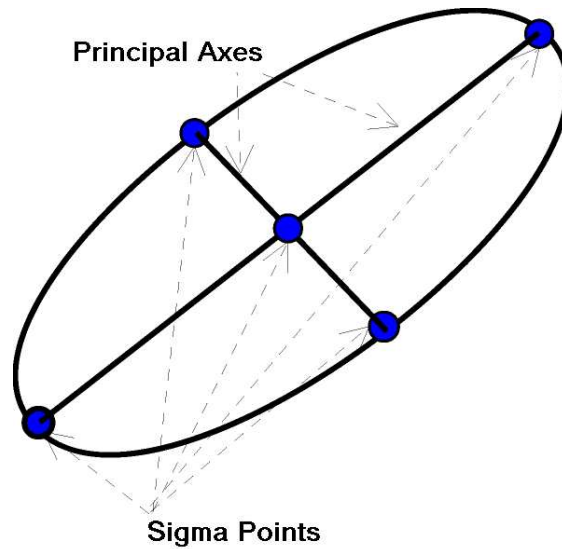


Figure 2.15: Sigma point placement in covariance

2.5 Comparison of Methods

Overview

In attempting to correlate, two types of errors occur. The first error is called a Correlation Miss (CM) and the latter a False Correlation (FC). The larger the size of the correlation window (set by a threshold), the larger the probability of FC but the probability of CM will be lower. For a smaller correlation window (set by a threshold), the opposite is true. Therefore the aim of the developer is to choose confidence thresholds for the methods to minimise the CM and FC probabilities simultaneously. Clearly both cannot be minimised at the same time and therefore a compromise has to be established depending on the application.

The system requirements of the radar deployment must determine the cost function of the CM and FC parameters during the system evaluation. In the ground-based air defence environment, it is crucial that the parameters are chosen so that two inbound enemy targets are processed as two different targets because twice the amount of defence is required to deter the threat. Therefore a high level of assurance is needed before deciding that two observations are from the same target. However, a larger correlation window could give better results in civilian ATC applications where the need will differ as the aircraft do not fly in close formations or make high G turns. The methods must therefore be compared, taking these requirements into account.

Simulation Setup

Both the Gaussian Difference Vector method (GDV) and Range Azimuth methods have a threshold setting that will greatly affect their performance. The performance

of the methods will also be affected by the position of the targets, the radar placement and measurement error characteristics. These methods can be tested for any given setup. For a complete analysis, it would be ideal, but unfortunately not possible due to the required processing power, to run the comparative simulation for every possible target formation at every possible position in the detection envelope for every possible radar configuration. Only one radar setup with two radars 20km from each other are simulated, with the targets then positioned at a few locations where the maximum performance deviations are expected, as illustrated in Figure 2.18.

To measure the FC percentage, two targets are simulated to be located 400 metres apart, flying next to each other toward the sensor array centre. The range observation from the sensors is much more accurate than the azimuth observation. Therefore it is easier to distinguish between two targets flying in a line (where the range observations are sufficient) than having them fly next to each other where primarily the azimuth observations are used for distinguishing. One radar sensor observes the one target and the other radar sensor observes the other. The question is whether the two observations from the two sensors may be correlating while they should not. The two targets could be simulated to be flying closer to each other, which would return worse FC results, but the idea remains the same. The measurement of the CM is simpler. A target at a predefined location is observed by two radar sensors with simulated errors. The question here is whether the two observations are not correlating while they should be. The correlation is run a 1000 times for every threshold setting to calculate the FC and CM probabilities.

Not only are these simulations used for system performance evaluation but these are also used to determine the threshold settings for the methods to obtain the required FC probability.

Results

It is not possible to compare the two methods by simply viewing the graphs of FC and MC rates at different thresholds. The interpretation of the results should rather be performed by applying them to operational requirements. The correlation system requirements for a military defence scenario typically requires a probability of less than 10% FCs. Therefore the threshold must be selected by inspecting the test case simulations, illustrated in Figure 2.16 by the vertical line, to satisfy the FC requirement. The corresponding CM probability at this threshold can be used to compare it with other methods with the same radar setup. The lower the CM probability at the required threshold setting to obtain 10% FC, the better the method. Six test cases were performed, of which two will now be explored in greater detail.

Test Case 1

In this simulation the target was almost at the edge of the 65 km observable envelope, at coordinate (40 km,40 km) or 57 km from the sensor centre, of the intersection of both sensor envelopes, as indicated in Figure 2.16. The Gaussian Difference Vectors method at 88.7% probability of Correlation Misses was 0.5% better than the Range Azimuth method with the threshold set for 10% probability of False Correlation.

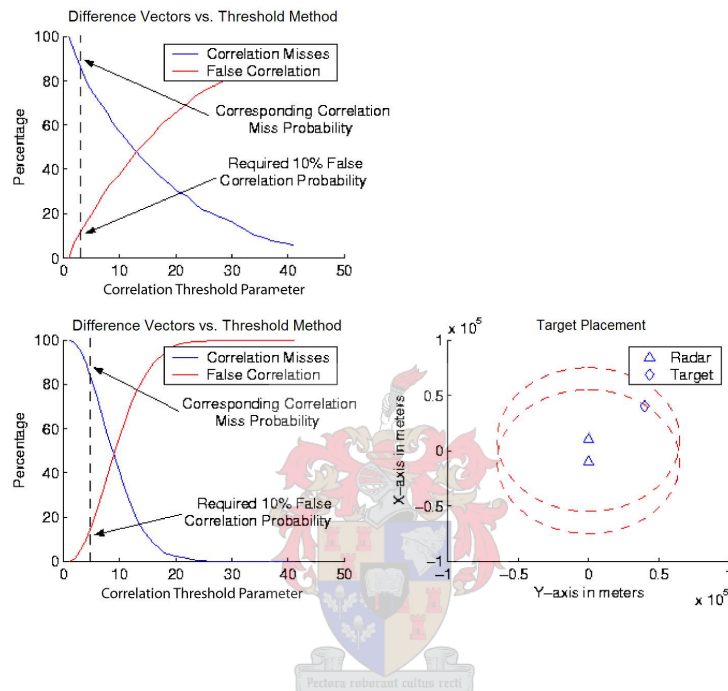


Figure 2.16: Results for Test Case 1

Test Case 2

This simulation is the same as the previous but the target is located at coordinate (20 km, 20 km) from the sensors, illustrated in Figure 2.17. The Gaussian Difference Vectors method was even better at 86.8% with the Range Azimuth method at 89.9%.

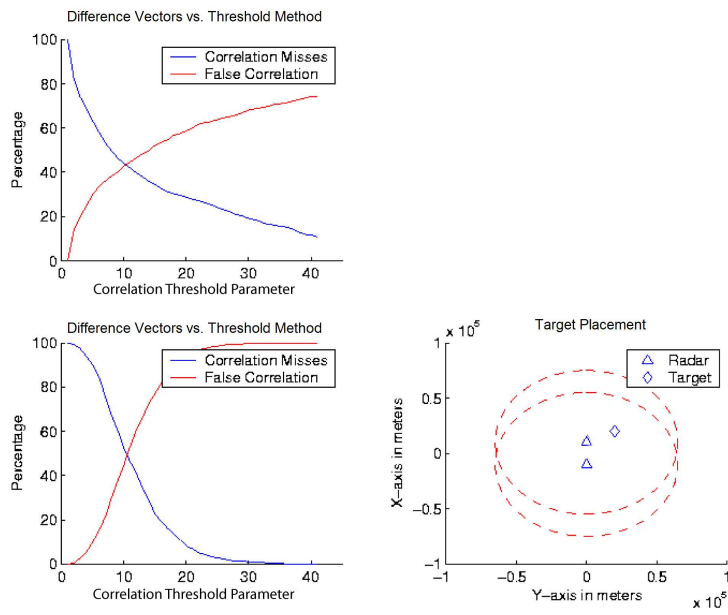


Figure 2.17: Results for Test Case 2

Comparison of all the Test Cases

Figure 2.18 shows six simulations to illustrate how the methods perform at the various locations and the results are tabulated in Table 2.1.

X(km)	Y(km)	GDV Threshold	$R\Theta$ Threshold	FC	CM GDV	CM $R\Theta$
50	0	1.45	2.45	10%	90.8%	88.4%
20	0	1.71	20.43	10%	81.5%	87.9%
40	40	2.65	4.08	10%	88.7%	89.2%
20	20	1.73	5.02	10%	86.8%	89.9%
0	20	1.83	4.85	10%	81.2%	91.5%
0	50	2.60	5.27	10%	89.7%	90.2%

Table 2.1: Correlation results from the various test locations

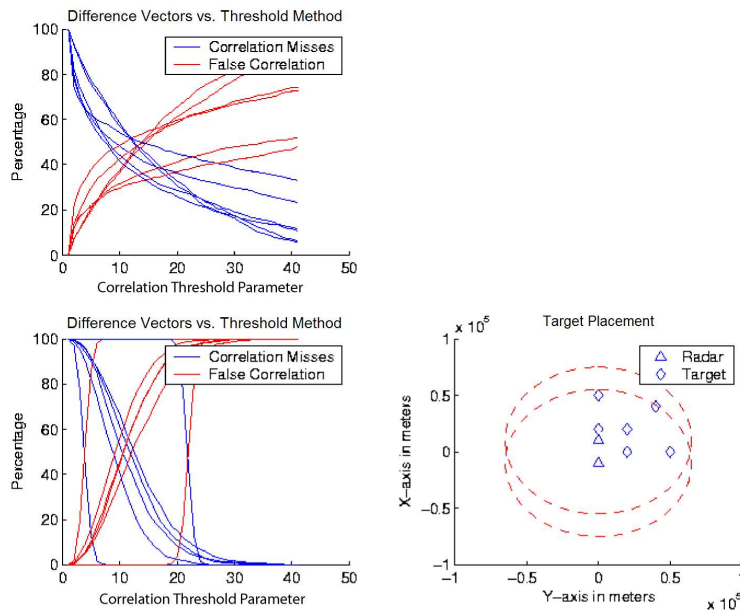


Figure 2.18: All test cases compared

As illustrated in Figure 2.19, the Range Azimuth method described in Section 2.3 was unstable when used on target positions in line with the two radars. This was due to the high dependence of the method on the characteristics of the single sensors and not the cumulative effect of the sensors.

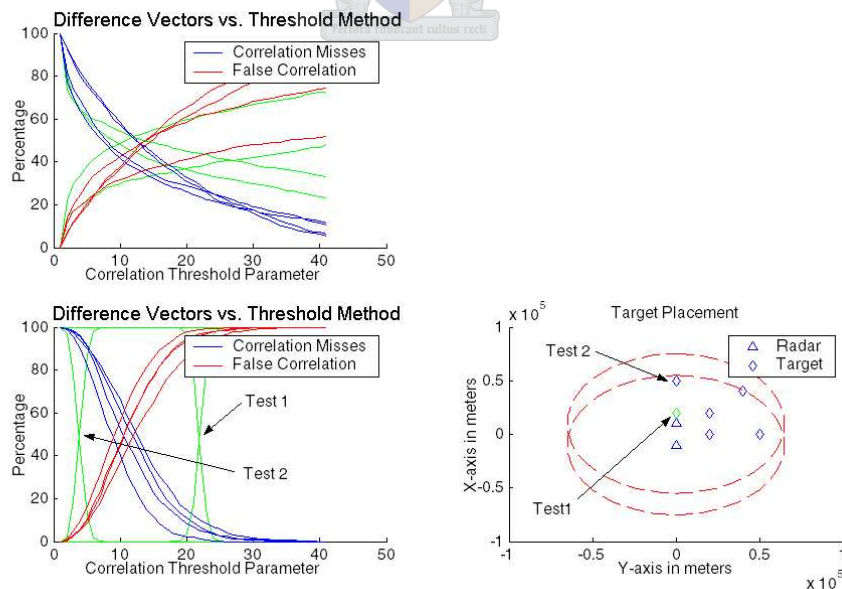


Figure 2.19: Comparative results to highlight $R\Theta$ problem

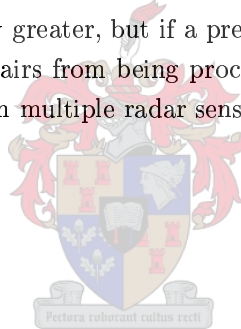
From this table, it is clear that, for most of the cases, the Gaussian Difference Vectors (GDV) method had a slightly smaller CM rate than the Range Azimuth ($R\Theta$) method.

Computational Complexity

Another requirement to be considered is the computational cost involved. The Gaussian difference vector correlation method requires much more processing power to calculate the covariance than Range Azimuth windows require. If this is a concern in the implementation of a correlation system, it is suggested that the simple range azimuth windows be used.

2.6 Conclusions

The Gaussian difference vectors method works slightly better than the Range Azimuth windows when the target is closer to the sensor array because it includes the characteristics of the two sensors in relation to each other and the target location. The computational cost of the GDV method in comparison to the Range Azimuth window is significantly greater, but if a pre-process test is run to prevent fairly improbable observation pairs from being processed, this method works well for correlating observations from multiple radar sensors.



Chapter 3

Data Association (or Gating) for Radar Tracking

The static correlation methods discussed in the previous chapter did not take into account target dynamics - track history, speed or heading. Although not directly accessible, the extra information can be derived by using tracking filters. Using this information will lead to much smaller and more accurate correlation threshold windows to correlate observations from multiple radars or associate a new observation with an existing target track. Bearing in mind the importance of IFF information, this aspect of the radar tracking system is very important in the ground-based air defence environment where airborne targets can perform high G manoeuvres close to other airborne targets. In civilian ATC, where aircraft fly predefined flight paths with low G manoeuvres far away from other aircraft, this will not be as crucial. This chapter discusses methods for associating single observations with existing target tracks for military applications by using the extra derived information.

3.1 2-D vs. 3-D Modelling with only 2-D measurements

Although 2-D radar systems can only observe the range and azimuth measurements of an aircraft that it detects, the aircraft still has an elevation component. As illustrated in Figure 3.1, the elevation has an effect on the measured range. Therefore, besides the fact that no information on the elevation of the target is available, the unknown elevation increases the uncertainty in the measured position. The maximum angle of sight and the maximum elevation limits of the radar determine the ceiling of the detection of targets. Figure 3.2 illustrates the target detection ceiling of the radar vs the range of the target. By taking into account the effect of the elevation on the measured range, the deviation of the measured range from the true ground range can be calculated as illustrated in Figure 3.3, given the true elevation. After the target is far enough from the radar so that the maximum angle of sight

has no more effect, the effect of the elevation on the measured range diminishes as the target range increases. Practically, the elevation of the target will cause the measured position to be further away from the radar than it really is. Figure 3.1 illustrates how the 2-D observations of target positions deviate from the actual positions if the aircraft is flying at an elevation of 8 000 metres.

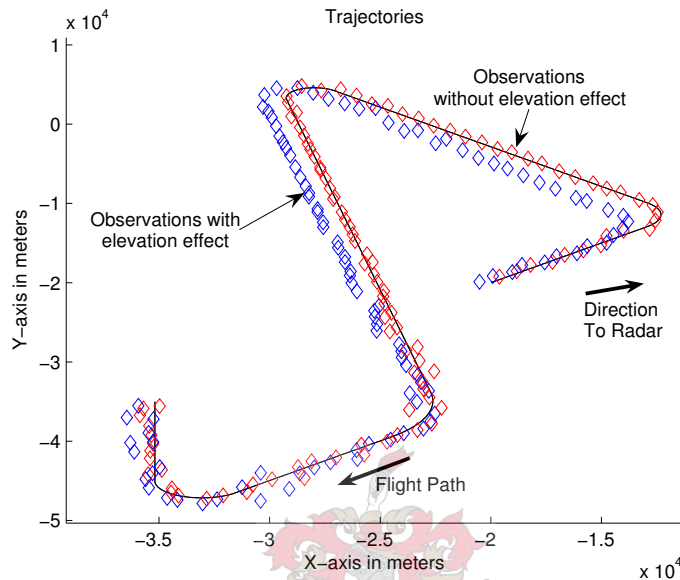


Figure 3.1: Effect of target elevation on target track observations

For even 2-D positioning, it is important to be able to measure the elevation of the target to correct the measured range, but this is not possible in 2-D radars. Using the elevation estimation method described Section 2.2, the elevation can be estimated using multiple 2-D radars, but due to high measurement noise, the method does not perform well with the range increasing.

All radars have a maximum angle of sight and a target detection ceiling. Combined, the maximum detection elevation is illustrated in Figure 3.2.

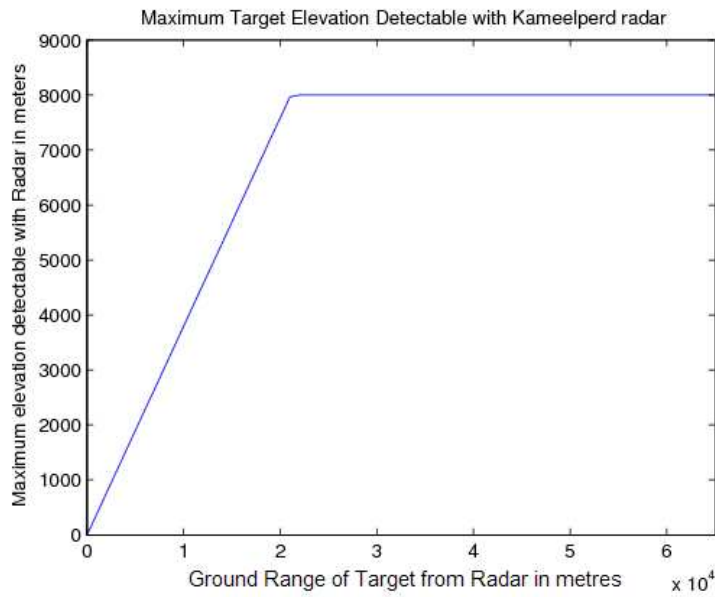


Figure 3.2: Maximum elevation detection envelope for the Kameelperd radar system

Using this maximum elevation of the target, the maximum effect of the elevation on the measured range decreases as the range increases, as illustrated in Figure 3.3.

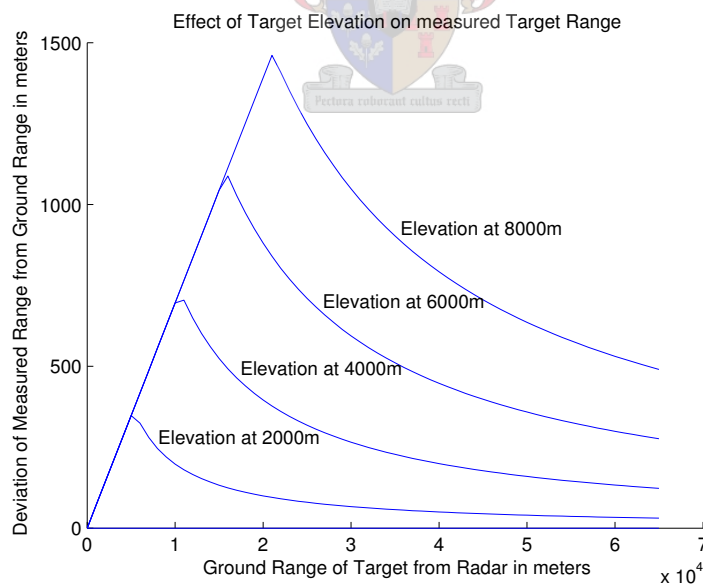


Figure 3.3: Effect of target elevation on measured range

To summarise, although the airborne target modelling, tracking and correlation processes inherently are 3-D problems, the hardware can only provide ground plane

2-D information that is affected by an elevation component. Without 3-D information, it is not possible to accurately model in 3-D. Therefore this thesis assumes a 2-D model, ignoring the lacking elevation information. Higher inaccuracies are therefore inevitable, but it can be accounted for by modelling larger uncertainties.

3.2 Range Azimuth Window

3.2.1 Overview

When a new observation is detected, the question is: To which existing target track does this observation belong? The observation is necessary to update the current position of the target, but of which track is this observation an update? Or could this be the first sighting of a new target to track?

The Range Azimuth window correlation method draws a rectangular window around the predicted position of each target track at the time instant of the new observation in question. The predicted position is obtained by using the laws of motion to the last known target position and derived current velocity of the track history. This extrapolation is described in detail in Section 4.2.1. If the new observation lies within this rectangular window, it correlates with the target track and is used by the tracking filters described in Chapter 4 to update the target track.

This method is currently used by Reutech Radar Systems in conjunction with an adaptive $\alpha\beta$ filter, as described in Section 4.3.1. This is because the window size is defined according to the state of the filter parameters and thus depends on the tracker method. The filter parameters are adapted as the target changes its manoeuvring profile. At the same time, the window sizes are also adapted accordingly to take into account the target behaviour. The window must be large enough on the one hand to be able to catch new observation updates if the target is manoeuvring, it must be smaller on the other hand to prevent observations from other targets to be used to update the target track. It is important to balance these conflicting requirements.

This method can only give a true or false result to the correlation and not a confidence value such as the methods which will be described in Section 3.3. This makes the Range Azimuth method undesirable in a multiple target environment, as an observation can test true for more than one target track, but is described and implemented to compare against a new method described in Section 3.3.

3.2.2 Algorithm

The algorithm is fairly straightforward. For reasons explained in Section 4.2.3, the tracking filter operates on the Cartesian coordinate plane, but because this algorithm operates easier in a Polar coordinate system, the predicted target position and the observation have to be transformed to this system. The algorithm is designed for the 2-D problem, because, as stated in Section 3.1, radar systems such as the Kameelperd only measure in 2-D data. The algorithm is therefore only in 2-D,

ignoring the elevation component, but adding the elevation component is a trivial matter. The following equations translate the predicted positions from the filter, $\begin{bmatrix} x_p \\ y_p \end{bmatrix}$, and observed, $\begin{bmatrix} x_o \\ y_o \end{bmatrix}$, positions to the polar coordinate system using the radar location $\begin{bmatrix} x_c \\ y_c \end{bmatrix}$ as the polar origin,

$$\begin{bmatrix} R_o \\ \Theta_o \end{bmatrix} = \begin{bmatrix} \sqrt{(x_o - x_c)^2 + (y_o - y_c)^2} \\ \arctan\left(\frac{(y_o - y_c)}{(x_o - x_c)}\right) \end{bmatrix}$$

$$\begin{bmatrix} R_p \\ \Theta_p \end{bmatrix} = \begin{bmatrix} \sqrt{(x_p - x_c)^2 + (y_p - y_c)^2} \\ \arctan\left(\frac{(y_p - y_c)}{(x_p - x_c)}\right) \end{bmatrix}.$$

After the translation, the prediction and the observation are compared in the following equation for correlation, using the range threshold R_w and azimuth threshold Θ_w ,

$$[|\Theta_p - \Theta_o| < \Theta_w] \text{ and } [|R_p - R_o| < R_w].$$

3.2.3 Illustration

Window Definition

Figures 3.4 and 3.5 illustrate how the correlation windows are defined around the predicted target position in relation to the radar sensor. Note how the major axes of the windows correspond to the expected azimuth observation error of the sensor and not just to the possibility of the target location and track heading.

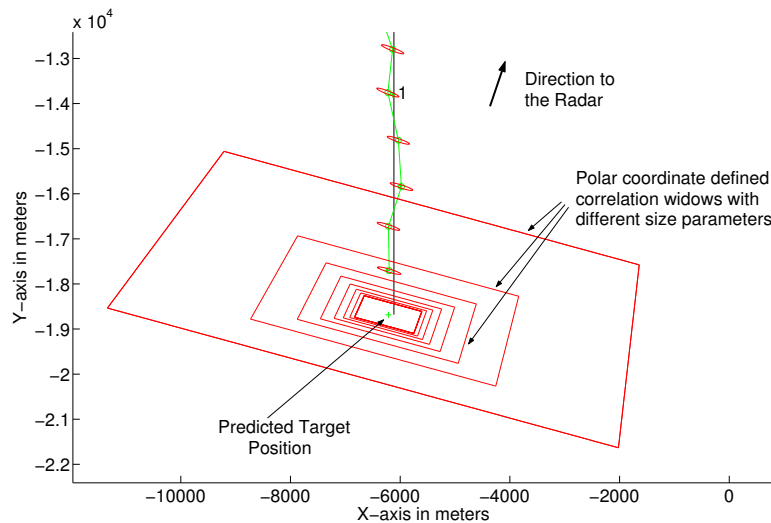


Figure 3.4: Correlation window definition A

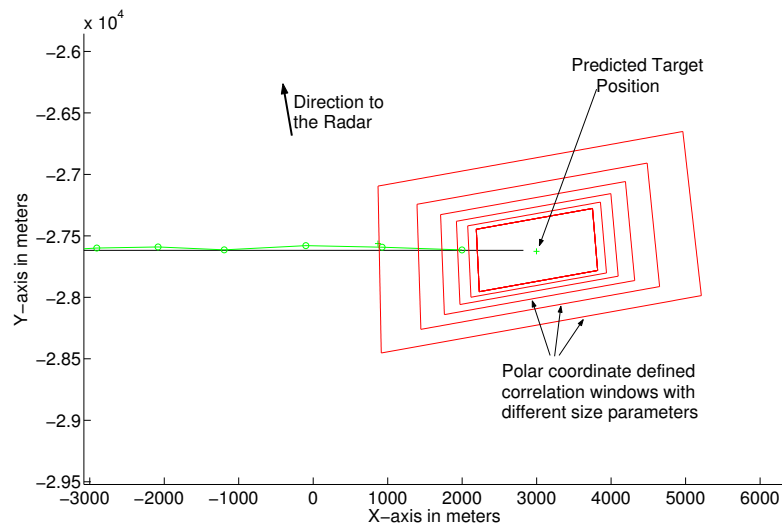


Figure 3.5: Correlation window definition B

Operation of window with a tracker system

Figure 3.6 illustrates the correlation method in action. Note how the size of the correlation window stays small until the observation falls outside the correlation threshold window at the onset of the manoeuvre. The red window indicates where the observation update falls outside the correlation window and how the indicated bigger window caught the observation.

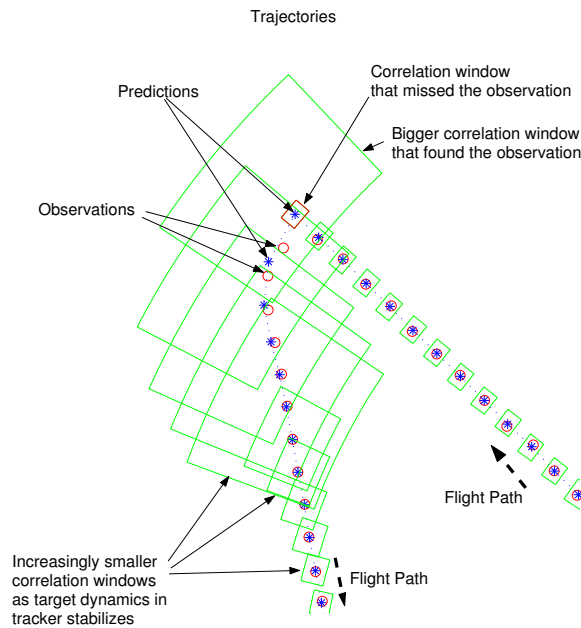


Figure 3.6: Correlation window operation with the $\alpha\beta$ filter

3.3 Propagated Particle Covariance Estimation

3.3.1 Motivation

The Range Azimuth method described in Section 3.2 depends on the filtering algorithm *state* and not the *possible positions* of the target. However, the new method attempts to correlate, using the *possible positions* of the target.

The heading and velocity can be determined from the first derivatives of the track filter. Thus it is relatively easy to extrapolate where the target will be at any instant in time after the last observation. Unfortunately this is only true if the target was not performing a manoeuvre since the last update. Current fighter jet technology allows a pilot to perform up to 9 G's of acceleration force. Even if the target is not manoeuvring, the ability to predict the position of the target, assuming there were no accelerations, relies heavily on the accuracy of the track velocity, heading and last known position. The idea behind the Propagated Particle is to model all these uncertainties by creating particles that will exhibit moderate and extreme cases of what the target could be doing, and to build a covariance model around the predicted positions of these particles for correlation purposes.

3.3.2 Position prediction of accelerating particles

This section describes how the position of a particle can be determined given the dynamics of the particle. A simple transition matrix used in tracking will not work because the G forces of a banking manoeuvre causes centrifugal acceleration and not a linear acceleration.

Motivation

The effect of acceleration on the position of a particle can normally be calculated if given the total time of movement t , the initial position $\underline{\mathbf{x}}$, the initial velocity $\underline{\mathbf{v}}$ and the acceleration $\underline{\mathbf{a}}$, as follows,

$$\underline{\mathbf{x}}_{k+1} = \frac{1}{2}\underline{\mathbf{a}}t^2 + \underline{\mathbf{v}}t + \underline{\mathbf{x}}_k.$$

Aircraft banking manoeuvres do not adhere to this linear type of acceleration. Instead it can be described as a more circular movement around a centre point with constant speed and centrifugal acceleration. This leads to an acceleration that, on the Cartesian plane, is non-linear. A geometric derivation of the banking aircraft movement is much simpler and easier to compute.

Algorithm

1. Firstly, the centre of the trajectory circle along which that particle will travel must be calculated given the acceleration Υ and initial speed V of the particle at time step k . The radius R of the circle is required for this and is calculated as follows,

$$R = \frac{V^2}{\Upsilon}.$$

2. The greater the acceleration, the smaller the circle radius. The opposite is also true. Please note the difference of the signs in calculating the centres for a left or right banking manoeuvre. Calculate the trajectory circle centre $\underline{\mathbf{c}}$, or $\begin{bmatrix} c_x \\ c_y \end{bmatrix}$, given the initial position $\underline{\mathbf{p}}_k$ or $\begin{bmatrix} p_{k,x} \\ p_{k,y} \end{bmatrix}$ and H_k as the initial heading of the particle,

Right Turn :

$$\underline{\mathbf{c}} = \underline{\mathbf{p}}_k + R \begin{bmatrix} \cos\left(\frac{\pi}{2} - H_k\right) \\ -\sin\left(\frac{\pi}{2} - H_k\right) \end{bmatrix}$$

Left Turn :

$$\underline{\mathbf{c}} = \underline{\mathbf{p}}_k + R \begin{bmatrix} \cos\left(\frac{\pi}{2} + H_k\right) \\ \sin\left(\frac{\pi}{2} + H_k\right) \end{bmatrix}.$$

3. The next step is to calculate the rate of angular turn ω around the centre and

the total turn angle θ at the specified time instant T into the manoeuvre,

$$\omega = \sqrt{\frac{\Upsilon}{R}}$$

$$\theta = \omega T.$$

4. If the centre $\underline{\mathbf{c}}$ of the banking circle, the radius R , the total turn angle θ and the time instant T into the manoeuvre is given, the predicted position of the particle $\underline{\mathbf{p}}_{k+1}$ can be calculated as follows,

Right Turn :

$$\underline{\mathbf{p}}_{k+1} = \underline{\mathbf{c}} + R \begin{bmatrix} -\cos\left(\theta + \frac{\pi}{2} - H_k\right) \\ \sin\left(\theta + \frac{\pi}{2} - H_k\right) \end{bmatrix}$$

Left Turn :

$$\underline{\mathbf{p}}_{k+1} = \underline{\mathbf{c}} + R \begin{bmatrix} \cos\left(\theta - \frac{\pi}{2} + H_k\right) \\ \sin\left(\theta - \frac{\pi}{2} + H_k\right) \end{bmatrix}.$$

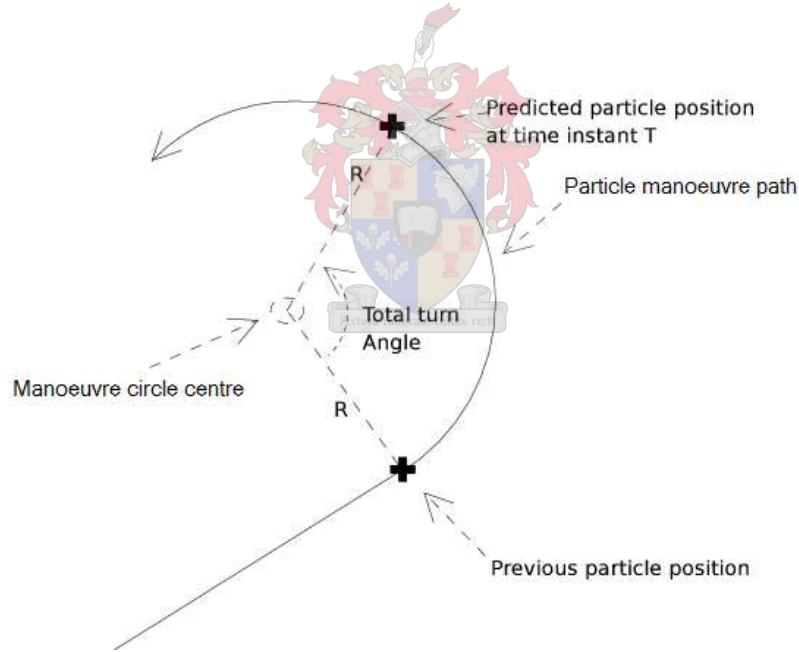


Figure 3.7: Geometric illustration of the particle manoeuvre formulation

3.3.3 Propagated Particles Covariance Correlation Algorithm

Stepwise procedure:

1. As stated earlier, the aim is to model where the target could be, given the uncertainties in the target model and possible behaviour. The position, speed,

heading and acceleration uncertainties have to be accounted for. Also to be considered it that the target can perform banking manoeuvres to the left or right with various amounts of acceleration. The first step is to determine the maximum position, speed s_{max} , heading h_{max} and acceleration a_{max} deviations which are expected from the predicted target state. The uncertainty of the predicted position can also be modelled by generating five sigma ($\underline{\sigma}_{1..5}$) points, given the observation noise covariance. The sizes of these limits will depend largely on the practical application of this method. The maximum expected acceleration a_{max} can be up to 9 Gs in a military airborne combat environment.

- Noise vectors are then created with elements to model moderate and maximum noise in the uncertainties. For greater accuracy in the modelling process, more elements at smaller intervals can be placed in these vectors, but that will require more processing. A simple example of how to define these noise vectors could be as follows, with $\underline{\mathbf{a}}_e$, $\underline{\mathbf{s}}_e$, $\underline{\mathbf{h}}_e$ and $\underline{\mathbf{p}}_e$ as the acceleration, speed, heading and position uncertainties respectively,

$$\begin{aligned}\underline{\mathbf{a}}_e &= \begin{bmatrix} 0 & \frac{1}{3}a_{max} & \frac{2}{3}a_{max} & a_{max} \end{bmatrix} \\ \underline{\mathbf{s}}_e &= \begin{bmatrix} -s_{max} & 0 & s_{max} \end{bmatrix} \\ \underline{\mathbf{h}}_e &= \begin{bmatrix} -h_{max} & 0 & h_{max} \end{bmatrix} \\ \underline{\mathbf{p}}_e &= \begin{bmatrix} \underline{\sigma}_1 & \underline{\sigma}_2 & \underline{\sigma}_3 & \underline{\sigma}_4 & \underline{\sigma}_5 \end{bmatrix}.\end{aligned}$$

- Next, the K particles are defined as all possible arrangements of the acceleration, speed, heading and position noises which can be added to the last known state of the target. In the illustrated example above, the acceleration vector has seven entries (three banking left, one not manoeuvring and three banking right), velocity has three entries, heading has three entries and position has five entries. This creates $K = 7 \times 3 \times 3 \times 5 = 315$ particles.
- Using the time lapse T from the last update and the states of the particles, the predicted state of the K particles $\underline{\mathbf{p}}_k$ is calculated using the method described in Section 3.3.2.
- The radar observation noise is also modelled with five sigma points, using the covariance of the sensor noise with the predicted particle positions as the centres. This will increase the number of predicted particle positions of the target to $K = 315 \times 5 = 1575$ particles.
- The predicted mean $\underline{\mathbf{p}}_m$ and covariance C using these predicted particles $\underline{\mathbf{p}}_{1..K}$ are then calculated,

$$\underline{\mathbf{p}}_m = E \left[\underline{\mathbf{p}}_{1..K} \right]$$

$$C = E \left[\left(\mathbf{p}_{1..K} - \mathbf{p}_m \right)^2 \right].$$

7. The last step is simply to calculate the confidence P of the possible update observation χ which is being correlated with the track, using the estimated mean \mathbf{p}_m , covariance C and the area A outside the contour of the covariance ellipse on which observation χ lies,

$$P = \int_A \left(\frac{1}{2\pi |C|^{\frac{1}{2}}} \exp \left(-\frac{1}{2} (\chi - \mathbf{p}_m)^t C^{-1} (\chi - \mathbf{p}_m) \right) \right).$$

8. A suitable threshold is then used to decide whether or not this observation is a valid track update. In the simulations, a threshold of 0.36 was set to ensure that the update had to be within one standard deviation of the predicted mean.

3.3.4 Illustration

The particles are used to model the various target state uncertainties. To illustrate this modelling process, figures are compiled to display the effect of these uncertainties. At first the effects of the position, velocity, heading and acceleration uncertainties on the particles are displayed individually and then displayed together to illustrate the creation of the model.

Particle modelling acceleration uncertainty

Figure 3.8 illustrates particles positioned to reflect the possible manoeuvres the target could have undertaken. The particles perform banking manoeuvres to the left and right with centrifugal accelerations that cause forces of 1 to 10 Gs. The diamonds in the figure with the labels “Dxx” are detections with sequence numbers.

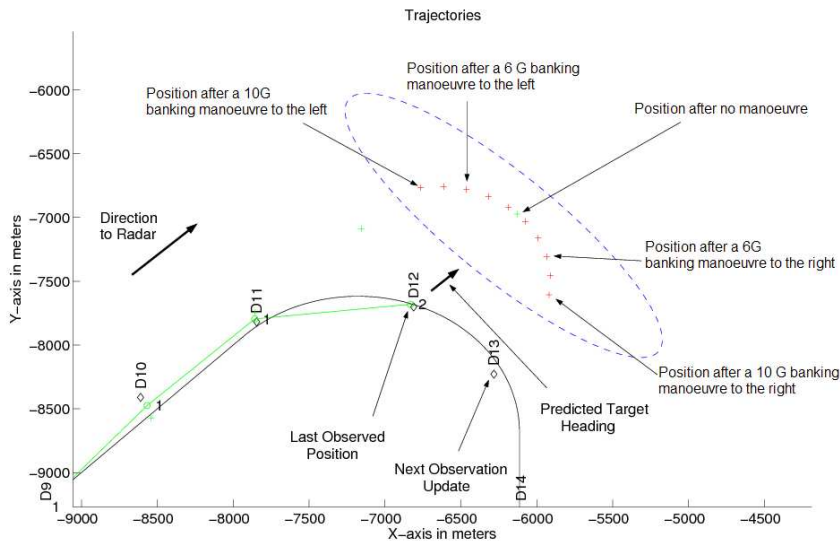


Figure 3.8: Effect of manoeuvre/banking acceleration on particles

Particle modelling of heading uncertainty

Figure 3.9 illustrates the particle positioning with 3 G banking manoeuvres to the left and right with 20% heading errors to the left and right. Compare this figure with the previous one to notice the slight difference in heading and acceleration deviations.

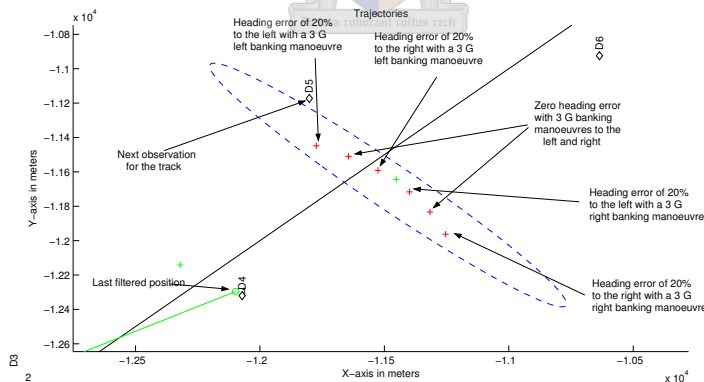


Figure 3.9: Effect of heading on particles

Particle modelling of speed uncertainty

Figure 3.10 illustrates the predicted particle positions to allow for a 30% higher or lower velocity error. The particles are again simulated to perform 3 G banking manoeuvres to the left and right.

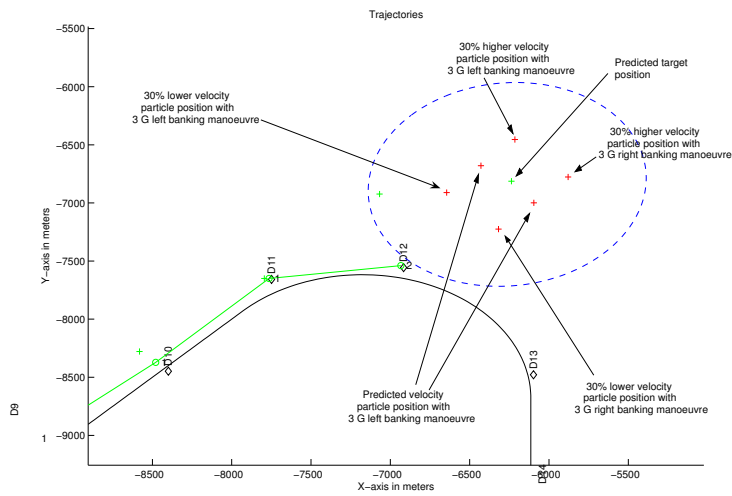


Figure 3.10: Effect of velocity on particles

Particle modelling of radar position measurement uncertainty

This part is the key to the final formation of the covariance ellipse because the distribution of these points completely depends on the position of the target in relation to the sensor. Figures 3.11 and 3.12 illustrate this aspect by placing sigma point groups at the estimated position of the particles after the 3G manoeuvres.

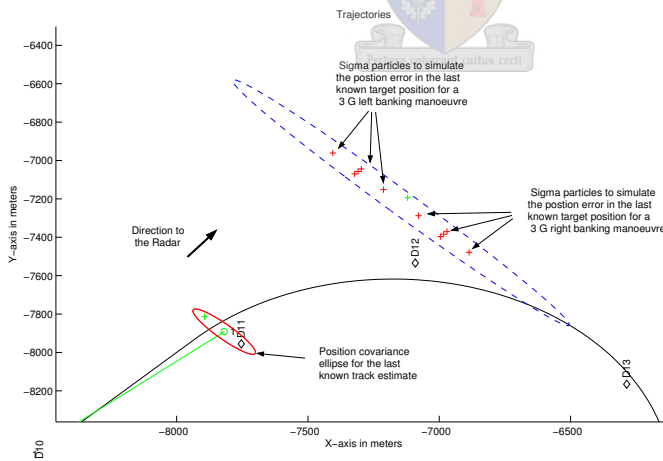


Figure 3.11: Effect of process position covariance on particles (1)

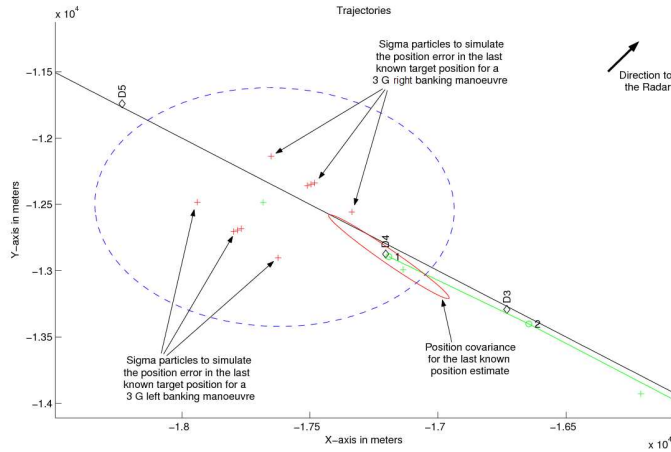


Figure 3.12: Effect of process position covariance on particles (2)

The last known target position noise covariance and the update observation noise covariance create a cumulative position covariance that has to be taken into account. In other words, the uncertainty of the last known target position and the uncertainty of the new observation have to be combined to form a realistic uncertainty. Figure 3.13 illustrates this by generating another five sigma particles with the sensor noise covariance for every one of the original five sigma particles. 10 G banking manoeuvres to the left and right were simulated to separate the two groups of particles for illustration purposes.

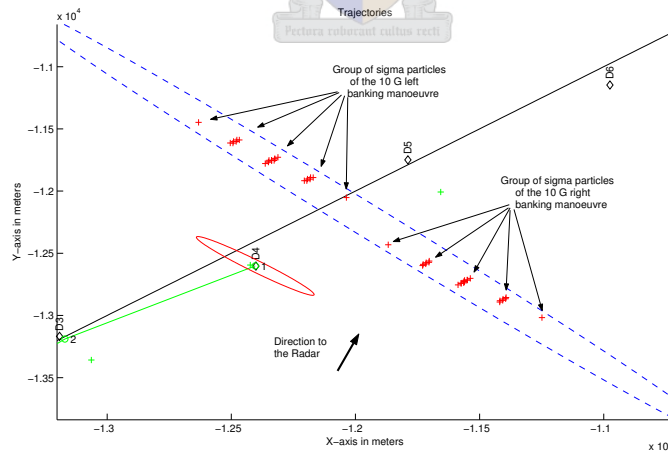


Figure 3.13: Effect of cumulative process and observation covariances on particles

Final Particle Modelling Product

Now all the particles are propagated and plotted in Figure 3.14. The blue ellipse is the estimated covariance of all the particle positions. The covariance can easily be scaled to alter the threshold of the correlation.

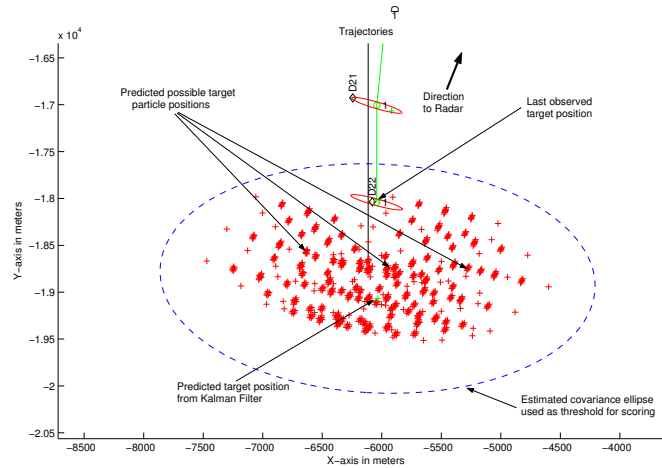


Figure 3.14: Particle positions after taking all unknowns into consideration

3.4 Comparison of Methods

Criteria

There are two criteria against which the methods must be compared. Firstly the ability of the window to find the update observation during a manoeuvre must be determined. Secondly the window must be as small as possible at all times to avoid observations from other targets being associated with the track. The two criteria are opposite requirements because, by increasing the window size, the ability to find the update observations is greater, but the probability of false associations is also greater. On the other hand, decreasing the window size decreases the probability of false associations, but the ability to find the update observation is also decreased.

The method performance comparison is performed by simulating the target observations of the radar and calculating the minimum window sizes for both methods required to associate the observation with the track. The size of the smallest correlation window, as defined by the Range Azimuth method to allow the observation to be associated with the track, is then compared with the size of the smallest window defined by the Propagated Particle correlation window to allow the observation to be associated with the track.

Results from a straight flight path

The Propagated Particles windows only gradually increases in size. Except for the closest observations, Propagated Particle windows are much smaller than the Range Azimuth windows which increase rapidly as the targets move further away from the radar. Figures 3.15 and 3.16 illustrate this aspect.

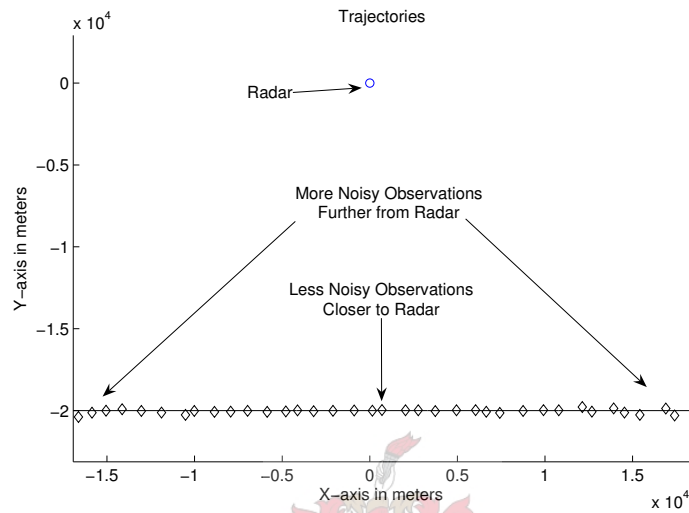


Figure 3.15: Correlation test for a straight path

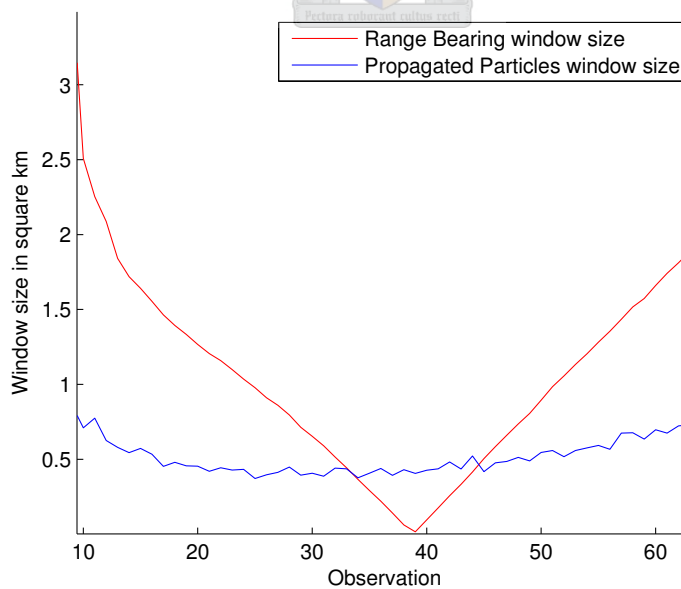


Figure 3.16: Window size results for a straight path

Results from a 4 G manoeuvre

Again the Propagated Particles window sizes are much smaller because the windows are designed to catch observations from flight paths of up to 10 Gs. Therefore it easily associated the observations from this 4 G manoeuvre with the track. The Range Azimuth windows need to be very large to associate the observations because the predicted positions from the tracker are far from the true update observations. Figures 3.17 and 3.18 illustrate this aspect.

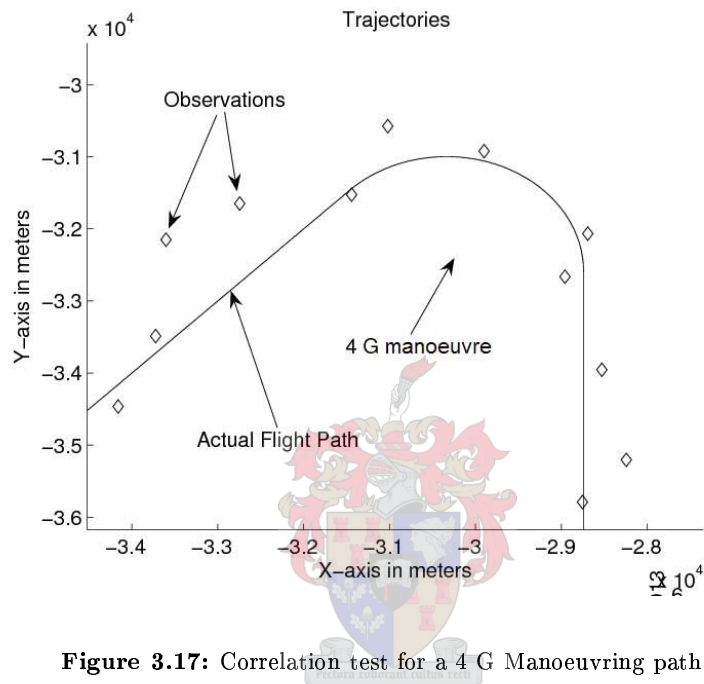


Figure 3.17: Correlation test for a 4 G Manoeuvring path

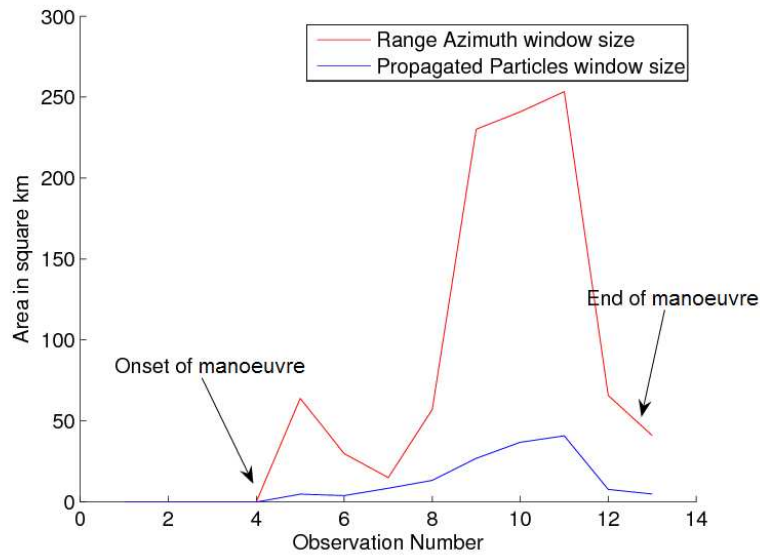


Figure 3.18: Window size results for a 4 G Manoeuvring path

Results from flight path with straight and manoeuvring sections

The results from a full Z flight path show how much more effective the Propagated Particle windows are because tailor-made windows are designed for every situation, whether the observations are close to or far from the sensor or whether the target is moving at a high or low velocity. Figures 3.19 and 3.20 illustrate this aspect.

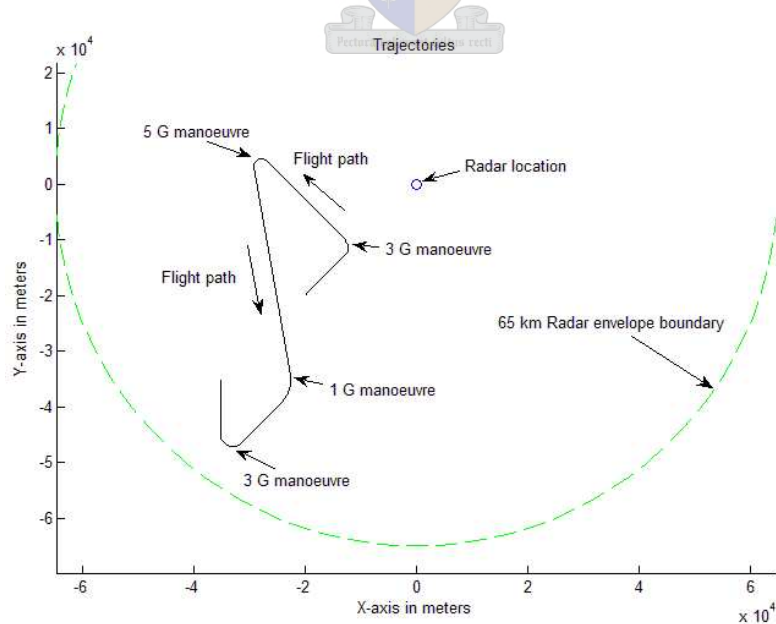


Figure 3.19: Correlation test for path with straight and manoeuvring sections

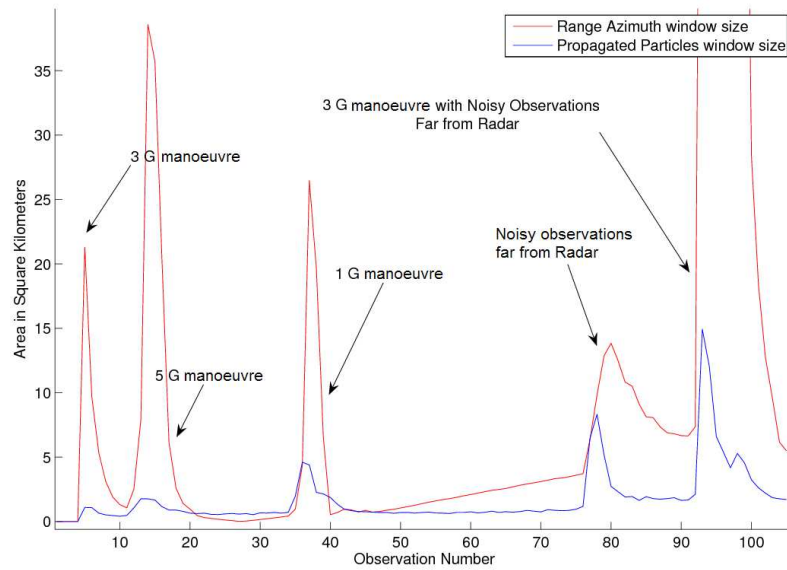


Figure 3.20: Window size results for a straight and manoeuvring paths

3.5 Conclusion

The Propagated Particle window sizes are generally smaller in all the test flight paths. This is because the window is tailor-made for the target if the radar position relative to the target and the target dynamics are given. The Range Azimuth window, on the other hand, does not perform this computation and relies on rather large predefined windows to find the update. Another benefit of the Propagated Particle window method when compared with the Range Azimuth window method is that it returns a confidence of how well the observation correlates with the track prediction. This is unavailable in the Range Azimuth method but is crucial for advanced tracking algorithms like the multiple hypothesis tracking system.

The added computational requirements of the Propagated Particle window are several orders of magnitude higher but practical with modern processors.

The Propagated Particle method suffers the limitation that the speed and heading of the target must be known. Thus the method can only be used after the track is more than about three updates old.

Chapter 4

Target Tracking

Radars provide periodical measurements on the position of targets, but has no information of how the new measurements relate to previous measurements. Thus the first problem is to find the relationship between measurements to form a track history of the targets. This problem becomes even trickier as the radar may fail to detect the target from time to time. The solutions to these correlation window problems, which are described in Chapter 3, must now be integrated into a tracking system. The measurement noise is the second problem to be dealt with. This noise can be reduced by applying appropriate filters to smooth the target trajectory. The filter performance relies largely on knowledge about the target movement, which gives rise to the third issue of target movement modelling. All these issues must be solved and implemented to be used in a real-time military tracking system.

4.1 The Track Manager

The Track Manager manages the tracking system. It interfaces the components of the system, as illustrated in Figure 4.1. It uses the correlation methods discussed in Chapter 3 to determine which target track to associate with the new observations. Each target track has to be initialised and updated (according to the model and filter used), maintained and stored in a track database.

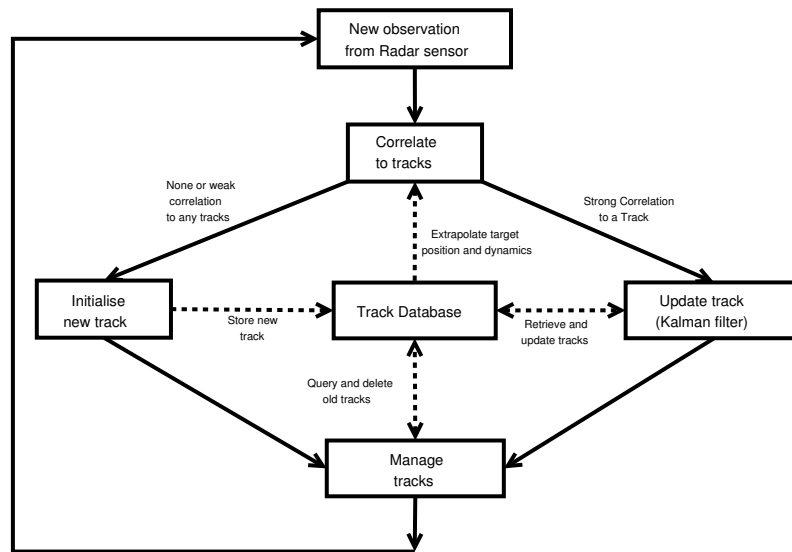


Figure 4.1: Track Management Flow Diagram

4.1.1 Correlation with existing tracks

In the multiple target tracking environment, the correlation methods must be able to give an indication of how well a target correlates with an existing track. Just having a true or false output from the correlation method is not desirable because one observation might be assigned to more than one track if the targets are in close proximity to each other. The correlation method discussed in Section 3.3 is based on a covariance calculated from the possible positions of the target. Given the observation and the covariance, a confidence can be calculated and used as a measure of correlation. Although it might still happen that an observation is assigned to more than one track because the calculated confidences are equal, this method is already a huge step forward. Multiple hypothesis tracking systems (which are outside the scope of this thesis) attempt to provide even better solutions to this problem.

The possible observation is compared with all nearby tracks and associated with the track with the maximum confidence. A new track is established if none of the confidences are above a predetermined threshold. Figure 4.2 illustrates the problem of having multiple tracks close to a possible update observation.

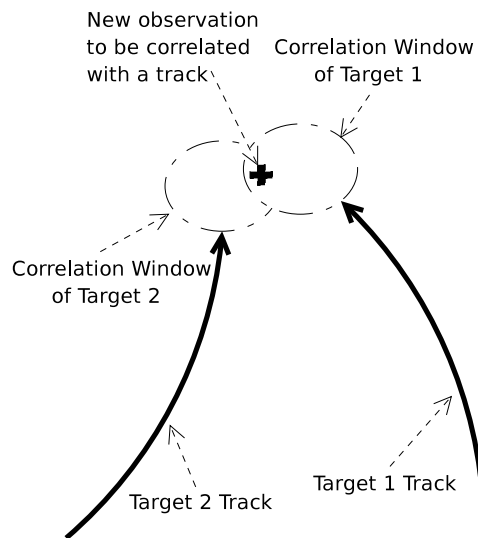


Figure 4.2: Correlation windows for track updating

4.1.2 Management of tracks

Calculating the assignment confidences of new observations on tracks that have missed observation updates is increasingly in danger of wrong update observation assignment because the process covariance and possible search area for possible observation updates increase rapidly. This problem is solved by deactivating and/or deleting these tracks from the database that have not been updated for a certain amount of time. This amount of time must be carefully selected as the target might not have been detected for a few seconds and then may appear again. The system must be able to realise that this could still be the same target in order to retain the IFF information of the track. This routine should be executed after every observation or as an interrupt handler at set intervals. Figure 4.3 illustrates the update search area of a track that has not been updated for a few scans.

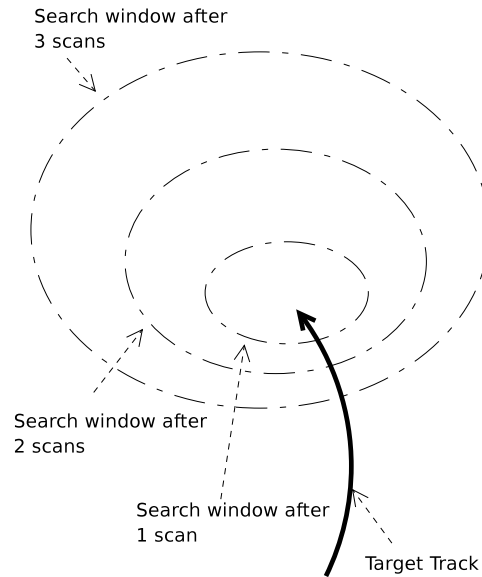


Figure 4.3: Correlation window size increase

4.2 Target Models

The Kalman filter, which is described in detail in Section 4.3.2, relies on models to predict the target movement from measured positions and known measurement noises. Choosing a model to fit the realistic movement of the target will greatly increase tracking success and noise reduction. Incorrect modelling of the target's movement leads to higher inaccuracies in position estimation or may even result in divergence of the estimated target position from the actual position.

The first step is to understand the airborne target behaviour. A commercial airliner has a very predictable straight flight path. A model with a constant velocity should suffice to describe its movement. On the other hand, a fighter jet performing a high G banking manoeuvre is better modelled as having constant acceleration. The next step is to derive the mathematical equations to support the model.

This thesis discusses models for a constant velocity, a constant acceleration and a mixture of the two.

4.2.1 Constant Velocity Target Model

This model assumes that the target is not experiencing any acceleration and is therefore moving at a constant velocity. A commercial airliner with a straight flight path is a typical example. The radar is assumed to be measuring range and azimuth only to give a 2-D position and the position will be described by the x and y elements on the Cartesian axes. The velocity components on these axes will be described by \dot{x} and \dot{y} . There are no elements for storing the acceleration because it is assumed to be zero. Bear in mind that the aircraft is moving in 3-D, but with only 2-D

measurements available, the target model can only be described in 2-D. Although the 2-D assumption in the model causes estimation inaccuracies, it will still model the target well. The additive noise in each of the state elements are assumed to be zero meaned white Gaussian. According to Newton's Law, at step $k + 1$ the four elements that will constitute the state vector $\underline{\mathbf{x}}_{k+1}$ of the target will be as follows, where T is the time difference between k and $k + 1$,

$$\underline{\mathbf{x}}_{k+1} = \begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} x_k + T \dot{x}_k \\ y_k + T \dot{y}_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix}.$$

Transition matrix A is a 4x4 matrix used to perform numerous multiplications to update the position and velocity estimates in an elegant manner with T as the time from the current state vector and the predicted state vector. This matrix makes it very easy to program and debug the equations of Newton's Law for motion,

$$A = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Now the transition matrix A propagates the previous target state vector $\underline{\mathbf{x}}_k$ to the next predicted target state vector $\underline{\mathbf{x}}_{k+1}$ by a simple matrix multiplication,

$$\underline{\mathbf{x}}_{k+1} = A \underline{\mathbf{x}}_k.$$

4.2.2 Manoeuvring Target Model

The manoeuvring model assumes that the target is experiencing acceleration. A fighter jet performing a banking manoeuvre or simply accelerating provides examples of this. The state vector $\underline{\mathbf{x}}_k$ will require six elements in the 2-D case to model the position (x, y) , velocity (\dot{x}, \dot{y}) and the acceleration (\ddot{x}, \ddot{y}) components. Again the additive noise in each of the state elements are assumed to be zero meaned white Gaussian. According to Newton's Law, at step $k + 1$ the six elements that will constitute the state vector $\underline{\mathbf{x}}_{k+1}$ of the target will be as follows where T is the time difference between k and $k + 1$,

$$\underline{\mathbf{x}}_{k+1} = \begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \\ \ddot{x}_{k+1} \\ \ddot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} x_k + T \dot{x}_k + \frac{1}{2}T^2 \ddot{x}_k \\ y_k + T \dot{y}_k + \frac{1}{2}T^2 \ddot{y}_k \\ \dot{x}_k + T \ddot{x}_k \\ \dot{y}_k + T \ddot{y}_k \\ \ddot{x}_k \\ \ddot{y}_k \end{bmatrix}.$$

Again the transition matrix A is used to perform the Newtonian Law transition multiplications to update the position, velocity and acceleration estimates in an elegant manner, with T as the time from the current state vector and the predicted state vector. The only difference between this transition matrix and the one described in Section 4.2.1 is that this one is extended to 6x6 to include the acceleration,

$$A = \begin{bmatrix} 1 & 0 & T & 0 & \frac{1}{2}T^2 & 0 \\ 0 & 1 & 0 & T & 0 & \frac{1}{2}T^2 \\ 0 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

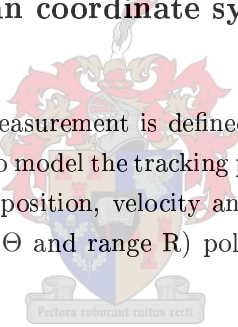
Again the transition matrix A also propagates the previous target state vector $\underline{\mathbf{x}}_k$ to the next predicted target state vector $\underline{\mathbf{x}}_{k+1}$ by a simple matrix multiplication,

$$\underline{\mathbf{x}}_{k+1} = A \underline{\mathbf{x}}_k.$$

4.2.3 Polar vs Cartesian coordinate system models

Coordinate system selection

The radar sensor observation measurement is defined in range R and azimuth Θ . Therefore it is natural and easy to model the tracking problem in range and azimuth. The state vector modelling the position, velocity and acceleration components of the target in the 2-D (azimuth Θ and range R) polar coordinate system at time step k is,



$$\underline{\mathbf{x}}_k = \begin{bmatrix} R \\ \Theta \\ \dot{R} \\ \dot{\Theta} \\ \ddot{R} \\ \ddot{\Theta} \end{bmatrix}.$$

On the other hand, the state vector could model these 2-D components in the Cartesian coordinate system as,

$$\underline{\mathbf{x}}_k = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \\ \ddot{x} \\ \ddot{y} \end{bmatrix}.$$

What could be described in one coordinate system as a linear movement, could

very well be non-linear in another. This is especially true in radar tracking applications where the relationship between the measurements and the system process is nonlinear. Specifically, the measurements are in range and azimuth but the aircraft movement is best described by a Cartesian coordinate system. If the polar coordinate system is used to model the target movement directly from the range and azimuth measurements, a non-linear modelling problem will result. Approximation methods described in [8] can be used to approximate a linear transition function to fit the non-linear problem, but these linearisation techniques are prone to divergence.

However, if the range and azimuth measurements are translated to the Cartesian coordinate system, the target movement could be modelled linearly with a transition as matrix shown before,

$$A = \begin{bmatrix} 1 & 0 & T & 0 & \frac{1}{2}T^2 & 0 \\ 0 & 1 & 0 & T & 0 & \frac{1}{2}T^2 \\ 0 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

It is very easy to translate positions between the Polar and Cartesian coordinate systems but it is a bit more difficult to translate covariances, which are also required. However, the covariances can be transformed using Monte Carlo simulations. The position measurements can be translated by means of a one-to-one translation as follows,

$$x = R \sin(\Theta)$$

$$y = R \cos(\Theta)$$

$$\Theta = \arctan\left(\frac{y}{x}\right)$$

$$R = \sqrt{x^2 + y^2}.$$

The radial velocity projection problem

An airborne target moves in a 3-D Cartesian plane. Therefore the velocity of the aircraft can be described fully by using 3 velocity components in the Cartesian plane. Certain radar systems can provide Doppler or range rate measurements by observing the phase shift in the reflected radar signal. Doppler basically is the rate at which the target is moving toward or from the radar. This, however, is only a projection of the true 3-D velocity components on the radial plane or range axis as illustrated in Figure 4.4. Therefore the Doppler measurement cannot be used directly to determine actual velocity of the target. In other words, it is not possible

to translate the velocity from the Polar to the Cartesian coordinate system, but it is possible to translate the Cartesian target velocity to the Polar coordinate system.

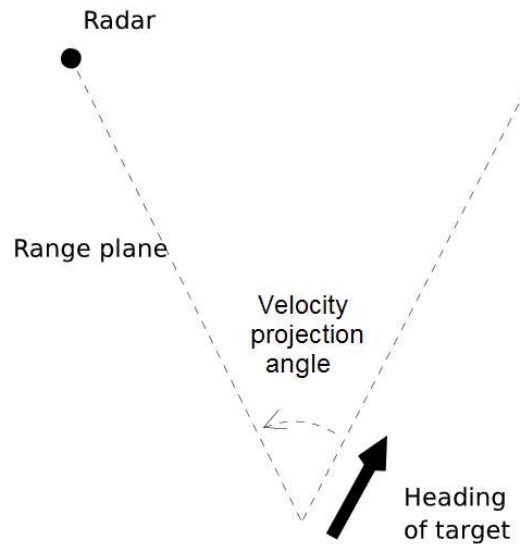


Figure 4.4: Velocity projection to the Range plane

Thus if the target is moving perpendicularly to the range axis, the range rate component will be zero, irrespective of the actual velocity of the target. This is illustrated in Figure 4.5. Another extreme occurs when the target is moving in line with the range axis of the radar where the range-rate measurement will be equal to the target velocity as illustrated in Figure 4.6.

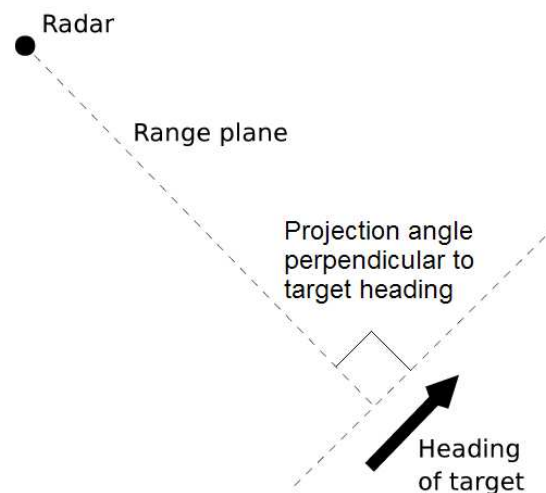


Figure 4.5: Velocity projection perpendicular to the target heading

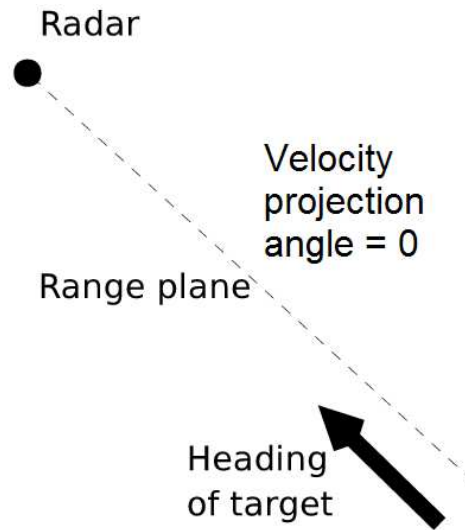


Figure 4.6: Velocity projection directly in the range plane

The calculation of the range-rate \dot{R} measurement is the speed V of the target, multiplied by the cosine of the projection angle θ_{proj} between the heading of the target and the direction to the radar from the target,

$$\dot{R} = V \cos(\theta_{proj}).$$

This translation is not reversible as the range-rate \dot{R} measurement is only a component of the target velocity. Thus the Doppler measurement can only be used to update the model if the model is defined in the polar plane. Although these limitations prevent the update of target position and velocity, it can be used to calculate a form of pseudo acceleration which will alert the system to possible manoeuvring as described in Section 4.4.

To summarise: The tracked aircraft moves in a Cartesian space and not a polar one. Although the radar measurements are in the polar plane and modelling can attempt to translate the movement equations to a polar plane, it is better to rather translate the radar measurements to the Cartesian plane which is native to the target's movement. In essence, the target's movement must not be modelled to fit a model but a model must be designed to fit the target's movement.

4.3 Filter Methods

In the radar system architecture, tracking is performed in a real-time Track-While-Scan system. This is an iterative procedure where the filtering must give the best estimate of the target position as the observations become available.

4.3.1 Alpha-Beta ($\alpha\beta$) Filter

Introduction

The alpha-beta filter has two gain parameters (which lie between 0 and 1) that weigh between the predicted position (based on target track history/dynamics) and the observed position of the target. These values are derived through trial and error. The larger the value of the parameter, the more the observation is relied upon for the current position of the target and the smaller the value of the parameter, the more the predicted target position is trusted. On the one extreme, where the parameter is equal to 1, the filtered target position will be exactly the observation of the target at that time instant. Thus no filtering is applied. On the other extreme if the parameters are set to 0 the prediction is used, ignoring input from the sensor, clearly an undesirable state of affairs.

Modelling the acceleration with a third (gamma) parameter is important to account for acceleration when the target is manoeuvring. The implementation used by Reutech Radar Systems ignores this parameter and is replaced with varying values of alpha and beta to compensate for this.

Algorithm

The filtering process consists of two phases. Firstly, a prediction phase where the previous state k of the target track is used to extrapolate the current position $k + 1$ of the target given the time lapse T since the previous state. The second phase corrects the predicted position $\bar{\mathbf{x}}_{k+1}$ and velocity $\bar{\mathbf{v}}_{k+1}$ of the target state with the observation \mathbf{z}_{k+1} and updates the previously estimated position $\hat{\mathbf{x}}_k$ and velocity $\hat{\mathbf{v}}_k$ of the track to $\hat{\mathbf{x}}_{k+1}$ and $\hat{\mathbf{v}}_{k+1}$. Note that the predicted velocity $\bar{\mathbf{v}}_{k+1}$ equals the previous velocity $\hat{\mathbf{v}}_k$ because the acceleration is modelled as zero,

Prediction:

$$\bar{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_k + T \cdot \hat{\mathbf{v}}_k$$

$$\bar{\mathbf{v}}_{k+1} = \hat{\mathbf{v}}_k$$

Correction:

$$\hat{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_{k+1} + \alpha (\mathbf{z}_{k+1} - \bar{\mathbf{x}}_{k+1})$$

$$\hat{\mathbf{v}}_{k+1} = \bar{\mathbf{v}}_{k+1} + \frac{\beta}{T} (\mathbf{z}_{k+1} - \bar{\mathbf{x}}_{k+1}).$$

Parameter selection

The Figure 4.7 illustrates the effect of the $\alpha\beta$ parameters on the filter performance in a 3 G target manoeuvre. The filtered track in blue had the parameters set to a very large setting to rely more on the observation than the prediction. The filtered red track on the other hand had its parameters set very small to rely on the prediction

more than the observation. This prediction leads to a divergence because the filter model does not include the acceleration from the manoeuvre.

Parameter	α	β
Red Track	0.345	0.055
Blue Track	0.833	0.5

Table 4.1: Parameter setup for $\alpha\beta$ filters

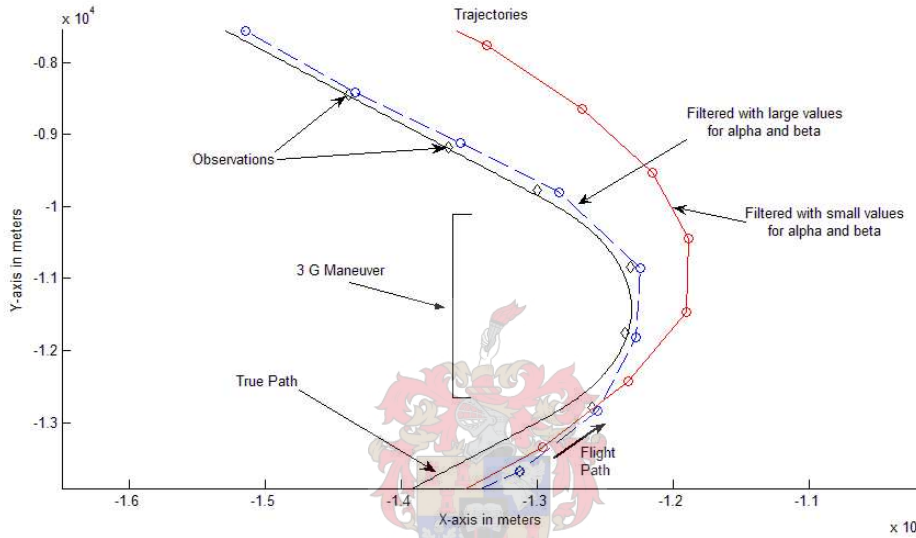
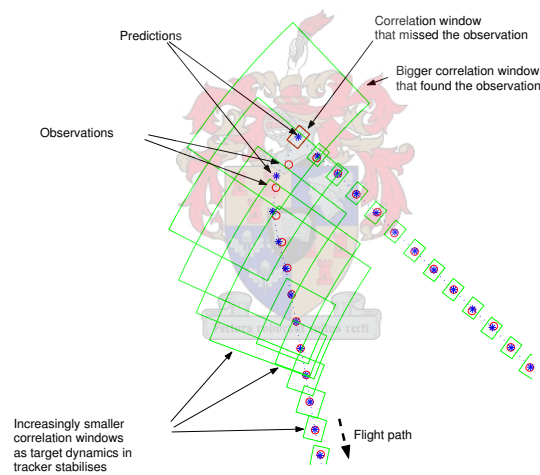


Figure 4.7: Effects of $\alpha\beta$ parameters on filter performance

Varying Parameter Method

As discussed earlier, the parameters have to be carefully selected for the best performance given the behaviour of the target. However, these parameters cannot remain the same because rapid changes in target movement need to be accommodated. The $\alpha\beta$ parameters are adjusted in real-time to avoid losing the ability to follow manoeuvres and to keep noise reduction on straight paths high. The basic idea is to use the distance of the new update observation from the predicted position to determine if the target is performing a manoeuvre. The filter parameters are selected according to this distance. More specifically, windows of increasing size with the predicted position as its centre is used for this purpose, as illustrated in Figure 4.8, just like in Section 3.2 where it was used to correlate observations to an existing track. Only here it is used to determine the filter parameters that are used. The smaller the manoeuvre that is determined, the smaller the parameter size. The greater the manoeuvre that is determined, the greater the parameter size to increase noise reduction on straight paths. The parameters used by Reutech Radar Systems are tabulated in Table 4.2.

Index	α Parameter	β Parameter	Range Size (m)	Azimuth Size (rad)
1	1	1	4074	0.4559
2	0.833	0.5	2037	0.2279
3	0.7	0.3	2037	0.2279
4	0.6	0.2	1018.5	0.1140
5	0.524	0.142	679.0	0.0760
6	0.464	0.107	509.2	0.0570
7	0.417	0.083	407.4	0.0456
8	0.378	0.067	339.5	0.0380
9	0.345	0.055	291.0	0.0326
10	0.318	0.055	254.6	0.0285
11	0.295	0.055	254.6	0.0285
12	0.257	0.055	254.6	0.0285
13	0.242	0.055	254.6	0.0285
14	0.228	0.055	254.6	0.0285

Table 4.2: $\alpha\beta$ Filter and correlation window parametersFigure 4.8: Correlation window operation with the $\alpha\beta$ filter

The array of parameters can either be accessed/changed according to a sequential assigning method or in a “jumpable” assigning method. With sequential assigning, the parameters are forced to be gradually increased or decreased when the observations continue to fall inside or outside the defined distance from the prediction. The “jumpable” assigning method allows the parameters to be changed to any level given the distance of the observation from the prediction. Figure 4.9 illustrates how much better the filtering is that is provided by the “jumpable” assigning method. The “jumpable” method is able to quickly keep track of the target during the manoeuvre whilst quickly changing to give strong noise reduction on the straight

sections. The sequential assigning method initially also had strong noise reduction on the straight section, but was unable to return to the path quickly enough after the manoeuvre.

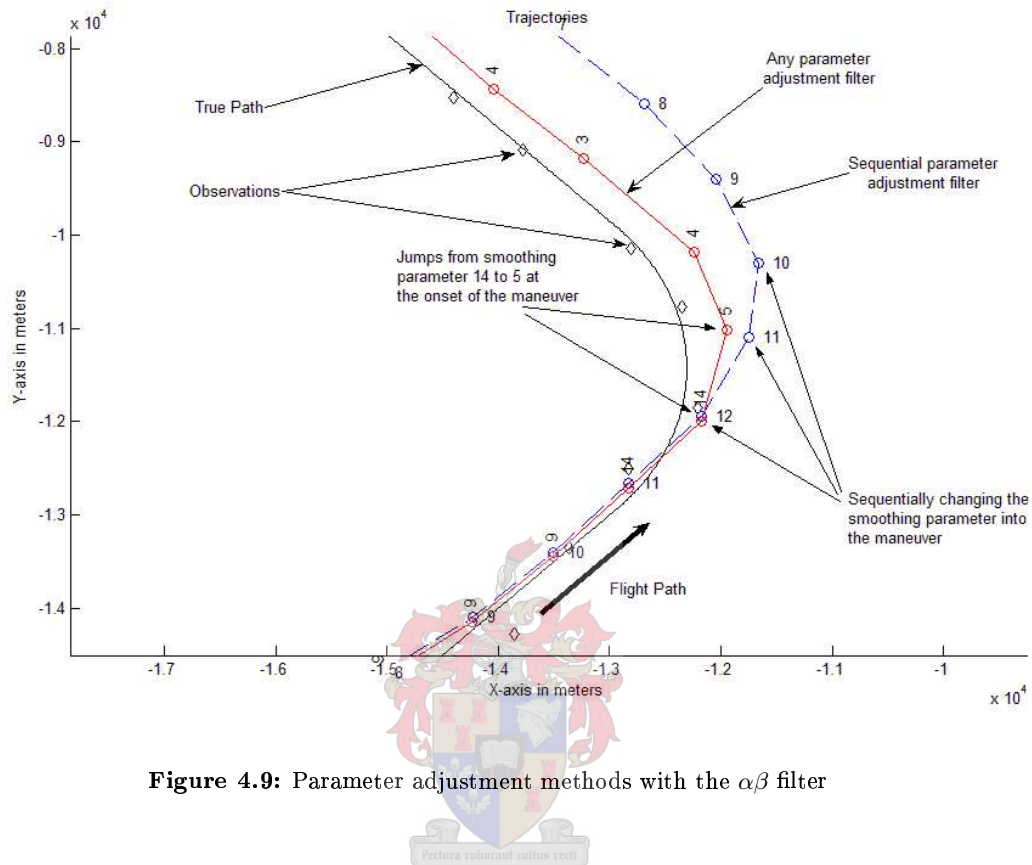


Figure 4.9: Parameter adjustment methods with the $\alpha\beta$ filter

Initialisation

The state vector consisting of the position and velocity estimates is initialised with the position set to the first observation and the velocity estimate set to zero. The $\alpha\beta$ parameters are set to the largest setting in the parameter array of Table 4.2 in order to reflect the uncertainty of the track.

4.3.2 Kalman Filter

A brief introduction

The Kalman filter is named after Rudolph E. Kalman, who published a paper [11] in 1960 on an iterative solution to the discrete observation filtering problem. The inclusion of process and measurement noise made this solution stand out from other predict-correct algorithms. Unlike other methods (like the $\alpha\beta$ filter described in Section 4.3.1) that requires external intervention to calibrate the gain parameters, the Kalman filter calibrates itself by including the process and measurement noise covariances.

Observation and Dynamic Models

The Kalman filter assumes that the current target state $\underline{\mathbf{x}}_{k+1}$ is linearly related to the previous target state $\underline{\mathbf{x}}_k$ through a transition matrix A and that the current target state $\underline{\mathbf{x}}_{k+1}$ is linearly related to the observation $\underline{\mathbf{z}}_k$ through an observation translation function B as illustrated in the next equations. The process noise $\underline{\mathbf{w}}_k$ and observation noise $\underline{\mu}_k$ are both white noise with no cross-correlation,

$$\underline{\mathbf{x}}_{k+1} = A_k \underline{\mathbf{x}}_k + \underline{\mathbf{w}}_k$$

$$\underline{\mathbf{z}}_k = B(\underline{\mathbf{x}}_k) + \underline{\mu}_k.$$

Prediction and Correction Algorithm

The first step of every Kalman filter iteration is to calculate the predicted next state vector $\bar{\underline{\mathbf{x}}}_{k+1}$ and the predicted process covariance estimate \bar{P}_{k+1} with the following two equations, given the transition matrix A and the process noise covariance Q_k ,

$$\begin{aligned}\bar{\underline{\mathbf{x}}}_{k+1} &= A\underline{\mathbf{x}}_k \\ \bar{P}_{k+1} &= AP_kA^T + Q_k.\end{aligned}$$

The predicted state vector $\bar{\underline{\mathbf{x}}}_{k+1}$ is calculated by applying the transition matrix A defined in Sections 4.2.1 and 4.2.2 to the previous state vector $\underline{\mathbf{x}}_k$. The predicted process covariance \bar{P}_{k+1} is also calculated by applying the transition matrix A to the previously estimated process covariance P_k and adding the process noise covariance Q_k . Observation matrix C is the relation between the state vector $\underline{\mathbf{x}}_k$ and the observation $\underline{\mathbf{z}}_k$.

After the prediction phase, the filter corrects the predicted estimate using the observation $\underline{\mathbf{z}}_{k+1}$ from the radar with the measurement noise covariance R associated with the observation. The following two equations correcting the prediction, which were derived by Kalman and presented in his paper [11], are

$$K = \bar{P}_{k+1}C^T (C\bar{P}_{k+1}C^T + R)^{-1}$$

$$P_{k+1} = (I - KC)\bar{P}_{k+1}$$

$$\bar{\underline{\mathbf{x}}}_{k+1} = \bar{\underline{\mathbf{x}}}_k + K(\underline{\mathbf{z}}_{k+1} - C\bar{\underline{\mathbf{x}}}_{k+1}).$$

Applying Kalman filtering to radar position prediction

The manoeuvring target model described in Section 4.2.2 requires a six-element state vector $\underline{\mathbf{x}}_k$ to store the 2-D position, velocity and acceleration elements,

$$\underline{\mathbf{x}}_k = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \\ \ddot{x} \\ \ddot{y} \end{bmatrix}.$$

A 6x6 transition matrix A based on Newton's Law is used to predict the next state vector $\underline{\mathbf{x}}_{k+1}$ where T is the time lapse between $\underline{\mathbf{x}}_k$ and $\underline{\mathbf{x}}_{k+1}$,

$$A = \begin{bmatrix} 1 & 0 & T & 0 & \frac{1}{2}T^2 & 0 \\ 0 & 1 & 0 & T & 0 & \frac{1}{2}T^2 \\ 0 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

The observation translation function B is the relation between observation $\underline{\mathbf{z}}_k$ and the state vector $\underline{\mathbf{x}}_k$ as follows, where x and y are the position components of state vector $\underline{\mathbf{x}}_k$,

$$B(\underline{\mathbf{x}}_k) = \begin{bmatrix} \sqrt{x^2 + y^2} \\ \arctan\left(\frac{y}{x}\right) \end{bmatrix}.$$

Because of this specific radar tracking problem, the process noise $\underline{\mathbf{w}}_k$ is defined in Cartesian coordinates and observation noise $\underline{\mu}_k$ is defined in the polar coordinate system, as discussed in Section 4.2.3.

In this application of the Kalman filter, the measurement noise covariance R will differ for every observation due to the repositioning of the target in relation to the radar. Therefore the noise covariance R has to be calculated, as described in Section 4.5, at every observation because it is not constant as in most filtering applications. The observation matrix C defines what, in the state vector, can be directly measured by the radar. As the radar can only measure 2-D positions, the matrix C will be,

$$C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Initialisation

The Kalman filter gain parameters are self-adjusting in order to obtain an optimal fit with the data, but the initial process covariance P_0 and the process noise covariance Q need to be set before the first iteration.

The process noise covariance Q was determined by trial and error for the best performance, but more tweaking might result in even better results. The smaller it is, the more the trust in the estimated process model. It is kept very small in the case of the 4-state Kalman filter modelling a straight flight path, because it is assumed that the velocity of the target is constant. The position noise standard deviations were set to 100 m and the velocity noise variances were set to 10 ms^{-1} . This is illustrated in matrix Q_4 ,

$$Q_4 = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}.$$

A much larger noise covariance is set when manoeuvring is expected by setting the acceleration noise standard deviations to at least $20ms^{-2}$. The position and velocity standard deviations were also set to 100 m and 10 ms^{-1} , as illustrated in matrix Q_6 ,

$$Q_6 = \begin{bmatrix} 100 & 0 & 0 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 0 & 0 & 20 \end{bmatrix}.$$

The process covariance P_0 is set to be very large, 10^5 for all variances, at the initialisation of the filter because the model has no information on the dynamics of the target at this stage. Therefore almost total trust must be placed in the observation and none in the prediction for the first few updates. Both the constant velocity and manoeuvring models adhere to this rule. Although this seems extreme, the Kalman filter quickly converges to an appropriate value as the first few observations are processed.

The initial process covariance P_0 for the constant velocity model is,

$$P_0 = \begin{bmatrix} 10^5 & 0 & 0 & 0 \\ 0 & 10^5 & 0 & 0 \\ 0 & 0 & 10^5 & 0 \\ 0 & 0 & 0 & 10^5 \end{bmatrix}.$$

The initial process covariance P_0 for the manoeuvring model is extended to also

include acceleration noise as follows,

$$P_0 = \begin{bmatrix} 10^5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10^5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10^5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^5 \end{bmatrix}.$$

Using the Z flight path as defined in Figure 4.10, the process covariances are shown in Figure 4.11.

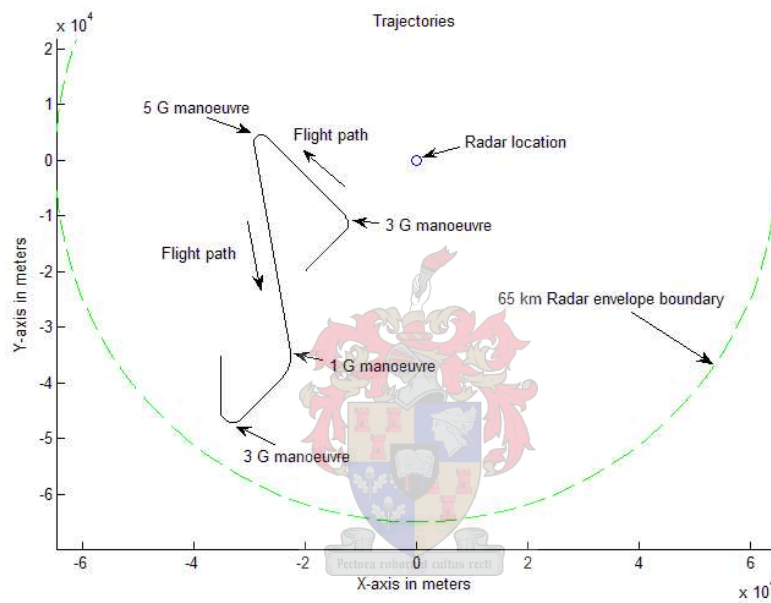


Figure 4.10: Z flight test simulation setup

Figure 4.11 illustrates how the rather large initial process covariance very quickly and automatically converges to smaller, more suitable values. The position covariance also illustrates how it adapts to the changing nature of the target movement and position in relation to the sensor. The values that are presented are the absolute size of the 2-D position σ_p , speed σ_s and acceleration σ_a standard deviation estimates obtained from the process covariance matrix P_k . The diagonal variance elements in the process covariance matrix P_k are used to calculate the position, velocity and acceleration standard deviations as follows,

$$P_k = \begin{bmatrix} \sigma_x^2 & & & & & \\ & \sigma_y^2 & & & & \\ & & \sigma_{\dot{x}}^2 & & & \\ & & & \sigma_{\dot{y}}^2 & & \\ & & & & \sigma_{\ddot{x}}^2 & \\ & & & & & \sigma_{\ddot{y}}^2 \end{bmatrix}$$

$$\sigma_p = \sqrt{\sigma_x^2 + \sigma_y^2}$$

$$\sigma_s = \sqrt{\sigma_{\dot{x}}^2 + \sigma_{\dot{y}}^2}$$

$$\sigma_a = \sqrt{\sigma_{\ddot{x}}^2 + \sigma_{\ddot{y}}^2}$$

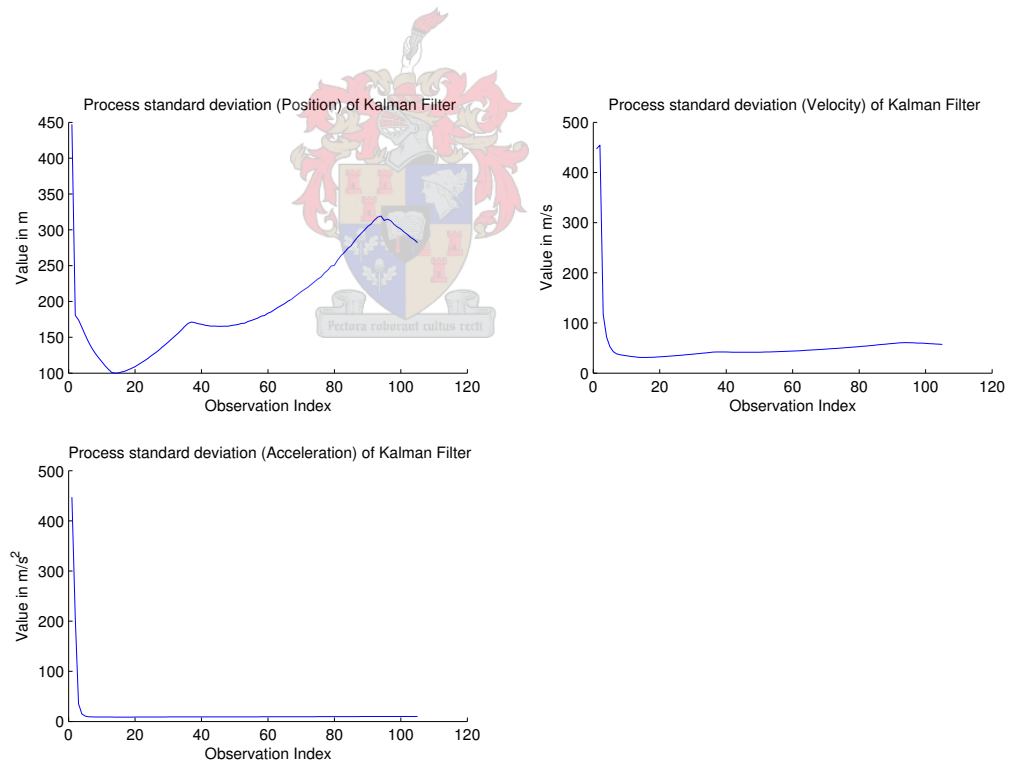


Figure 4.11: Process Covariance Adjustment

Illustrations of P_0 and Q selection

Two filter realisations are depicted in Figure 4.12 to illustrate the importance of selecting a very large initial process covariance. The first filtered trajectory is from a Kalman filter with a very small initial process covariance P_0 as shown here,

$$P_0 = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}.$$

The second filtered trajectory is from a Kalman filter with a very large initial process covariance P_0 as shown here,

$$P_0 = \begin{bmatrix} 10^5 & 0 & 0 & 0 \\ 0 & 10^5 & 0 & 0 \\ 0 & 0 & 10^5 & 0 \\ 0 & 0 & 0 & 10^5 \end{bmatrix}.$$

As illustrated in Figure 4.12, the filter with the small process covariance almost completely ignores the second observation and takes a while to “realise” its dependence on the observations because it has no prior knowledge of target velocity. The filter with the large process covariance relies heavily on the observations with the first observations but relaxes this reliance quickly as the first order derivatives develop. This is the better option as it has no knowledge of the target velocity during those first observations. The initial process covariances P_0 selected for this thesis were chosen to be large for better performance at the onset of the tracks.

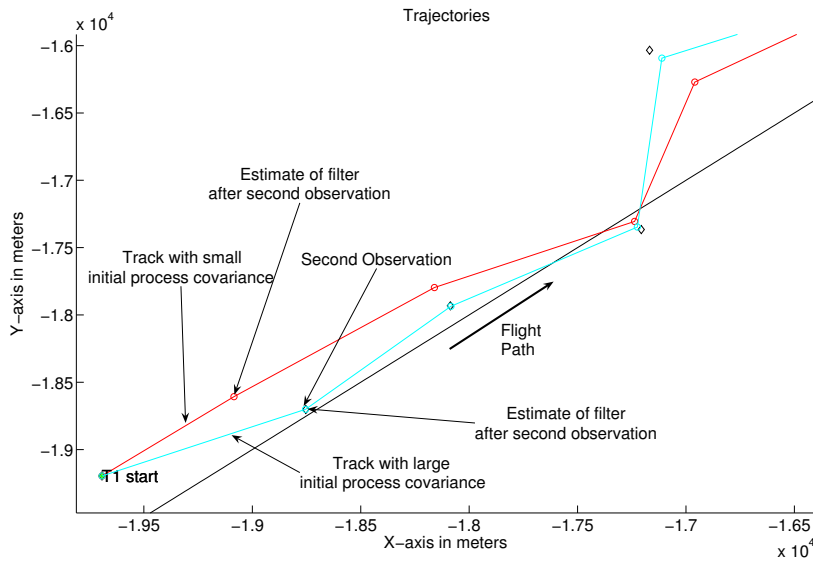


Figure 4.12: Effect of initial process covariance selection

The target model determines the process noise covariance Q . In the constant velocity model, a small process noise covariance is used. This leads to smoother filter output but has the danger of divergence if there happens to be acceleration from a manoeuvre because it is unable to handle changes in dynamics. The manoeuvring model determines a larger covariance that will lead to better target tracking on manoeuvres but will have a much less smoother filter output. Figure 4.13 illustrates how the filter with the larger process noise covariance was able to successfully follow the target through the manoeuvre and how the other filter with the smaller process covariance was not able to follow and diverged.

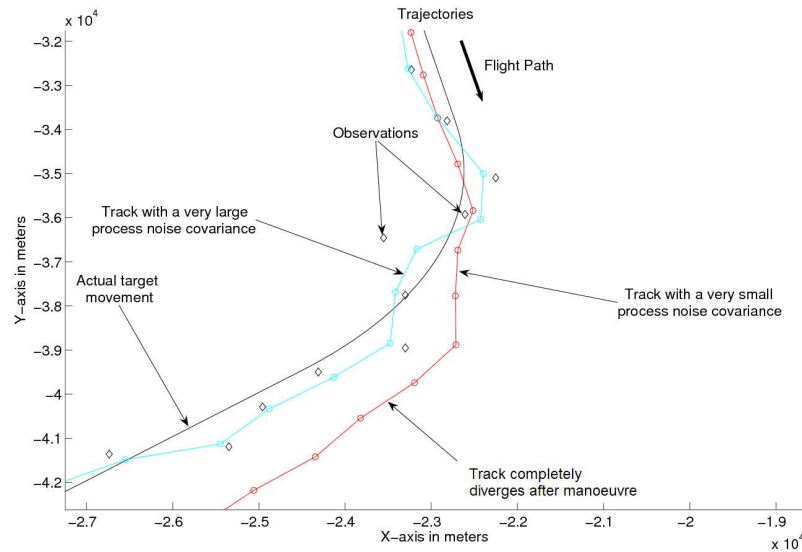


Figure 4.13: Effect of process covariance noise selection

4.3.3 Interactive Manoeuvre Model (IMM) Filter

Introduction

As the behaviour of the target changes, so should the model used to estimate the target movement [12]. Changing the parameters of the filter like the gain adjusting $\alpha\beta$ filter described in Section 4.3.1 is better, but not ideal, as it attempts to compensate for manoeuvres and does not attempt to model them. In Section 4.3.2, the Kalman filter used two models to perform optimal track filtering for manoeuvring and non-manoeuvring sections. Now the question is raised: Why not switch between these models as the behaviour of the target changes? Restarting the filter with a new model when a manoeuvre is detected is easy, but can manoeuvres be detected?

Manoeuvre detection is discussed in Section 4.4. For now, let's just assume that there is an external tracking process that monitors the target for manoeuvres and will inform the IMM filter of target behaviour change.

The 4-state Kalman filter with small process noise (described in Section 4.3.2) are used to filter sections of the target track that is relatively straight with no accelerations. The 6-state Kalman filter with large process noise is used to filter manoeuvring sections of the target track with large accelerations. At the onset of a manoeuvre the IMM switches to the 6-state filter and at the end of the manoeuvre it switches back to the 4-state filter, as illustrated in Figure 4.14. At every switch the selected filter process covariance is reset to initial values.

This combination provides high noise reduction on straight flight paths and good target tracking on manoeuvring sections, as illustrated in Figure 4.15.

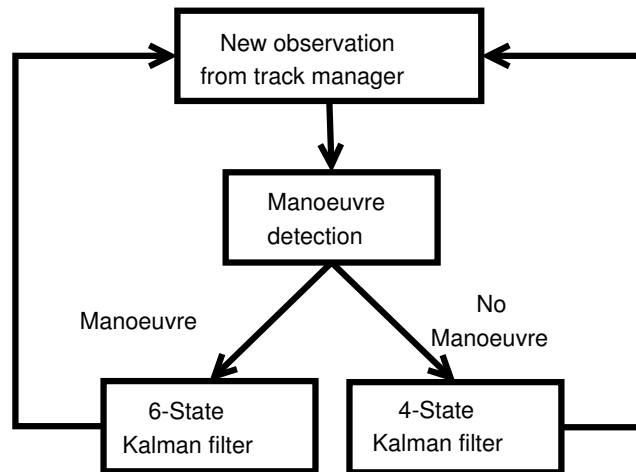


Figure 4.14: IMM filter construction

In Figure 4.15, the switching operation of the IMM filter is shown at the onset and again at the end of the manoeuvre. This allows a much better position estimation of the track in the manoeuvring and non-maneuvring segments.

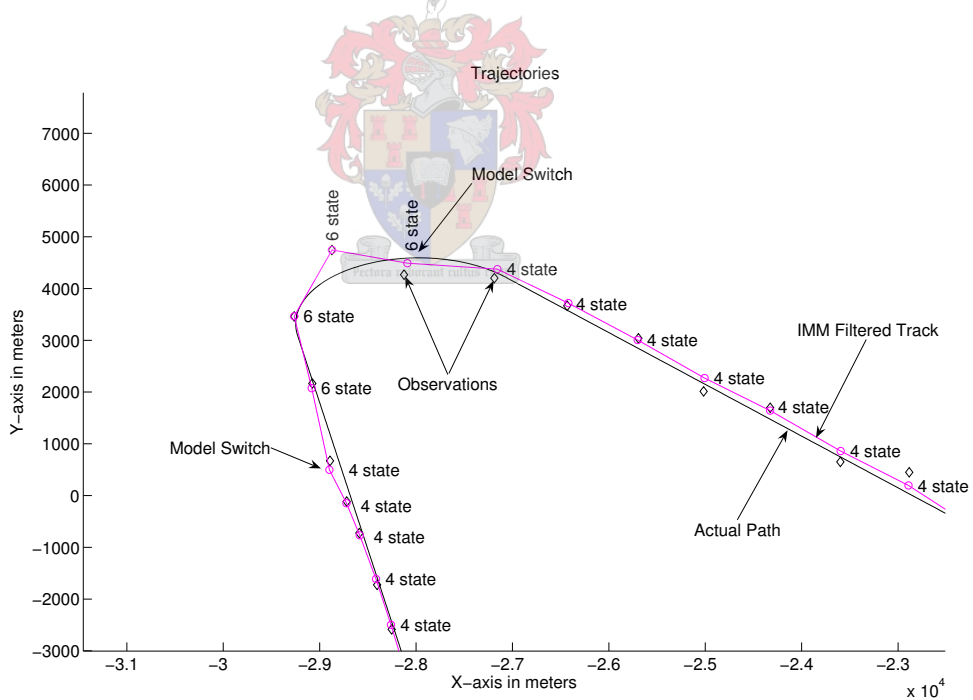


Figure 4.15: Illustration of IMM performance

4.3.4 Performance evaluation of filters

Filter configurations and simulation setup

Five filters are compared [4] and tabulated below in Table 4.3. All filters are designed and implemented to operate in 2-D although the target movement occurs in 3-D. Although the aircraft in reality will be flying in 3-D, the simulations are restricted to 2-D flight paths for simplicity because the radar systems used in the simulations cannot measure elevation. Please refer to Section 3.1 for a discussion on the effect of ignoring elevation in tracking. However, if the radar hardware allows for elevation measurements, the state vectors simply have to be increased accordingly to incorporate 3-D information in the tracking.

The first two filters use the $\alpha\beta$ algorithm. The only difference is that the first, identified as $\alpha\beta - 1$, uses the sequential parameter selection as described in Section 4.3.1. The second, identified as $\alpha\beta - 2$ uses the “jumpable” parameter selection method also described in Section 4.3.1.

The third, identified as Kalman-4, is a Kalman filter (described in Section 4.3.2) that uses the constant velocity 4-state model described in Section 4.2.1. The fourth, identified as Kalman-6, is also a Kalman filter with a 6-state model that uses the manoeuvring target model described in Section 4.2.2.

The fifth, identified as IMM, uses the Interactive Manoeuvre Model described in Section 4.3.3, which switches between the Kalman-4 and Kalman-6 filters.

Filter Identifier	Algorithm	State Vector Size
$\alpha\beta - 1$	$\alpha\beta$	4
$\alpha\beta - 2$	$\alpha\beta$	4
Kalman-4	Kalman	4
Kalman-6	Kalman	6
IMM	Kalman	alternates 4 and 6

Table 4.3: Filters setup

The first simulation consists of a Z flight path that the aircraft follows at a velocity of 250 m/s (900 km/h or Mach 0.735) and performs four manoeuvres with various levels of acceleration as indicated in Figure 4.16. The flight path is designed to include manoeuvring and straight flight sections that are close to and far from the radar. The second simulation as indicated in Figure 4.18 tested the noise reduction when the flight path was straight. The third simulation was on a single 4 G manoeuvre as illustrated in Figure 4.19 to test manoeuvre handling. These simulations were not tested on actual sensor data due to the costs involved. A Matlab implementation simulated the aircraft movement and the expected radar observations of the target given the noise covariances and hardware specifications of the Kameelperd radar system designed by Reutech Radar Systems.

Comparison Calculation

A baseline against which to evaluate the filter performances was selected. Specifically the position, speed and heading estimations of the filters were compared. Ultimately the noise in the observed position, speed and heading of the target at time step k had to be compared to the noise in the derived position, speed and heading estimates of the filter. The problem here was that there were no directly observable speed or heading measurements of the target from the radar for comparison. To solve this problem, the speed S_k and heading H_k “measurements” at observation k were calculated directly from the last two measured position observations Z_k and Z_{k-1} in the next two equations where y_k and x_k were the two 2-D components of the position and T was the time lapse between observations $k-1$ and k ,

$$S_k = \frac{Z_k - Z_{k-1}}{T}$$

$$H_k = \arctan\left(\frac{y_k - y_{k-1}}{x_k - x_{k-1}}\right).$$

Next, the error of the filter estimates were calculated as the absolute difference between the filter estimates and the true target state. The next equation calculated the noise in the filter estimates at observation k where P_e , S_e and H_e were the filter noise in the position, speed and heading estimates respectively. The position, speed and heading estimated from the filter are P_f , S_f and H_f respectively. The true position, speed and heading of the target are P_{true} , S_{true} and H_{true} respectively. The noise in the estimates was simply calculated to be the absolute differences between the filtered estimates and the true values,

$$\begin{bmatrix} P_e \\ S_e \\ H_e \end{bmatrix} = \begin{bmatrix} |P_f - P_{true}| \\ |S_f - S_{true}| \\ |H_f - H_{true}| \end{bmatrix}.$$

Next, the RMS values of the position, speed and heading noises of the filter were calculated for the whole simulation of K observations by calculating the root of the mean of the square of the noise values,

$$\begin{bmatrix} P_{rms} \\ S_{rms} \\ H_{rms} \end{bmatrix} = \begin{bmatrix} \sqrt{\frac{1}{K} \sum_{k=1}^K P_{k,e}^2} \\ \sqrt{\frac{1}{K} \sum_{k=1}^K S_{k,e}^2} \\ \sqrt{\frac{1}{K} \sum_{k=1}^K H_{k,e}^2} \end{bmatrix}.$$

The comparison between the filters was performed by running five parallel iterative (Track-While-Scan) architectures, each using a different filtering algorithm on the same set of realistic observations of the target. This process was repeated 10 000 times to generate a statistical RMS error means for the three simulations that are tabulated in Tables 4.4, 4.5 and 4.6.

The reduction percentile displayed in the tables refers to the increase or decrease of noise from using the particular filter as opposed to using no filter (or only observations) in the estimates. It is very possible for a filtering algorithm to return results worse than the unfiltered observations if the filter is not set up correctly. These reduction percentiles simplify the performance difference between the filter methods. The reduction percentile is calculated as follows, where Z_e is the error in the observed measurement and F_e is the error in the filtered estimate,

$$\text{Reduction Percentile} = \frac{Z_e - F_e}{Z_e}.$$

In order to appreciate the results of the position error in the filtered output, please refer to Section 4.5.3 for an illustration of the noise in the measured target position.

Estimation accuracy results for a flight path with manoeuvring and straight sections

Using the flight path setup as indicated in Figure 4.16 to test the filter, Figure 4.17 illustrates the performance of the filters at the first 3 G manoeuvre.

Overall, the 6-state Kalman and the IMM filter position error reductions were the highest due to their abilities to recover quickly from manoeuvres. The IMM filter scored a 25% position error reduction but the 4-state Kalman filter showed a 45% increase in position error noise due to its high reliance on a model that does not include the possibility of manoeuvres. The $\alpha\beta - 2$ filter does not have any position noise reduction in this simulation.

The velocity error V_e reductions and heading error H_e reductions of the target go hand in hand because the error velocity or heading estimate does not lead to better predictive results without the other. The $\alpha\beta - 1$ showed a 73% reduction in the velocity estimate error but had a 4% increase in the heading estimate error. This was due to the slow parameter adjusting of the $\alpha\beta - 1$ filter. The 4-state Kalman filter (Kalman-4) is in first place with a 75% noise reduction in velocity and a 4% noise reduction in the heading estimate. The IMM and the 6-state Kalman filters showed reductions but did not perform as well in the velocity and heading estimation. The reasoning is that the IMM and 6-state Kalman filters very quickly assume that velocity and heading noise point to an acceleration and model it as such, but the 4-state Kalman filter does not model acceleration and ignores it as noise. This advantage of the 4-state filter leads to a huge disadvantage in position-filtering. When reading the results in Table 4.5, it is important to note that the closest part of the flight path was 15 km from the radar where the expected observation noise

is the order of 200 metres and the furthest parts were 62 km from the radar where the expected observation noise is of the order of 750 metres.

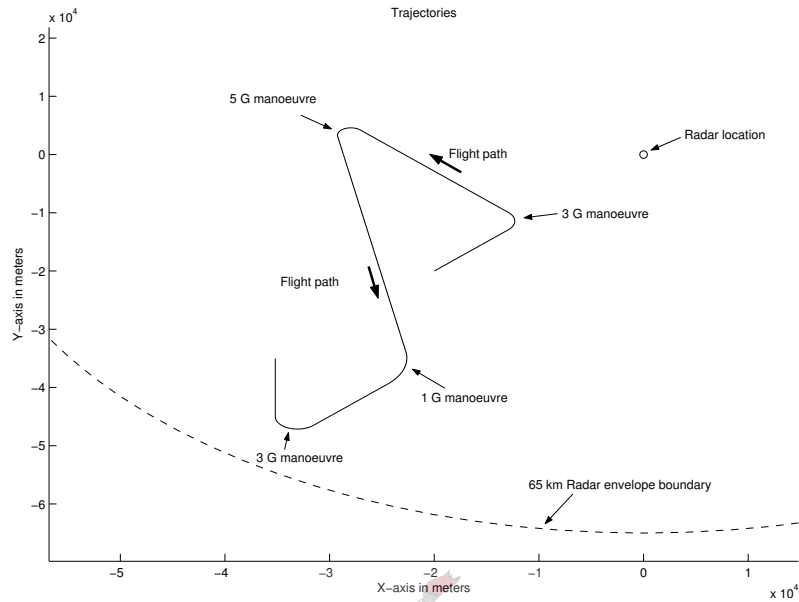


Figure 4.16: Z flight test simulation setup

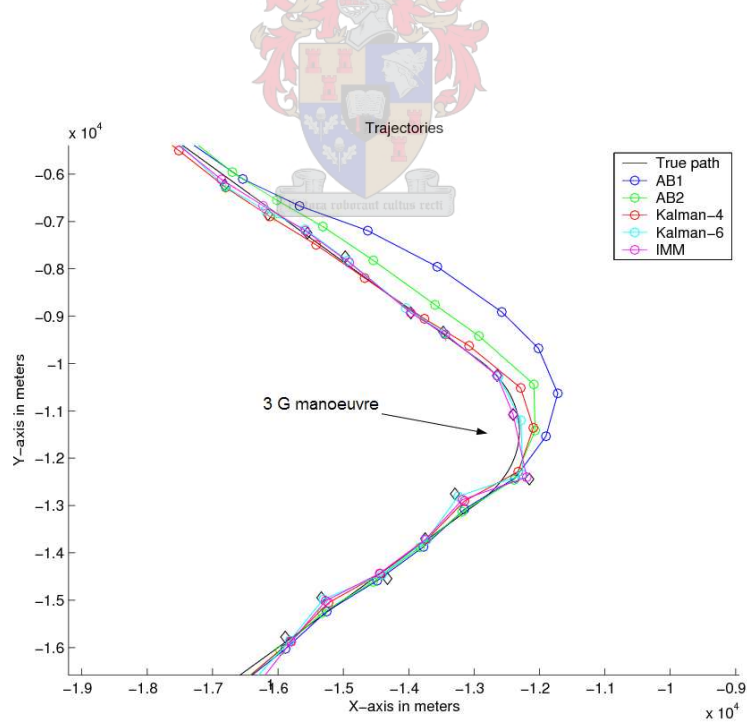


Figure 4.17: Z flight simulation close-up at the first manoeuvre

Filter	P_{rms} ; Reduction	S_{rms} ; Reduction	H_{rms} ; Reduction
$\alpha\beta - 1$	571.61 m (-17%)	29.81 m/s (73%)	2.71° (-4%)
$\alpha\beta - 2$	438.83 m (0%)	33.92 m/s (66%)	2.24° (-5%)
Kalman-4	699.55 m (-45%)	26.99 m/s (75%)	2.19° (4%)
Kalman-6	369.07 m (16%)	59.85 m/s (41%)	2.16° (-3%)
IMM	328.02 m (25%)	71.16 m/s (29%)	2.05° (2%)

Table 4.4: Filter results from Z flight simulation

Estimation accuracy for a straight flight path

In this simulation, the Kalman-4 filter was the best due to its model and process noise covariance selection which best fitted a straight flight path. The other filters also showed noise reduction but not as much. The IMM filter also performed very well because it was almost completely in the Kalman-4 mode for the duration of the simulation. In Table 4.5, it is important to note that the closest part of the flight path was 40 km from the radar where the expected observation noise is in the order of 500 metres and the furthest parts were 65 km from the radar, where the expected observation noise is about 800 metres.

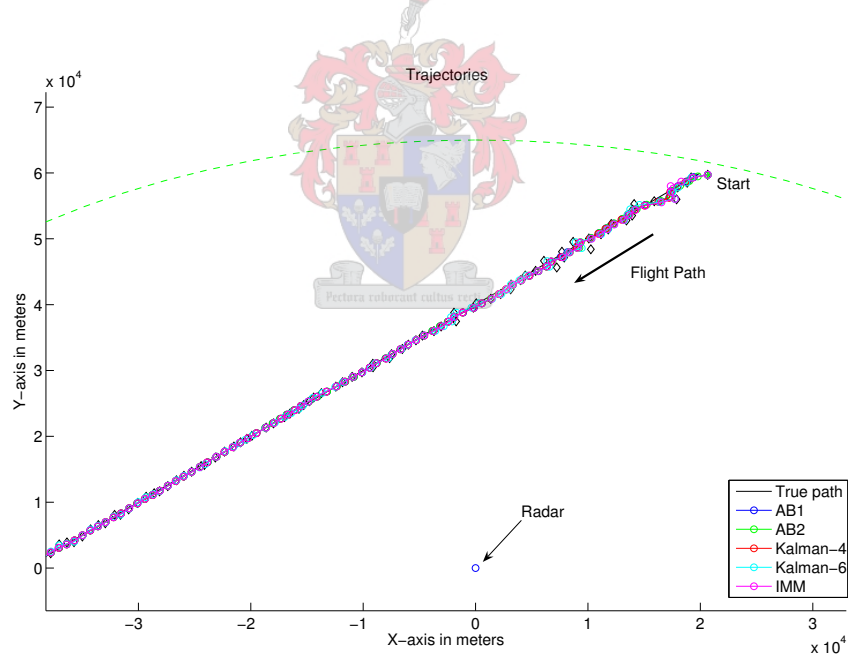


Figure 4.18: Straight flight simulation setup

Filter	P_{rms} ; Reduction	S_{rms} ; Reduction	H_{rms} ; Reduction
$\alpha\beta - 1$	274.45 m (47%)	15.55 m/s (87%)	6.25° (-5%)
$\alpha\beta - 2$	289.98 m (44%)	23.72 m/s (80%)	6.25° (-5%)
Kalman-4	265.72 m (49%)	11.65 m/s (90%)	6.25° (-5%)
Kalman-6	421.34 m (17%)	59.64 m/s (49%)	6.24° (-5%)
IMM	286.35 m (45%)	19.05 m/s (84%)	6.25° (-5%)

Table 4.5: Filter results for a straight flight path

Estimation accuracy for a 3 G manoeuvre

Like the Kalman-4 performance on straight paths, the Kalman-6 filter is designed for manoeuvres and is the only filter that show a reduction in position noise. It is interesting to note that the Kalman-4 filter showed the best velocity noise filtering. This can be attributed to the fact that airborne targets do not change speed as rapidly within manoeuvres as they change heading. The 4-state Kalman filter assumes constant velocity and therefore models the velocity the best. The IMM filter was in second place again because it was mostly in the 6-state Kalman filtering mode. When reading results Table 4.6, it is important to note that this simulation was performed 45 km from the radar where the expected observation noise is about 550 metres.

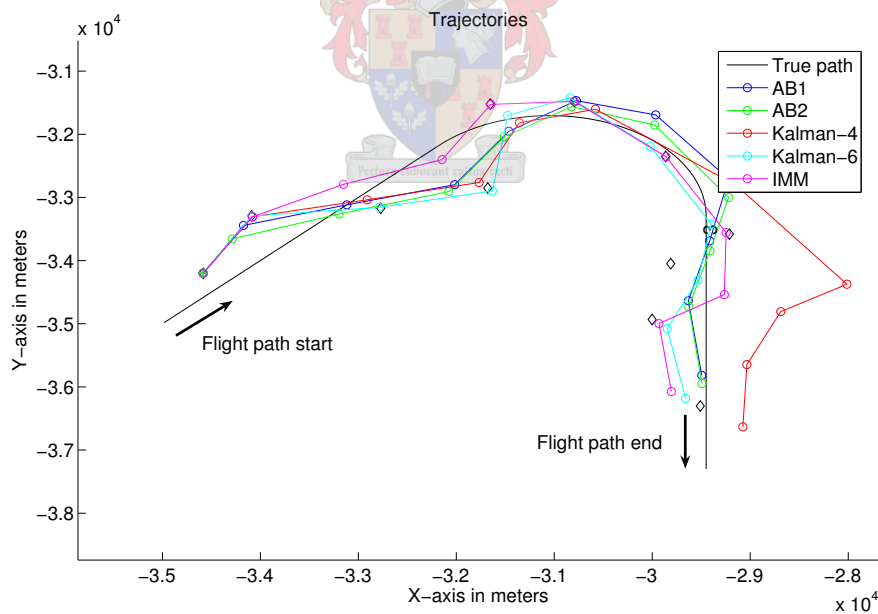


Figure 4.19: Manoeuvre simulation setup

Filter	P_{rms} ; Reduction	S_{rms} ; Reduction	H_{rms} ; Reduction
$\alpha\beta - 1$	501.54 m (-37%)	43.14 m/s (41%)	2.43° (10%)
$\alpha\beta - 2$	483.89 m (-8%)	65.20 m/s (29%)	3.02° (7%)
Kalman-4	687.00 m (-38%)	38.65 m/s (62%)	3.40° (5%)
Kalman-6	412.57 m (10%)	103.53 m/s (-11%)	3.24° (2%)
IMM	478.82 m (-1%)	133.65 m/s (-38%)	3.48° (-2%)

Table 4.6: Filter results for a manoeuvring flight path

Computational complexity and memory usage

The $\alpha\beta$ filter has been used extensively in industry for years because it is simple and very quick to compute. The Kalman filter on the other hand requires more processing. The primary computational bottleneck in the Kalman filter lies in the matrix inversion which is done to compute the Kalman gain factors, but, with processing resources that are available today, this is not a problem.

Both the Kalman and the $\alpha\beta$ filter implementations uses very little memory as no more than the previous state is stored. The Kalman filter also stores the process covariance of the previous state, but this simply is an additional 6×6 matrix. Therefore memory is also not an issue.

4.4 Manoeuvre Detection

4.4.1 Overview

Manoeuvre detection is necessary for the IMM filter described in Section 4.3.3 to select the appropriate model for filtering. However, there is nothing directly measurable with radar technology that will indicate a manoeuvre. The question is simply: How can the onset and end of a manoeuvres be detected?

Acceleration is the one distinctive feature of a manoeuvre. Although this cannot be measured with the radar, it can be calculated. The measurement noise is too great to calculate the acceleration directly from the observed position. Therefore the filtered position is used to derive this. This can also be obtained, without extra derivation, from a Kalman filter that models acceleration, as described in Sections 4.2.2 and 4.3.2.

Although Doppler provides only the component of the target velocity projected toward the radar, as described in Section 4.2.3, it can also be used to calculate a component of the acceleration. The advantage of Doppler is that it requires only one differentiation as opposed to using the second derivative of the filtered target position that requires two. The higher the order of differentiation, the more sensitive the calculation is to noise. But also bear in mind that Doppler is only one component of the target velocity and therefore does not contain the true velocity as illustrated in Section 4.2.3.

4.4.2 The second derivative of the filtered target position

The most distinctive feature of a manoeuvre is the acceleration. The 6-state model illustrated in Section 4.2.2, when used with the Kalman filter, automatically estimates the two-dimensional components of the acceleration in the fifth and sixth entry of the target state vector. Other filters like the $\alpha\beta$ filter requires the computation of the acceleration, but this is simply the second derivative of the position estimate.

Mathematically, given the time T_k and position P_k at observation k , the discrete data point velocity, using the central difference formula, at time instant k is,

$$P'_k = \frac{P_{k+1} - P_{k-1}}{T_{k+1} - T_{k-1}}.$$

The central difference formula, however, causes a computation lag of one observation because it requires one future and past observation to compute the current discrete derivative. Radar tracking system application demands real-time processing, and therefore the one-sided difference formula is preferable. Note the change in indexing compared to the central difference formula,

$$P'_k = \frac{P_k - P_{k-1}}{T_k - T_{k-1}}.$$

The acceleration or second derivative can then also be computed using the one-sided difference formula of the first derivative,

$$P''_k = \frac{P'_k - P'_{k-1}}{T_k - T_{k-1}}.$$

When expanded this becomes,

$$P''_k = \frac{\frac{P_k - P_{k-1}}{T_k - T_{k-1}} - \frac{P_{k-1} - P_{k-2}}{T_{k-1} - T_{k-2}}}{T_k - T_{k-1}}.$$

As a result, this method requires the current and two past observations to compute the second derivative. Another issue concerns the sensitivity of the second derivative to the large amounts of system noise. Although this method is theoretically sound, it is a bit too unstable to be trusted alone to detect manoeuvres as illustrated in Figure 4.20.

4.4.3 The first derivative of the observed Doppler information

Doppler information is much better for manoeuvre detection because it provides one component of the target velocity. Note that, because Doppler provides only one component of the target velocity, it cannot reliably be used to measure velocity as it is only a measurement of the range-rate: the rate at which the range of the target to the radar changes. The advantage of Doppler usage is that the sensitivity

to sensor noise is less because one less level of derivation is estimated. The Doppler derivative or acceleration of the range is calculated with the one-sided difference formula, as follows,

$$D'_k = \frac{D_k - D_{k-1}}{T_k - T_{k-1}}.$$

The Doppler measurement is a component of the velocity measurement that is projected on the range axis only, as illustrated in Section 4.2.3. Although these limitations prevent the direct estimation of the velocity with the Doppler measurement, the measurement can still be used to compute a pseudo-acceleration estimate for manoeuvre detection.

4.4.4 Comparison

For comparison, the simulation flight path setup illustrated in Figure 4.16 was used. Again, please note that the simulation was performed on a 2-D flight path (e.g. an aircraft that flew at a constant elevation) because the radar system is unable to measure elevation as explained in Section 3.1. As illustrated in Figure 4.20, the Doppler method performed much better in detecting the onset of the manoeuvre and had less noise when there was no manoeuvring, although it sometimes failed to keep on indicating higher levels of acceleration in the later stages of the manoeuvre. The second derivative of the filtered position took a bit longer to detect the onset of a manoeuvre and it had considerable amounts of noise because discrete second derivatives are ill-conditioned. The Doppler method was used in the IMM filter switching with an appropriate threshold setting.

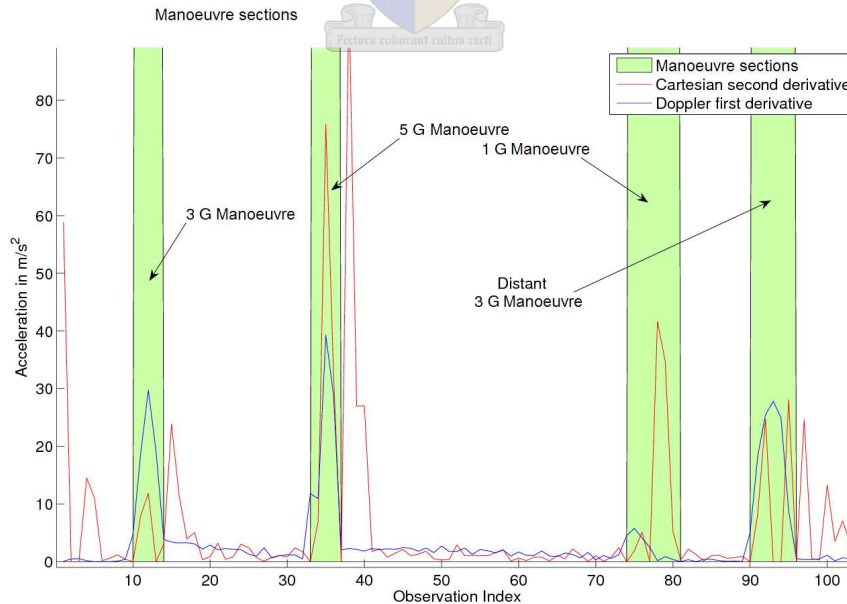


Figure 4.20: Results for manoeuvre detection

4.5 Observation Covariance calculation

4.5.1 Overview

One input of the Kalman filter is the observation noise covariance of the observing radar to compute the optimal target position estimate. The observation covariance of a radar is simply the observation noise variances of the range and azimuth detections that are defined in the polar coordinate system. However, because the modelling of the target is better in the Cartesian coordinate system, as discussed in Section 4.2.3, the Kalman filter operates in the Cartesian coordinate system. This requires translation of the noise variances to the Cartesian coordinate system, given the position of the target relative to the radar.

4.5.2 Algorithm

The easiest way to add the expected radar noise would be to translate the position of the target in the Cartesian coordinate system to the Polar coordinate system with the radar sensor as the centre. The position of the observation is denoted with $\begin{bmatrix} x_o & y_o \end{bmatrix}$ and the radar sensor position as $\begin{bmatrix} x_c & y_c \end{bmatrix}$,

$$\begin{bmatrix} R_o \\ \Theta_o \end{bmatrix} = \begin{bmatrix} \sqrt{(x_o - x_c)^2 + (y_o - y_c)^2} \\ \arctan\left(\frac{(y_o - y_c)}{(x_o - x_c)}\right) \end{bmatrix}.$$

Next, the five sigma points $\underline{\chi}_{0..4}$ are defined, one at the centre and four one standard deviation away from the centre on the principal axes of the covariance ellipse,

$$\underline{\chi}_0 = \begin{bmatrix} R_o \\ \Theta_o \end{bmatrix}, \underline{\chi}_1 = \begin{bmatrix} R_o + \sigma_R \\ \Theta_o \end{bmatrix}, \underline{\chi}_2 = \begin{bmatrix} R_o - \sigma_R \\ \Theta_o \end{bmatrix}$$

$$\underline{\chi}_3 = \begin{bmatrix} R_o \\ \Theta_o + \sigma_\Theta \end{bmatrix}, \underline{\chi}_4 = \begin{bmatrix} R_o \\ \Theta_o - \sigma_\Theta \end{bmatrix}.$$

Then translate these five sigma points back to the Cartesian Coordinate system,

$$\begin{bmatrix} x \\ y \end{bmatrix}_{\chi_n} = \begin{bmatrix} R \cos(\Theta) \\ R \sin(\Theta) \end{bmatrix}_{\chi_n}.$$

After the means \bar{E} of the sigma points are subtracted from the sigma points, the observation covariance R is estimated to be the covariance of these five translated sigma points,

$$\bar{E} = E \left[\begin{bmatrix} x \\ y \end{bmatrix}_{\chi_{0..5}} \right]$$

$$R = E \left[\left(\begin{bmatrix} x \\ y \end{bmatrix}_{\chi_{0..5}} - \bar{E} \right)^2 \right].$$

4.5.3 Illustration of observation noise

The Kameelperd radar system used in the thesis has a range variance σ_R of 18 m, an azimuth variance of 0.01222 radians (or 0.7 degrees) and a maximum detection range of 65 km as tabulated in Table A.2. Figure 4.21 illustrates the typical measured position noise of a radar with these noise characteristics.

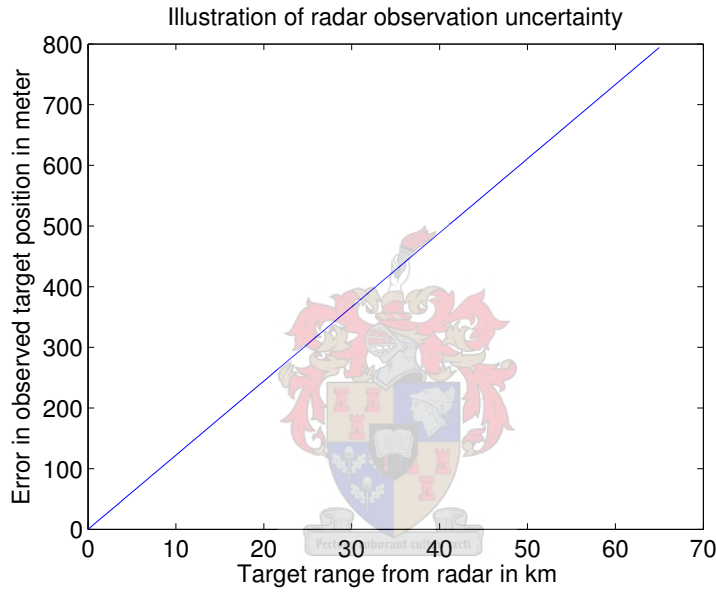


Figure 4.21: Illustration of typical radar observation noise

4.6 Conclusion

The Kalman filter implementations improves on the $\alpha\beta$ filters due to the ability to model the target movement and the statistical analysis of the measurement noise. The Cartesian coordinate system was also found to be better suited to the tracking application than the Polar system in which radars measure. The IMM filter, which uses the best of the 4- and 6-state features, performs best in the combined simulations, due to its ability to switch between target models. Although more computational power is required to compute the Kalman solution than the $\alpha\beta$ solution, it is not a notable issue today as the processing speed of computer systems have increased drastically. Therefore the IMM, which actually comprises of multiple linear Kalman filters, would be the chosen filter for this application.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis the integration of observations from different sensors was investigated. First the problem was addressed of correlating single observations from two different sensors. The problem was to decide whether the observations from different sensors belonged to the same target. The basic problem had to do with the uncertainty in the observations—radar observations are contaminated by noise, which limits the accuracy of the observations. For this problem a Monte-Carlo-type procedure was developed that illustrated the inherent limitations of the system due to noise. We showed that the resolution (the distance that two targets need to be apart in order to be observed as separate targets) depends on the position of the targets relative to the sensors. It is clear that a multi-sensor setup provides definite advantages.

The investigation mentioned above does not take any dynamics into account. In practise, the target flies along specific flight paths, even if changes in direction are expected. These flight paths, or radar tracks, provide additional information that should be exploited. To this end, two basic procedures were investigated, the alpha-beta filters and Kalman filters, as well as the IMM filter.

If a plane flies along a steady path, i.e. there are no sudden changes in direction, it is rather straightforward to track its motion since its behaviour is eminently predictable from past observations. For this type of motion, a simple Kalman filter with little system noise tracks the target without any problems. If, however, the target undergoes sudden changes, prediction and estimation of its motion is much harder. The approach followed for this thesis was to use two separate Kalman filters, one that was tuned for steady motion and another for modelling sudden changes. If sudden changes can be detected, it is possible to switch between the two filters. It was noted that the alpha-beta filters did not perform particularly well under these conditions. The basic problem was that they are not capable of modelling the uncertainties in the dynamics or observations.

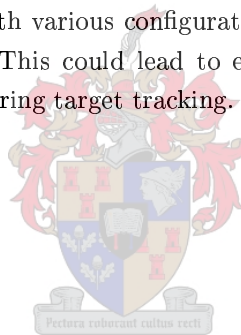
In order to detect sudden changes, target acceleration has to be calculated to detect manoeuvres. Using the second derivative of the measured target position

proved to be too unstable to be used as the acceleration estimate of the target. It was found that using the first derivative of the Doppler measurement was the best approach to this problem.

The Kalman filter implementations performed better than the $\alpha\beta$ filters due to the ability to model the target movement and its statistical analysis of the measurement noise. The IMM filter, which uses the best of the 4- and 6-state features, performed best in the combined simulations due to its ability to switch between target models. Although much more computational power is required to compute the Kalman solution than the $\alpha\beta$ solution, it is not a notable issue today as the processing speed of computer systems has increased drastically. Therefore the IMM, which actually consists of multiple linear Kalman filters, would be the chosen filter for this application.

5.2 Future Work

Some implementations of the IMM suggest the usage of additional specialised Kalman filters. This report only switched between two filtering modes, but there is no reason why multiple Kalman filters with various configurations could not switch depending on the target movement. This could lead to even better filtering results in manoeuvring and non-manoeuvring target tracking.



Bibliography

- [1] Anderson, Jon M., Nonlinear Suppression of Range Ambiguity in Pulse Doppler Radar, *Graduate School of Engineering and Management Air Force Institute of Technology Air University Air Education and Training Command.* (2001)
- [2] Bizup, David F. and Brown, Donald E., The Over-Extended Kalman Filter - Don't Use It!, *Systems and Information Engineering, University of Virginia, USA* (2002)
- [3] Bizup, David F. and Brown, Donald E., Manoeuvre Detection Using the Radar Range Rate Measurement, *Systems and Information Engineering, University of Virginia, USA.* (2002)
- [4] Bizup, David and Brown, Donald E., A Sufficient Comparison of Trackers. (2003)
- [5] Buddy, H Jean and Younker, John, A New Multi-Sensor Track Fusion Architecture for Multi-Sensor Information Integration, *Lockheed Martin Aeronautical Systems Company Marietta, Georgia.* (2003)
- [6] Dirk Tenne and Tarunraj Singh, The Higher Order Unscented Filter, *Centre for Multisource Information Fusion, The State University of New York at Buffalo.* (2001)
- [7] Ivo Ihrke, Some notes on ellipses, *Max-Planck-Institute for Information.* (2004)
- [8] Julier, Simon J and Uhlmann, Jeffery K, A new Extension of the Kalman Filter to Nonlinear Systems, *The Robotics Research Group, Department of Engineering Science, The University of Oxford.* (1997)
- [9] Julier, Simon J and Uhlmann, Jeffery K, Reduced Sigma Point Filters for the Propagation of Means and Covariances Through Nonlinear Transformations, *Dept. of Computer Engineering and Computer Science University of Missouri-Columbia.* (2003)
- [10] Julier, Simon J and Uhlmann, Jeffery K. A New Extension of the Kalman Filter to Nonlinear Systems, *The University of Oxford.* (1997)

- [11] Kalman, R. E, A New Approach to Linear Filtering and Prediction Problems, *Transactions of the ASME - Journal of Basic Engineering*, 82 (Series D): 35-45. Copyright ASME. (1960)
- [12] Karlsson, Erik, Decentralised tracking with feedback, adaptive sample rate and IMM, *Master's Thesis at the Division of Automatic Control at Linköping University (LiTH), Sweden.* (2000)
- [13] Lainiotis, D. G. and Papapaskevas, Paraskevas, Joint estimation and identification of lidar log power returns in a switching environment, *Optical Society of America.* (1996)
- [14] Mueller, K. Tysen and Robinson, John E. III, Final Approach Spacing Tool (FAST) Velocity Accuracy Performance Analysis. *American Institute of Aeronautics and Astronautics* (1998)
- [15] Morrell, D. R. and Stirling W. C., An Extended Set-valued Kalman Filter, *Arizona State University, USA.* (2003)
- [16] Moody, Leigh, Sensors, Sensor Measurement Fusion and Missile Trajectory Optimisation, *Cranfield University College of Defence Technology Department of Aerospace, Power and Sensors.* (2003)
- [17] Kelso, Thomas Sean, Ph.D, Temporal Clustering in the Multi-Target Tracking Environment, *The University of Texas at Austin.* (1988)
- [18] Kelly, Alonzo, A 3-D State Space Formulation of a Navigation Kalman Filter for Autonomous Vehicles, *The Robotics Institute Carnegie Mellon University 5000 Forbes Avenue Pittsburgh, PA 15213.* (1994)
- [19] Klett, Alfred, DeMey, Leo, Erath, Wolfgang and Nemecek, Peter, Calibration Of An Alpha-Beta Moving Filter Particulates Monitor, *Berthold Technology GmbH & Co. KG, Calmbacherstr. 22, 75323 Bad Wildbad, Germany.* (2002)
- [20] Singh, Tarunraj, Advanced Tracking Filters for Manoeuvring Targets in a Littoral Environment, *Centre for Multisource Information Fusion, The State University of New York at Buffalo.* (1999)
- [21] Tenne, Dirk and Singh, Tarunraj, Optimal Design of $\alpha - \beta - (\gamma)$ Filters, *Centre for Multisource Information Fusion State University of New York at Buffalo.* (2000)
- [22] Tenne, Dirk and Singh, Tarunraj, Circular Prediction Algorithms - Hybrid Filters, *Department of Mechanical & Aerospace Engineering, State University of New York at Buffalo.* (2002)
- [23] Tobias, Martin and Lanterman, Aaron D., Probability Hypothesis Density-Based Multi-target Tracking With Bi-static Range and Doppler Observations, *School of Electrical and Computer Engineering Georgia Institute of Technology Atlanta, Georgia U.S.A.* (2004)

- [29] Van Zandt, James R., Boost phase tracking with an unscented filter, *MITRE Corporation, MS-M210, 202 Burlington Road, Bedford MA 01730, USA*. (2003)
- [25] Van der Merwe, Rudolph and Wan, Eric, Gaussian Mixture Sigma-Point Particle Filters for Sequential Probabilistic inference in dynamic State-Space Models, *OGI School of Science & Engineering, Oregon Health & Science University*. (2003)
- [26] Van der Merwe, Rudolph and Wan, Eric, Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models, *OGI School of Science & Engineering Oregon Health & Science University*. (2001)
- [27] Van der Merwe, Rudolph and Wan, Eric, The Square-Root Unscented Kalman Filter for State and Parameter-estimation, *OGI School of Science & Engineering Oregon Health & Science University*. (2001)
- [28] Van der Merwe, Rudolph, Doucet, Arnaud, De Freitas, Nando and Wan, Eric, The Unscented Particle Filter, *Oregon Graduate Institute Electrical and Computer Engineering*. (2000)
- [29] Van Zandt, James R., Boost phase tracking with an unscented filter, *MITRE Corporation, MS-M210, 202 Burlington Road, Bedford MA 01730, USA*. (2003)
- [30] Weizhen Dong, Brian D. Jeffs and J. Richard Fisher, Kalman Tracking and Bayesian Detection for RADAR RFI Blanking, *Department of Electrical and Computer Engineering, Brigham Young University*. (2004)
- [31] Wan, Eric A., Van der Merwe, Rudolph and Nelson, Alex T., Dual Estimation and the Unscented Transformation, *Oregon Graduate Institute of Science & Technology Department of Electrical and Computer Engineering*. (2000)
- [32] Williams, Jason L. , Gaussian Mix Reduction for tracking multiple manoeuvring targets in clutter, *Department of the Air Force Air University, Wright-Patterson Air Force Base, Ohio*. (2003)
- [33] Zhansheng Duan, Chongzhao Han and X. Rong Li, Sequential Nonlinear Tracking Filter with Range-rate Measurements in Spherical Coordinates, *National Key Fundamental Research & Development Programs (973) of China*. (2003)

Appendix A

Radar sensor noise parameters

This thesis was funded to design and implement correlation and tracking algorithms for the commercially deployed Kameelperd and Page military radar systems. Both of these systems are designed and constructed by Reutech Radar Systems. The sensor parameters used in this thesis are tabulated in Tables A.1 and A.2.

<i>Page</i> Characteristics	Value
Azimuth deviation	0.00349 radians
Range deviation	12.5 metres
Elevation deviation	n/a
Doppler deviation	n/a
Maximum Range Envelope	15000 metres
Maximum Angle of Sight	0.34907 radians
Maximum Detection Height	5130 metres
Scan Rate / Rotation Rate	4 seconds

Table A.1: Page sensor parameters

<i>Kameelperd</i> Characteristics	Value
Azimuth deviation	0.01222 radians
Range deviation	18 metres
Elevation deviation	n/a
Doppler deviation	n/a
Maximum Range Envelope	65000 metres
Maximum Angle of Sight	0.38921 radians
Maximum Detection Height	8000 metres
Scan Rate / Rotation Rate	4 seconds

Table A.2: Kameelperd sensor parameters