12-2011

The Extended Preferred Ordering Theorem for Radar Tracking Using the Extended Kalman Filter

Donald Leskiw
Syracuse University

Follow this and additional works at: https://surface.syr.edu/eecs_etd
Part of the Electrical and Computer Engineering Commons

Recommended Citation
https://surface.syr.edu/eecs_etd/315

This Dissertation is brought to you for free and open access by the College of Engineering and Computer Science at SURFACE. It has been accepted for inclusion in Electrical Engineering and Computer Science - Dissertations by an authorized administrator of SURFACE. For more information, please contact surface@syr.edu.
Abstract

A certain problem in nonlinear estimation exists in radar tracking. Usually radar detections provide instantaneous position measurements in radar (polar) coordinates at discrete times, while tracks (estimated positions and motions over continuous time) are determined in rectangular coordinates; and the linear Kalman filter (LKF) is used as the estimator. Less common, the LKF is used to determine the tracks in radar coordinates, which are then converted into rectangular coordinates. Rarely is the extended Kalman filter (EKF) used, where the tracks are directly determined in rectangular coordinates from the radar detections via a local linearization. And so most radar tracks tend to be biased – and their Kalman covariance matrices are inconsistent with the true ones. Of course, some techniques have been proposed for “debiasing” them and making their mean squared errors “consistent” with the covariance matrices determined by the tracking filter. It is shown here, however, that the leading one for debiasing the LKF can make the biases worse; and a remedy for that is provided. But the focus is upon the EKF. In an earlier work by this author – dubbed the Preferred Ordering Theorem (POT) – it was shown that the linearization errors in range of the EKF can be virtually eliminated by using the measurement components of a detection recursively in a certain order: azimuth first and range last. But that has a fundamental limitation, namely, that “preferred” order. And so here a new version is provided, dubbed the Extended-POT (EPOT). Not only can the EPOT be more efficient than the POT in certain settings, but under it the measurements may be used in any order with virtually the same results.
THE EXTENDED PREFERRED ORDERING THEOREM
FOR RADAR TRACKING USING THE
EXTENDED KALMAN FILTER

by

Donald Myron Leskiw

B.S.E.E. Newark College of Engineering (1974)
M.S.E.E. New Jersey Institute of Technology (1977)

Dissertation
Submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Electrical Engineering of Syracuse University

December 2011
Acknowledgement

First, I want to express my sincere appreciation to Professor Hong Wang for serving as my advisor, especially for his encouragement and helpful guidance toward the completion of this dissertation. Appreciation is also due to Professors Carlos Hartmann, Philipp Kornreich, Tapan Sarkar, and Fritz Schlereth, for serving on the Examination Committee; and to Professor Lixin Shen for serving as its Chair. And I am very thankful for my wife, Lanjing Zhou, who also teaches at Syracuse University; also, my children Elanor and Michael, Lanjing’s son Eden Ma, and my sisters Mary Mag and Margie – they all encouraged me to complete this work.

And I wish to acknowledge the following individuals who helped me over the years toward this goal. Charles R. Pederson, my first “boss,” who at Riverside Research Institute (RRI) lead me away from electromagnetics and radar cross-section modeling to Kalman filtering. The Air Force’s Richard R. Paul provided me with extraordinary tracking problems; and Carl Kukkonen and David W. Curkendall of Caltech’s Jet Propulsion Laboratory provided me with the means for their solution – also Geoffrey C. Fox and Thomas D. Gottschalk at Caltech. And Jurgen Moser, who at the Courant Institute inspired me: he showed me the interplay among the “geometric, algebraic, and topological points-of-view,” which may be overheard here. Finally, I am deeply grateful to Kenneth S. Miller, formerly Professor of Mathematics at Columbia University and New York University: my friend and colleague at RRI, we co-authored a book on Kalman filtering, and several research papers on tracking. From Ken I learned applied mathematics, and the importance of notation. But, most of all, his friendship and encouragement over the years have been vital to me personally.
# Table of Contents

1 **INTRODUCTION** .................................................................................................................. 1

1.1 **BACKGROUND ON THE PROBLEM** ................................................................................. 2

1.2 **SUMMARY AND ORGANIZATION OF THIS DISSERTATION** ............................................. 4

1.3 **A RADAR SYSTEM CONTEXT FOR THE TRACKING PROBLEM** ................................. 6

2 **MATHEMATICAL PRELIMINARIES** .................................................................................... 9

2.1 **THE PRINCIPAL SPACES** ............................................................................................... 9

   2.1.1 *The Principal Differentials and Jacobian Matrices* ....................................................... 12

   2.1.2 *The Basic Random Vectors* .......................................................................................... 14

2.2 **DEFINITIONS OF DETECTIONS AND TRACKS** ............................................................ 17

   2.2.1 *Representations of Detections* ....................................................................................... 17

   2.2.2 *Representations of Tracks* ............................................................................................ 19

2.3 **THE BASIC TRACKING EQUATIONS** .............................................................................. 23

   2.3.1 *The Prediction Step* ..................................................................................................... 24

   2.3.2 *The Update Step* ......................................................................................................... 25

2.4 **A SPECIAL CASE** .......................................................................................................... 27

2.5 **THE BASIC ESTIMATION CASES AND NOTATION** ..................................................... 29

3 **ILLUSTRATION OF THE PROBLEM** ............................................................................... 32

3.1 **THE BASIC ESTIMATION BIAS PROBLEM** .................................................................. 32

   3.1.1 *The Scalar-Weight Case* .............................................................................................. 34

   3.1.2 *The Matrix-Weight Case* ............................................................................................ 37

3.2 **THE EXTENDED KALMAN FILTER** .............................................................................. 40

3.3 **DISCUSSION OF THE BASIC ESTIMATION PROBLEM** .............................................. 42

4 **ANALYSIS OF THE PLKF UPDATE** ............................................................................... 47

4.1 **THE PSEUDO-MEASUREMENT ERRORS** ................................................................. 47

   4.1.1 *The Expected Value of the Pseudo-Measurements* ................................................... 48
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>THE SCALAR-WEIGHT CONSISTENT EPOT</td>
<td>101</td>
</tr>
<tr>
<td>7.2</td>
<td>THE MATRIX-WEIGHT CONSISTENT EPOT CASE</td>
<td>109</td>
</tr>
<tr>
<td>7.3</td>
<td>EXTENSION TO HIGHER-ORDER MODELS OF MOTION</td>
<td>112</td>
</tr>
<tr>
<td>7.4</td>
<td>THE ABOLISHMENT OF THE “PREFERRED ORDERING” BY THE EPOT</td>
<td>112</td>
</tr>
<tr>
<td>8</td>
<td>CONCLUDING REMARKS</td>
<td>115</td>
</tr>
<tr>
<td>8.1</td>
<td>SUMMARY OF THIS RESEARCH</td>
<td>115</td>
</tr>
<tr>
<td>8.2</td>
<td>PROPOSED ACTIVITIES FOR FOLLOW-ON RESEARCH</td>
<td>117</td>
</tr>
<tr>
<td>8.2.1</td>
<td>The Effect of Observation Geometry Non-Stationarity</td>
<td>117</td>
</tr>
<tr>
<td>8.2.2</td>
<td>Evaluation of the PLKF and EKF in the Maximum Likelihood Sense</td>
<td>119</td>
</tr>
<tr>
<td>8.3</td>
<td>CONTRIBUTIONS OF THIS WORK</td>
<td>121</td>
</tr>
<tr>
<td></td>
<td>APPENDIX</td>
<td>122</td>
</tr>
<tr>
<td></td>
<td>REFERENCES</td>
<td>127</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The basic tracking system</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>The basic estimation bias problem</td>
<td>33</td>
</tr>
<tr>
<td>3</td>
<td>The basic estimation paths</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>The basic rectangular estimation sequences</td>
<td>36</td>
</tr>
<tr>
<td>5</td>
<td>The basic radar estimation sequences</td>
<td>37</td>
</tr>
<tr>
<td>6</td>
<td>The scalar- and matrix-weight cases (rectangular coordinates)</td>
<td>39</td>
</tr>
<tr>
<td>7</td>
<td>The scalar- and matrix-weight cases (radar coordinates)</td>
<td>39</td>
</tr>
<tr>
<td>8</td>
<td>PLKF and EKF estimates of $(x, y)$</td>
<td>41</td>
</tr>
<tr>
<td>9</td>
<td>CB-PLKF and CB-EKF estimates of $(r, a)$</td>
<td>41</td>
</tr>
<tr>
<td>10</td>
<td>PLKF and EKF rectangular estimates $(\sigma_A = \pi/12)$</td>
<td>42</td>
</tr>
<tr>
<td>11</td>
<td>CB-PLKF and CB-EKF radar estimates $(\sigma_A = \pi/12)$</td>
<td>43</td>
</tr>
<tr>
<td>12</td>
<td>Comparison of the sample means (rectangular cases)</td>
<td>44</td>
</tr>
<tr>
<td>13</td>
<td>Comparison of the sample means (radar cases)</td>
<td>44</td>
</tr>
<tr>
<td>14</td>
<td>Normalized sample standard deviations (rectangular cases)</td>
<td>45</td>
</tr>
<tr>
<td>15</td>
<td>Normalized sample standard deviations (radar cases)</td>
<td>45</td>
</tr>
<tr>
<td>16</td>
<td>“Debiased” scalar-weight PLKF estimates</td>
<td>56</td>
</tr>
<tr>
<td>17</td>
<td>Accuracy of the “debiased” scalar-weight PLKF estimates</td>
<td>56</td>
</tr>
<tr>
<td>18</td>
<td>“Debiased” matrix-weight PLKF biases</td>
<td>58</td>
</tr>
<tr>
<td>19</td>
<td>Matrix-weight UCCM with “consistent” covariance matrices</td>
<td>59</td>
</tr>
<tr>
<td>20</td>
<td>Accuracies of the “debiased” PLKF’s (scalar-weight cases)</td>
<td>60</td>
</tr>
<tr>
<td>21</td>
<td>Comparison of the “prior-weight” MB-PLKF with the basic scalar-weight PLKF</td>
<td>64</td>
</tr>
<tr>
<td>22</td>
<td>Accuracies of prior-matrix weight and scalar-weight PLKF’s</td>
<td>64</td>
</tr>
<tr>
<td>23</td>
<td>Effectiveness of the POT in rectangular coordinates</td>
<td>75</td>
</tr>
<tr>
<td>24</td>
<td>Effectiveness of the CB-POT (converted back into radar coordinates)</td>
<td>75</td>
</tr>
<tr>
<td>25</td>
<td>Effectiveness of the EPOT in rectangular coordinates</td>
<td>77</td>
</tr>
<tr>
<td>26</td>
<td>Effectiveness of the CB-EPOT (converted back into radar coordinates)</td>
<td>77</td>
</tr>
</tbody>
</table>
Figure 55 Comparison between the AR-EPOT and RA-EPOT (inconsistent case) 114
Figure 56 EPOT estimates for air and missile defense scenario ($\sigma_A = 1.5$) 117
Figure 57 M-PLKF, POT, and EPOT root-mean-squared errors ($\sigma_A = 1.5$) 118
Figure 58 M-PLKF, POT, and EPOT root-mean-squared errors ($\sigma_A = 2.5$) 118
Figure 59 Sample distributions of measurements and estimates 120

List of Tables

Table 1 Possible Motions of Certain Classes of Objects 6
1 Introduction

A certain problem in nonlinear estimation exists in radar tracking. Usually radar detections provide instantaneous position measurements in radar (polar) coordinates at discrete times, while the tracks of those objects (estimated positions and motions over continuous time) are determined in rectangular coordinates. And most often an estimator like the linear Kalman filter (LKF) is used. Less common, the tracks are determined in radar coordinates, and then converted into rectangular coordinates. Rarely is the so-called extended Kalman filter (EKF) employed, since it determines the rectangular tracks directly from the radar detections by using local linear approximations. Indeed:

“Most … simply transform the polar coordinates to cartesian coordinates directly which modifies the noise process slightly. In this manner, the bias errors which build in the funny curvilinear coordinates is avoided and the filter gains are only suboptimal by trivial amounts due to the transformation. … this is the normal practice [emphasis added]… .”

(anonymous IEEE reviewer of [1]).

And so most radar tracks tend to have unknown biases – and their Kalman covariance matrices are inconsistent with the true ones.

Over the years techniques for “debiasing” such LKF tracks have been proposed. But it is shown below that the leading ones can make those biases worse – a remedy for that shall be provided. The focus here, however, shall be upon the EKF. In an earlier work, now called the Preferred Ordering Theorem (POT), it was shown that the EKF’s linearization errors in range can be reduced significantly by using the measurement components of a detection recursively in a certain order: azimuth first and range last. But it shall also be shown below that the POT is not very effective at shorter ranges. And so
a new version will be provided, dubbed the **Extended-POT (EPOT)**. Not only is the EPOT more efficient than the POT at shorter ranges, but under it the scalar measurement components of a radar detection can be used recursively in any order to update an EKF track, with virtually the same results.

### 1.1 Background on the Problem

A Kalman filter is a linear estimator with a quadratic cost function [2-4]. (The popular alpha-beta radar tracking filter may be considered to be a steady-state Kalman filter [5, 6].) The Kalman estimation equations are essentially those of the Weighted Least-Squares estimator; and they provide the Best Linear Unbiased Estimate when the true inverse covariance matrices are used as the weights. If all the models are linear and all the errors have zero means and gaussian distributions, the BLUE provides the Gauss-Markov estimate, which is optimal in the unbiased and minimum variance sense [7].

Unfortunately, most radar tracking problems are nonlinear and nongaussian. First, radar measurements are most naturally determined in radar coordinates (range is based upon time-delay measurements; and azimuth is based upon the directivity of the antenna, and beamforming [8]). And, second, the gaussian assumption is generally not valid. Radar measurement errors are often said to be gaussian, but range is always positive and azimuth is limited to its principal values. Indeed, in radar tracking the gaussian assumption is weakly motivated by the Central Limit Theorem; it is strongly motivated by the tractability it gives to an analysis [9].

Of course, nowadays bone fide nonlinear estimators are available. For example, particle filters provide “statistically linearized” estimates that can be optimal in the limit [10, 11]. But to be effective they require a large number of independent detections (or simulated ones). And they tend to be “computationally intensive” [12]. Real-time radar
tracking systems generally avoid computationally intensive methods [13-15]. And so most rectangular tracks are biased [16, 17], and their covariance matrices determined by the tracking filter are inconsistent with the true ones [18].

Techniques for mitigating the LKF estimation bias have been proposed. For example, the Unscented Kalman Filter (UKF) models certain parameters of the measurements’ probability distribution, transforms those, and determines them within a Kalman filtering framework [19]. Others say that the transformed measurements should be “debiased” before being used. For example, the Debiased Consistent Converted Measurements (DCCM) method subtracts the expected value of the estimation errors before using the transformed measurements [20, 21]; and the Unbiased Consistent Converted Measurements (UCCM) method dilates the transformed measurements [22, 23]. Unfortunately, in order to work, those methods require some knowledge of the underlying probability distribution of the measurements, which is usually unknown in practice. And so research continues in this area [24-27].

Before that, Schmidt (1962) suggested that the models be linearized recursively, which became known as the extended Kalman filter (EKF) [28]; and Jazwinski (1970) proposed that the linearization errors be modeled using the so-called system noise [29] – for others see pp. 5-9 of [30]. But the EKF linearization errors tend to be systematic: they accumulate in the estimates, which can lead the track to diverge from its object. A “second-order” EKF may be used, or an “iterated” EKF, but they require significantly more computation than the (first-order) EKF [55].

Fortunately, a certain noncommutativity exists when the components of a radar measurement are used recursively to update an EKF track in rectangular coordinates.
And there is a preferred ordering: azimuth first and range last [31]. Here that method is called the Preferred Ordering Theorem (POT). But the POT is counterintuitive to a common belief that the most accurate measurement should be used first, so as to obtain a better linearization (see p. 166, [32]) – such usually implies the worst order, since range measurements are generally much more accurate than angle ones (in cross-range).

The original motivation of the POT was a Best Estimate of Trajectory (BET) problem for a certain precision reentry vehicle. Since then it has been used in real time tracking applications [33], and in non-realtime settings like the Lincoln Orbit Determination (LODE) program of MIT Lincoln Laboratory [31]. Others have combined it with the DCCM/ UCCM [34, 35], and with the UKF [36].

1.2 Summary and Organization of this Dissertation

The key findings of this research were published in [38] and [61], namely,


This Dissertation is divided into eight Chapters, which are summarized here, plus an Appendix (and all references are provided after the Appendix). To expedite the presentation, the problem is restricted to the Euclidean plane. And, to make the bias and consistency issues more pronounced, the system noise is zero – for the most part, the “system model” is exact (just the update equations are of concern here).
Chapter 1 provides this introduction. And a radar systems context for the tracking problem is also provided (following this section).

Chapter 2 specifies the basic notation and presents the basic estimation equations. And the various trackers to be used in the sequel are summarized.

Chapter 3 illustrates the radar tracking biases by using a simple example that Julier and Uhlmann employed to motivate their Unscented Kalman Filter (UKF) [37]. And there the detailed equations of the various tracking filters are provided.

Chapter 4 presents a new analysis of the LKF biases. Also, the situation where the popular “debiasing” methods make the biases worse is identified – and a remedy is given. These results are illustrated using the Julier and Uhlmann “exemplar.”

Chapter 5 provides a new analysis of the EKF linearization errors; and the EKF with the POT is illustrated. But the POT is seen to be not effective at short ranges. And so the Basic-EPOT (B-EPOT) is then defined and illustrated [38]. And it is seen to be more efficient than the POT (and much better than the “debiased” PLKF’s). The Julier and Uhlmann “exemplar” is also used in this Chapter.

Chapter 6 illustrates the results of the previous two Chapters by using a more stressing tracking problem, one that Bar Shalom has used in the DCCM/UCCM literature over the years [20, 26]. Except for the EKF with the POT, all the trackers are now seen to have problems – and the B-EPOT is much less effective.

Chapter 7 analyzes the Kalman updates further; and one more extension to the POT is given, dubbed the position-velocity consistency constraint. This EPOT is then demonstrated using the DCCM/UCCM “exemplar.” Not only is it as effective as the POT, it is also seen to abolish the preferred ordering. That is, under the
EPOT, the EKF may now use the measurement components of a detection in any order to update the track recursively, with virtually the same results.

Chapter 8 provides concluding remarks; and some areas for future work are identified. An itemized summary of the contributions of this work is also provided.

1.3 A Radar System Context for the Tracking Problem

To provide a “systems” context for this work, a notional radar concept for wide-area surveillance and tracking is described here – for others see [39].

Let the primary missions of the radar be: detect certain objects-of-interest as they enter the (instantaneous) field-of-view (FOV) of its antenna, and issue alarms for the new ones; and determine the positions and motions of those objects while they are within the FOV, and report them periodically. The radar’s antenna is assumed to rotate in azimuth at a fixed rate – a complete $2\pi$ rotation is called a scan (the antenna may also scan electronically over a limited instantaneous FOV as it rotates). Some assumptions on the possible motions of the objects-of-interest are also needed – see, for example, Table 1.

<table>
<thead>
<tr>
<th>Table 1 Possible Motions of Certain Classes of Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>TARGET</td>
</tr>
<tr>
<td>SHIPS</td>
</tr>
<tr>
<td>MILITARY AIRCRAFT</td>
</tr>
<tr>
<td>MISSILES</td>
</tr>
<tr>
<td>HELICOPTERS</td>
</tr>
<tr>
<td>CIVIL AIRCRAFT</td>
</tr>
</tbody>
</table>

Reference:
A. Farina and F. A. Studer, Radar Data Processing Volume I - Introduction and Tracking, Research Studies Press, LTD., Letchworth England 1985 (Table 3.2)

The radar is assumed to operate by periodically transmitting a pulsed signal in certain directions, determined by the orientation of the antenna, and beamforming. And each
transmission defines a beam, a cone shaped volume determined by the beamwidth of the antenna. The interval between successive transmissions is called a dwell. During each dwell the radar attempts to detect the returned reflections of the transmitted signal from objects that are in or near the beam, and measures their respective positions.

Surveillance beams are nominally scheduled according to some pre-determined search template; and surveillance tracks are determined from the detections in those beams, scan-to-scan. Detections from overlapping beams that are believed to be associated with the same object may be first merged together before being used, to provide just one merged detection per object per scan, as well as better measurements. Given a surveillance track, however, the radar may decide that a more accurate track is needed. In which case an active track is determined by using dedicated beams that are centered at predicted positions of the object. Such beams can provide more accurate measurements, and provide them at a higher rate. Also, active tracks may be updated beam-to-beam, using the detections individually, not merged ones.

A top-level view of the tracking system is provided below in Figure 1. First, given a new beam’s worth of detections, any tracks that might have been detected are determined, called the candidates for that beam’s detections. Those tracks are then correlated (a process) with the beam’s detections; and the ones that are believed to be associated (a decision) with the same objects are determined. Tracks that have new associated detections are then updated (say, once a scan for surveillance tracks, and beam-to-beam for active tracks). Scan-based correlation may determine whether a certain subset of detections should become a new track; and whether an existing track should be dropped. It may also determine whether a given track should be split (bifurcated), or whether two
or more tracks should be merged together. Finally, for a surveillance track, its likely
detection time in the next scan is determined, and the track is predicted there. And for an
active track, its next set of update beams in the next scan are determined (and those
beams are requested).

It should be noted, however, that a radar track generally has two phases: \textit{track initialization} and \textit{track maintenance}; and that initialization usually includes \textit{verification} – the
object’s existence is an assumption. Indeed, in practice a radar tracking algorithm
generally consists of a predictor-corrector loop (such as that illustrated by Figure 1), plus
an initialization step (not illustrated above). In this work, the emphasis is upon the
spurious estimation errors, which exist in both track initialization and track maintenance.
The unique characteristics of both phases shall be seen in the illustrations that are
provided in the sequel.
2 Mathematical Preliminaries

The purpose of this Chapter is to define the notation that shall be used in the sequel, and to present the basic radar tracking equations. Whereas, the focus of the analysis will be upon the spurious linear estimation errors that are caused by the nonlinear transformation between radar and rectangular coordinates, “coordinate-less” Euclidean space is used to define the basic problem first, followed by its radar and rectangular coordinate representations. The notation is summarized in the conclusion of the Chapter; and there the various trackers that shall be used in the sequel are also outlined.

2.1 The Principal Spaces

Recall that Euclidean space, denoted \( \mathbb{E} \), is homogeneous and isotropic, with no point or direction inherently distinguished, but its points and sets of points may be chosen freely. And there the common notions of distance and angle are invoked axiomatically; and translation, rotation, reflection isometries are also defined [40-43]. Accordingly, the radar is said to be a certain fixed point, \( O \in \mathbb{E} \). And the object to be tracked is another point, \( P \), which is arbitrary except for \( P \neq O \). Denote the directed line segment in \( \mathbb{E} \) from \( O \) to \( P \) by \( OP \). Write the distance between \( O \) and \( P \) as \( \|OP\| \) (a unit length is tacitly chosen in \( \mathbb{E} \)); and represent the orientation and positive sense of \( OP \) by an abstract unit vector, \( e_{op} \) (such is defined more formally at the end of this section).

Now choose a fixed rectangular coordinate frame in \( \mathbb{E} \), denoted by \( [O;(e_x,e_y)] \), where \( O \) specifies the origin, and \( (e_x,e_y) \) is an ordered pair of orthonormal (abstract) unit vectors that represent the orientation and positive sense of the coordinate axes. Under \( [O;(e_x,e_y)] \) the rectangular coordinates of the radar and object are respectively
(0, 0) and (x, y). And, given (x, y), with (x, y) ≠ (0, 0), the *radar coordinates* of the object, *range* and *azimuth*, are defined to be

\[ r = (x^2 + y^2)^{1/2} \quad \text{and} \quad a \equiv \arctan(y, x) \]  

(1)

(the four-quadrant inverse tangent function is being used here, \(-\pi < a \leq +\pi\), not the two

quadrant one, \(\phi \equiv \arctan(\sin \phi / \cos \phi), \ -\pi/2 < \phi \leq +\pi/2\). In which case, \(r = \|OP\|\) and

\[ \mathbf{e}_{op} = e_x \cos a + e_y \sin a \]. And given \((r, a)\) as the radar coordinates of \(P \in \mathbb{E}\), the corre-

sponding rectangular coordinates are

\[ x = r \cos a \quad \text{and} \quad y = r \sin a \]. \hspace{1cm} (2)

Finally, \(r\) and \(a\) in (2) may have any values, and so the phrase “the *principal values* of

\((r,a)\)” shall be used to indicate \(r > 0\) and \(-\pi < a \leq +\pi\).

Now for arbitrary radar and rectangular coordinates, say \((r, a)\) and \((x, y)\), not

necessarily related by (1) and (2), let them respectively correspond to the points \(R\) and

\(X\) in \(\mathbb{E}\). Either \(X = R\) or \(X \neq R\). If \(X = R\), then usually \((r, a) \neq (x, y)\); and if

\((r, a) = (x, y)\), then usually \(X \neq R\). Of course, when \((x, y)\) and \((r, a)\) are related by (1)

and (2), then \(X = R\). But a special case exists: \(X = R \iff (r, a) = (x, y)\) if and only if

\(x = r \geq 0\) and \(y = a = 0\). This special case shall be used in the sequel to illustrate the

effects of the nonlinear coordinate transformations upon the estimators.

Column vectors of radar and rectangular coordinates shall also be used to represent

points in \(\mathbb{E}\), namely,

\[ \mathbf{\rho} = \begin{bmatrix} r \\ a \end{bmatrix} \quad \text{and} \quad \mathbf{\xi} = \begin{bmatrix} x \\ y \end{bmatrix} \]. \hspace{1cm} (3)

Such are members of distinct vector spaces where the standard basis is used, say \(\mathbf{\rho} \in \mathbb{R}\)
and $\xi \in \mathbf{X}$. Their transposes will be written as $\rho^T = (r, a)$ and $\xi^T = (x, y)$, with a space after the comma to distinguish them from their corresponding coordinate points. For convenience, when the components of $\rho$ and $\xi$ in (3) are related by (1) and (2), those vectors shall be said to be functions of one another, $\rho = h(\xi)$ and $\xi = h^{-1}(\rho)$. Of course, $h^{-1}$ is not generally the inverse of $h$ per se; only if the radar coordinates are restricted to their principal values are $h$ and $h^{-1}$ inverses of one another.

Finally, the (abstract) unit vectors $e_{op}, e_x, e_y$, used above to represent the orientation and positive sense of directed line segments in $\mathbb{E}$, are not members of $\mathbf{X}$ or $\mathbb{R}$. Rather, they are members of an affine space that is isomorphic to $\mathbb{E}$, written $(\mathbb{E}; X)$ [44], simply denoted by $\mathbf{A}$. Objects in $\mathbf{A}$ shall always be denoted using lowercase bold-italic symbols. And the symbol “$e$” shall always denote unit vectors there. The null vector in $\mathbf{A}$ and the column vector of zeros in $\mathbf{X}$ and $\mathbb{R}$ shall all be denoted by $0$. Note that $\mathbb{E}$, $\mathbf{X}$, $\mathbf{A}$ are isomorphic to one another, while $\mathbb{R}$ is not isomorphic to any of them.

For example, given $\xi$ and $\rho$ in (3), the corresponding objects in $\mathbf{A}$ are written $\xi$ and $\rho$. When $(r, a)$ and $(x, y)$ are related by (1) and (2), then

$$
\xi \equiv xe_x + ye_y = (x, y) \begin{bmatrix} e_x \\ e_y \end{bmatrix} \quad \text{and} \quad \rho \equiv re_x(a) + 0e_y(a) = (r, 0) \begin{bmatrix} e_x(a) \\ e_y(a) \end{bmatrix},
$$

(4)

with

$$
\begin{bmatrix} e_x(a) \\ e_y(a) \end{bmatrix} = \begin{bmatrix} \cos a & \sin a \\ -\sin a & \cos a \end{bmatrix} \begin{bmatrix} e_x \\ e_y \end{bmatrix},
$$

(5)

(the arguments of $e_x(a)$ and $e_y(a)$ will be dropped when the context allows).

Denote the matrix in (5) by $O(a)$. It represents a (positive) rotation in $\mathbb{E}$ about the origin, $O$, by the azimuth angle, $a$. Under $O(a)$ the $[O; (e_x, e_y)]$ frame is rotated onto
another rectangular coordinate frame, \( [O;(e_r,e_a)] \). Note that \( O(a) \) is orthonormal,

\[
O^{-1}(a) = O^T(a), \text{ and that } O^{-1}(a) = O(-a).
\]

Given \((x,y)\) and \((r,a)\), respectively the rectangular and radar coordinates of \( X \) and \( R \) in \( E \), when \( X = R \) then \( \xi = \rho \), while \( \xi = \rho \) need not be true. Indeed,

\[
\xi = (x, y) \begin{bmatrix} e_x \\ e_y \end{bmatrix} = (x, y)O^T(a)O(a) \begin{bmatrix} e_x \\ e_y \end{bmatrix} = (r, 0) \begin{bmatrix} e_r \\ e_a \end{bmatrix} = \rho .
\]

And \( \|\xi\| = \|\rho\| = r = \|OP\| \), and \( e_x = e_z \cos a + e_y \sin a = e_{op} \).

### 2.1.1 The Principal Differentials and Jacobian Matrices

Now given \( \rho \in R \) and \( \xi \in X \) as defined above, when \( \rho = h(\xi) \) and \( \xi = h^{-1}(\rho) \) their differentials are related as [45]

\[
\begin{bmatrix}
dr(x,y) \\
da(x,y)
\end{bmatrix} = \begin{bmatrix}
\frac{\partial}{\partial x} r(x,y) & \frac{\partial}{\partial y} r(x,y) \\
\frac{\partial}{\partial x} a(x,y) & \frac{\partial}{\partial y} a(x,y)
\end{bmatrix}
\begin{bmatrix}
dx \\
dy
\end{bmatrix} \tag{7}
\]

and

\[
\begin{bmatrix}
dx(r,a) \\
dy(r,a)
\end{bmatrix} = \begin{bmatrix}
\frac{\partial}{\partial r} x(r,a) & \frac{\partial}{\partial a} x(r,a) \\
\frac{\partial}{\partial r} y(r,a) & \frac{\partial}{\partial a} y(r,a)
\end{bmatrix}
\begin{bmatrix}
dr \\
da
\end{bmatrix} . \tag{8}
\]

In the sequel, the above two expressions shall also be written as

\[
d\rho = J(\xi)d\xi \text{ and } d\xi = J^{-1}(\rho)d\rho , \tag{9}
\]

and the matrices definitized as

\[
J(\xi) = \frac{dh(\xi)}{d\xi} \text{ and } J^{-1}(\rho) = \frac{dh^{-1}(\rho)}{d\rho} . \tag{10}
\]
The expressions in (9) are linear maps between $\mathbf{R}$ and $\mathbf{X}$, which have a nonlinear dependency upon a parameter, respectively $\xi$ and $\rho$. The matrices defined by (10) may also be viewed as matrix-valued point functions, written $\mathbf{J}(\mathbf{X})$ and $\mathbf{J}^{-1}(\mathbf{R})$. And when $\mathbf{X} = \mathbf{R} = \mathbf{P}$, $\mathbf{P} \neq \mathbf{O}$, then $\mathbf{I} = \mathbf{J}(\mathbf{X})\mathbf{J}^{-1}(\mathbf{R}) = \mathbf{J}^{-1}(\mathbf{R})\mathbf{J}(\mathbf{X})$.

Of course, $\mathbf{J}$ is the Jacobian matrix of the transformation defined by (1),

$$
\mathbf{J}(\xi) \equiv \mathbf{J}(x, y) = \begin{bmatrix}
\frac{x}{\sqrt{x^2 + y^2}} + \frac{y}{\sqrt{x^2 + y^2}} \\
-\frac{y}{\sqrt{x^2 + y^2}} - \frac{x}{\sqrt{x^2 + y^2}}
\end{bmatrix},
$$

and $\mathbf{J}^{-1}$ is the Jacobian matrix defined by (2),

$$
\mathbf{J}^{-1}(\rho) = \mathbf{J}^{-1}(r, a) = \begin{bmatrix}
\cos a & -r \sin a \\
+\sin a & r \cos a
\end{bmatrix}.
$$

And when $\rho = \mathbf{h}(\xi)$ and $\xi = \mathbf{h}^{-1}(\rho)$, then $\mathbf{J}$ and $\mathbf{J}^{-1}$ may also be written as

$$
\mathbf{J}(\rho) = \mathbf{J}(r, a) = \begin{bmatrix}
\cos a & +\sin a \\
-(\sin a)/r & (\cos a)/r
\end{bmatrix}
$$

and

$$
\mathbf{J}^{-1}(\xi) \equiv \mathbf{J}^{-1}(x, y) = \begin{bmatrix}
\frac{x}{\sqrt{x^2 + y^2}} & -\frac{y}{\sqrt{x^2 + y^2}} \\
+\frac{y}{\sqrt{x^2 + y^2}} & \frac{x}{\sqrt{x^2 + y^2}}
\end{bmatrix}.
$$

And hybrid parameterizations are also defined,

$$
\mathbf{J}(x, y; r) = \begin{bmatrix}
\frac{x}{r} & +\frac{y}{r} \\
-\frac{y}{r} & \frac{x}{r}
\end{bmatrix} \text{ and } \mathbf{J}^{-1}(a; x, y) = \begin{bmatrix}
\cos a & -y \\
\sin a & +x
\end{bmatrix}.
$$

Note that the arguments of $\mathbf{J}$ and $\mathbf{J}^{-1}$ serve two purposes at once: the symbol specifies the parameterization (principal, alternative, or hybrid); and its value specifies the point at which the matrix is being determined. And when $\rho$ and $\xi$ both represent $\mathbf{P}$, then (using the polymorphism of arguments defined above),
\[ I = J(p)J^{-1}(p) = J(\xi)J^{-1}(\xi) = J(p)J^{-1}(\xi). \]  

Finally, \( J \) and \( J^{-1} \) may be factored into products of rotation matrices and metric tensors, written

\[ J(\xi) = D^{-1}(r)O(a) \quad \text{and} \quad J^{-1}(r) = O^T(a)D(r), \]

with

\[ D(r) = \begin{bmatrix} 1 & 0 \\ 0 & r \end{bmatrix} \quad \text{and} \quad O(a) = \begin{bmatrix} \cos a & + \sin a \\ -\sin a & \cos a \end{bmatrix}. \]

Note that \( D^{-1}(r) = D(1/r) \), \( r \neq 0 \), and \( O^T(a) = O^T(-a) = O(-a) \). Also,

\[ \|dx\|^2 = dx^T dx = dr^T D(r)O(a)O^T(a)D(r)dr = dr^T D^2(r)dr, \]

which implies

\[ \|dx\|^2 = dx^2 + dy^2 = dr^2 + r^2 da^2. \]

### 2.1.2 The Basic Random Vectors

In the sequel, column vectors such as \( \xi \) and \( p \) will be realizations of *random vectors*, denoted \( X \) and \( R \) [46] – the transposes of \( X \) and \( R \) shall be written \( X^T = (X, Y) \) and \( R^T = (R, A) \) – in this context the symbols \( X \), \( R \), etc. denote scalar random variables, not Euclidean points (the basic notation will be summarized at the end of this Chapter).

The expected values of \( X \) and \( R \) are written

\[ \mathcal{E}X = \mu_X = \begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix} \quad \text{and} \quad \mathcal{E}R = \mu_R = \begin{bmatrix} \mu_R \\ \mu_A \end{bmatrix}, \]

with \( \mathcal{E} \) denoting the expectation operator. And the covariance matrix of \( X \) is

\[ \text{cov}(X) \equiv \mathcal{E}(X - \mu_X)(X - \mu_X)^T = \Sigma_X = \begin{bmatrix} \sigma^2_X & \sigma_{XY} \\ \sigma_{YX} & \sigma^2_Y \end{bmatrix}, \]
with $\sigma_{XY} = \sigma_{YX}$. When $\Sigma_X$ is positive definite, $\sigma_X^2 > 0$ and $\sigma_{X}^2 \sigma_{Y}^2 - \sigma_{XY}^2 > 0$. Similarly,

$$\text{cov}(R) \equiv \mathcal{E}(R - \mu_R)(R - \mu_R)^T = \Sigma_R = \begin{bmatrix} \sigma_R^2 & \sigma_{RA} \\ \sigma_{RA} & \sigma_A^2 \end{bmatrix}.$$ 

For convenience, a given mean vector and its covariance matrix will be written together, as $(\mu_X; \Sigma_X) \in (X; X^2)$, where $\mu_X \in X$ and $\Sigma_X \in X^2$. And when $X$ and $R$ have gaussian distributions, that shall be indicated by writing $X \sim \mathcal{N}(\mu_X; \Sigma_X)$ and $R \sim \mathcal{N}(\mu_R; \Sigma_R)$. For example, if $X \sim \mathcal{N}(\mu_X; \Sigma_X)$, the density function of $X$ is

$$p_X(\xi) = \frac{1}{2\pi(\det\Sigma_X)^{1/2}} \exp \left[-\frac{1}{2}(\xi - \mu_X)^T \Sigma_X^{-1} (\xi - \mu_X)\right]. \quad (21)$$

Now using the functions $h$ and $h^{-1}$ as defined in the previous section, the random vectors $X$ and $R$ may be transformed as $R' = h(X)$ and $X' = h^{-1}(R)$. And given $\mu_X \in X$ and $\mu_R \in R$, one can also determine $h(\mu_X)$ and $h^{-1}(\mu_R)$. But usually $\mu_R \neq h(\mu_X)$ and $\mu_X \neq h^{-1}(\mu_R)$, because $h$ and $h^{-1}$ are nonlinear while $\mathcal{E}$ is a linear operator.

Similarly, the covariance matrices of $X$ and $R$ may be transformed as

$$R'(\xi) = J(\xi) \Sigma_X J^T(\xi) \quad \text{and} \quad \Xi'(\rho) = J^{-1}(\rho) \Sigma_R J^{-T}(\rho) \quad (22)$$

($J^{-T}$ is the transpose of $J^{-1}$). But usually $R' \neq \Sigma_{R'}$ and $\Xi' \neq \Sigma_{X'}$ – the transformations in (22) simply represent changes of bases. Of course, under a linear transformation, say $Y = HX$, where $H$ is a constant matrix,

$$\mu_Y = \mathcal{E}Y = \mathcal{E}HX = H\mu_X \quad \text{and} \quad \Sigma_Y = \mathcal{E}(Y - \mu_Y)(Y - \mu_Y)^T = H\Sigma_X H^T. \quad (23)$$

And, in that case, if $X \sim \mathcal{N}(\mu_X; \Sigma_X)$, then $Y \sim \mathcal{N}(H\mu_X; H\Sigma_X H^T)$ [47].
In the sequel, pairs such as \((\mu_X; \Sigma_X)\) and \((\mu_R; \Sigma_R)\) shall be transformed using \(h\) and 

\(h^{-1}\) together with the expressions in (22). In particular, two operators are defined, \((h; J)\) and \((h^{-1}; J^{-1})\), and their respective actions are written succinctly as

\[
(h; J) : (\mu_X; \Sigma_X) \mapsto (\mu'; R') \quad \text{and} \quad (h^{-1}; J^{-1}) : (\mu_R; \Sigma_R) \mapsto (\xi'; \Xi').
\]  

(24)

Note that in (24) the arguments of \(h\) and \(h^{-1}\) are tacitly used as the arguments of \(J\) and 

\(J^{-1}\). But, under the polymorphism of arguments defined in the previous section for \(J\) and 

\(J^{-1}\), instead of (22), one may also write

\[
R'(\rho) = J(\rho)\Sigma_X J^T(\rho) \quad \text{and} \quad \Xi'(\xi) = J^{-1}(\xi)\Sigma_R J^{-T}(\xi).
\]  

(25)

However, when \(\xi\) and \(\rho\) represent the same point, say \(P \in \mathbb{E}, P \neq O\), then \(J(\rho) = J(\xi)\) and 

\(J^{-1}(\xi) = J^{-1}(\rho)\). In which case \(R'(\xi) = R'(\rho)\) and \(\Xi'(\rho) = \Xi'(\xi)\).

Finally, in the sequel, to distinguish realizations of \(X\) and \(R\) from their deterministic counterparts, measurements of \(\xi\) and \(\rho\) shall be adorned as \(\bar{\xi}\) and \(\bar{\rho}\), and estimates adorned as \(\hat{\xi}\) and \(\hat{\rho}\) (similarly, scalar measurements and estimates will be written \(\bar{x}\) and 

\(\hat{x}\), etc.) And measurements and estimates will written with their covariance matrices as 

\((\bar{\rho}; \Sigma_R)\) and \((\hat{\xi}; \Sigma_X)\). However, in radar tracking the underlying random vectors of given measurements and estimates are usually unknown, and surrogates for the true covariance matrices are used instead. Here such surrogates shall be called associated covariance matrices, denoted \(R\) and \(\Xi\) (and \(X\)); and pairs such as \((\bar{\rho}; R)\) or \((\hat{\xi}; \Xi)\) will be transformed using \((h; J)\) and \((h^{-1}; J^{-1})\), as in (24).
2.2 Definitions of Detections and Tracks

Here a radar detection is assumed to provide an instantaneous measurement on the radar coordinates of some point in $\mathbb{E}$, say $\vec{R}$; and a given detection is assumed to have a certain weight, denoted $w$, which is based on the signal-to-noise power ratio of the received signal. Also, here a track is defined to be a sequence of such detections at distinct times that are believed to be associated with some object. A track that is a single unassociated detection, a singlet set, is an initial-hypothetical track (if a subsequent detection becomes associated with it, the existence of its object is said to be verified). Given its sequence of detections, a track is determined recursively by using a filter. A model for the possible motion of the object is chosen; and its next detection is predicted. And, if a subsequent detection becomes associated with that prediction, the track is updated. But the association, prediction, and updating operations are determined using either the radar or rectangular coordinate representations (or both). And the more recent detections and the more accurate measurements are given greater weight. Given detections of the same object, different coordinate representations can lead to different tracks. In the next two subsections, the coordinate representations of detections and tracks that shall be used in the sequel are given.

2.2.1 Representations of Detections

Here a detection is defined to be a weighted point, denoted $(\vec{R}; w)$, where $\vec{R} \in \mathbb{E}$ is determined by $(\vec{r}, \vec{a})$, and $w > 0$. In $\mathbb{A}$ such is written $(\vec{\rho}; w)$. If the measurements are transformed into rectangular coordinates, using (2), then $\vec{X}$ denotes the point in $\mathbb{E}$, and $\vec{\xi}$ is used in $\mathbb{A}$. Of course, $\vec{X} = \vec{R}$ and $\vec{\xi} = \vec{\rho}$, since (2) is exact. And so $(\vec{R}; w) = (\vec{X}; w)$ and $(\vec{\rho}; w) = (\vec{\xi}; w)$ — also, $w \vec{\rho} = w \vec{\xi}$ in $\mathbb{A}$. 
Now using the column vector representations of $R$ and $X$, the basic coordinate forms of a detection are $(\bar{p}; I/w) \in (R; R^2)$ and $(\bar{\xi}; I/w) \in (X; X^2)$, where $\bar{p}^T = (\bar{r}, \bar{a})$ and $\bar{\xi} = h^{-1}(\bar{p})$ – here an inverse weight matrix is being used, $I/w$. The detection may also be represented by $w\bar{p} \in R$ and $w\bar{\xi} \in X$. But, recall, when $\bar{X} = \bar{R}$, then usually $\bar{p} \neq \bar{\xi}$ – indeed, when $(\bar{R}; w) = (\bar{X}; w)$, then $w\bar{p} \neq w\bar{\xi}$ (Pr = 1).

Range and azimuth measurements, however, usually have disparate accuracies (such are respectively functions of the bandwidth and beamwidth of the radar system [39]). And so, instead of a scalar-weight, a weight-matrix may be better, say

$$W = \begin{bmatrix} w_r & 0 \\ 0 & w_a \end{bmatrix},$$

with $w_r$ and $w_a$ both positive. In which case, letting, $R = W^{-1}$, the radar detection is written $(\bar{p}; R)$. In rectangular coordinates such is written $(\bar{\xi}; \Xi)$, where $\bar{\xi} = h^{-1}(\bar{p})$ and $\Xi = J^{-1}(\bar{p})RJ^{-1}(\bar{p})$.

Note that

$$R^{-1}\bar{p} = \begin{bmatrix} w_r \bar{r} \\ w_a \bar{a} \end{bmatrix} \quad \text{and} \quad \Xi^{-1}\bar{\xi} = \begin{bmatrix} w_r \bar{X} \\ w_a \bar{Y} \end{bmatrix}$$

(see the sequel for details). That is, not only does $R^{-1}\bar{p} \neq \Xi^{-1}\bar{\xi}$ (Pr = 1), but the effective weight of the transformed detection in rectangular coordinates is inconsistent with the given weight of the detection (Pr = 1).

Now if $\bar{p}$ is a realization of a known random vector, say $R$, and the covariance matrix of $R$, denoted $\Sigma_R$, is also known, then the detection may also be written as
\((\tilde{\rho}; \Sigma_R)\). Here \(R = \rho + \tilde{R}\), where \(\rho^T = (r, a)\) is the true position of the object. Usually \(\tilde{R} \sim \mathcal{N}(0; \Sigma_R)\) is assumed, with

\[
\Sigma_R = \begin{bmatrix}
\sigma^2_r & 0 \\
0 & \sigma^2_a
\end{bmatrix},
\tag{28}
\]

If \(\Sigma_R\) is unknown, the detection is written \((\tilde{\rho}; \mathbf{R})\), with \(\mathbf{R}\) some symmetric and positive definite associated covariance matrix. In either case, given \((\tilde{\rho}; \Sigma_R)\) or \((\tilde{\rho}; \mathbf{R})\), the corresponding representation in rectangular coordinates is determined using the second expression in (24), either as

\[
(h^{-1}; J^{-1}) : (\tilde{\rho}; \Sigma_R) \mapsto (\tilde{\xi}; \Xi) \quad \text{or} \quad (h^{-1}; J^{-1}) : (\tilde{\rho}; \mathbf{R}) \mapsto (\tilde{\xi}; \Xi)
\tag{29}
\]

(the context will specify whether \(\Sigma_R\) or \(\mathbf{R}\) is the pre-image). In the sequel, primes will sometimes be used on these outcomes, as \((\tilde{\xi}'; \Xi')\), to emphasize that \(\tilde{\xi}'\) is a pseudo-measurement, a realization of some \(X'\), and that \(\Xi'\) is not the true covariance matrix of \(X'\).

### 2.2.2 Representations of Tracks

The basic representations of tracks are similar to those defined above for detections: for example, \((\hat{x}; X)\), an estimated vector and its associated covariance (or inverse-weight) matrix. But for a track to be valid, some assumptions on the possible motion of the object are also needed. Briefly, the unknown motion of an object is assumed to define a curve in \(\mathbb{E}\), a one parameter continuous set of points, \(\{P(t) : t_{\min} \leq t \leq t_{\max}\}\), with \(t_{\min} < t_{\max}\) (the independent variable is time). And the curve is assumed to be sufficiently smooth (see below). It is said to be degenerate if \(P(t) = P(\tau)\), for all \(\tau\) and \(t\) in \([t_{\min}, t_{\max}]\); and if \(P(t) = P(\tau) \Rightarrow t = \tau\), then the curve is said to be simple [48].
More formally, an inertial rectangular coordinate frame of reference is invoked [49], namely, \([O; (e_x, e_y)]\). The instantaneous (rectangular) position and velocity vectors with respect to that frame are then \(\xi\) and \(\dot{\xi}\). And the instantaneous (inertial) acceleration of the object is \(a \equiv \ddot{\xi}\), which is assumed to be piecewise continuous – \(a\) is also assumed to be continuous to the right, and continuous at \(t_{\text{max}}\).

Under the assumptions given above for the possible motion of the object, when \(a\) is independent of \(\xi\) and \(\dot{\xi}\), the equation of the actual motion is [50]

\[
\frac{d}{dt} \begin{bmatrix}
\xi \\
\dot{\xi}
\end{bmatrix} = \begin{bmatrix}
0 & 1 \\
0 & 0
\end{bmatrix} \begin{bmatrix}
\xi \\
\dot{\xi}
\end{bmatrix} + \begin{bmatrix}
a
\end{bmatrix},
\]

(30)

And so, letting \(\xi_0 = \xi(t_0)\) and \(\dot{\xi}_0 = \dot{\xi}(t_0)\) at \(t_0\), the position and motion of the object at \(t\) are [51]

\[
\begin{bmatrix}
\xi(t) \\
\dot{\xi}(t)
\end{bmatrix} = \Phi(t_0, t) \begin{bmatrix}
\xi_0 \\
\dot{\xi}_0
\end{bmatrix} + \int_{t_0}^{t} \Phi(\tau, t) \begin{bmatrix}
0 \\
a(\tau)
\end{bmatrix} d\tau,
\]

(31)

where \(\Phi\) is the one-sided Green’s function matrix associated with the matrix in (30).

Recall that the fundamental property of \(\Phi\) is \(\Phi(a, c) = \Phi(a, b)\Phi(b, c)\), and that \(\Phi(t, t) = I\) and \(\Phi^{-1}(\tau, t) = \Phi(t, \tau)\). In particular, letting

\[
F = \begin{bmatrix}
0 & 1 \\
0 & 0
\end{bmatrix},
\]

(32)

\(\Phi(\tau, t) = e^{(t-\tau)F}\). In which case,

\[
\Phi(t-\tau) = \Phi(\tau, t) = \begin{bmatrix}
1 & t-\tau \\
0 & 1
\end{bmatrix}.
\]

(33)

Of course, in radar tracking \(\xi\) and \(\dot{\xi}\) are generally unknown, and \(a\) is also unknown (and the times at which \(a\) is discontinuous are also unknown).
Now the basic filtering problem of radar tracking is: given a sequence of radar detections on the object, at times \( t_k \), \( k = 0,1,\ldots,K \) (henceforth, \( K \) shall be used exclusively to denote certain integers), ordered as \( t_m < t_n \) when \( m < n \), recursively determine estimates of \( \xi_k \equiv \xi(t_k) \) and \( \dot{\xi}_k \equiv \dot{\xi}(t_k) \). If the object is believed to be maneuvering, \( \ddot{\xi}_k = \ddot{\xi}(t_k) \) may also need to be estimated; and if the object is believed to be motionless (the degenerate case), only \( \xi_k \) needs to be estimated.

Now let \( \mathbf{x}^T = (\xi^T, \dot{\xi}^T) \), and let \( \mathbf{f}^T = (\mathbf{0}^T, \mathbf{a}^T) \). In which case, (30) becomes

\[
\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{f}, \quad \mathbf{x}_0 = \mathbf{x}(t_0),
\]

and the solution is

\[
\mathbf{x}(t) = \Phi(t-t_0)\mathbf{x}_0 + \int_{t_0}^{t} \Phi(t-\tau)\mathbf{f}(\tau)d\tau.
\]

Accordingly, the track is denoted by \( (\hat{\mathbf{x}}; \mathbf{X}) \), with \( \mathbf{X} \) some associated covariance matrix of \( \hat{\mathbf{x}} \) (the determination of \( \mathbf{X} \) is discussed in the next section). In this representation, the elements of the track are respectively

\[
\hat{\mathbf{x}} = \begin{bmatrix} \hat{\xi} \\ \dot{\hat{\xi}} \end{bmatrix} \quad \text{and} \quad \mathbf{X} = \begin{bmatrix} \Xi_{\xi} & \Xi_{\xi\dot{\xi}} \\ \Xi_{\dot{\xi}\xi} & \Xi_{\dot{\xi}} \end{bmatrix},
\]

with \( \mathbf{X} \) tacitly assumed to be symmetric and positive definite (\( \Xi_{\xi} \) and \( \Xi_{\dot{\xi}} \) symmetric and positive definite, and \( \Xi_{\xi\dot{\xi}} = \Xi_{\dot{\xi}\xi}^T \)). When the object is assumed to be motionless or maneuvering, the track will still denoted by \( (\hat{\mathbf{x}}; \mathbf{X}) \), but in those cases the elements are

\[
\hat{\mathbf{x}} = \hat{\xi} \quad \text{and} \quad \mathbf{X} = \Xi_{\xi},
\]

and
\[
\hat{x} = \begin{bmatrix}
\xi
\end{bmatrix}
\text{ and } \begin{bmatrix}
\xi
\dot{\xi}
\end{bmatrix}
\begin{bmatrix}
\xi
\xi
\end{bmatrix} = \begin{bmatrix}
\xi
\xi
\end{bmatrix}.
\]

(38)

These three cases are usually called \textit{constant position} (CP), (37), \textit{constant velocity} (CV), (36), and \textit{constant acceleration} (CA), (38). Note that, in the CP case,

\[
F = 1 \quad \text{and} \quad \Phi(t - \tau) = 1,
\]

the CV case has (32) and (33), and in the CA case

\[
F = \begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
0 & 0 & 0
\end{bmatrix}
\quad \text{and} \quad \Phi(t - \tau) = \begin{bmatrix}
1 & t - \tau & (t - \tau)^2 / 2 \\
0 & 1 & t - \tau \\
0 & 0 & 1
\end{bmatrix}.
\]

(40)

Radar coordinates may also be used to denote a radar track, \((\hat{\mathbf{r}}; \mathbf{R})\), with \(\hat{\mathbf{r}}\) an estimate of \(\mathbf{r}\), and \(\mathbf{R}\) the associated covariance matrix of \(\hat{\mathbf{r}}\). And the CV model may be used in radar coordinates, \(\hat{\mathbf{r}} = \mathbf{F}\mathbf{r}\) with \(\mathbf{r}^T = (\hat{\mathbf{p}}^T, \hat{\mathbf{p}}^T)\). But such defines a spiral in \(\mathbb{E}\); and, in that case, \(CP, CV, CA\) are misnomers – more formally, they denote the (kinematic) order of the model, \(CP \equiv 0, CV \equiv 1, CA \equiv 2\). Note that the same symbol may be used for the associated covariances matrices of detections and tracks. In the sequel, such will be distinguished by writing \((\hat{\mathbf{p}}; \mathbf{R})\) and \((\hat{\mathbf{r}}; \mathbf{R})\) when the context requires. And \((\hat{\zeta}; \mathbf{X})\) and \((\hat{\mathbf{x}}; \mathbf{X})\) will also be used when needed.

Finally, tracks in rectangular coordinates are transformed into radar coordinates, and vice versa. The position sub-vectors are transformed using (1) and (2), and the velocity and acceleration sub-vectors transformed using the first and second derivatives of those functions. For example, a rectangular CV track, \((\hat{\mathbf{x}}; \mathbf{X})\), where \(\hat{\mathbf{x}}^T = (\hat{\xi}^T, \dot{\xi}^T)\), is
transformed into a radar CP track, \((\hat{\rho}; \hat{R})\), by using \((h; H)\), with \(\rho = h(\xi)\) and \(H = [J \ 0]\) – in the CP and CA cases, respectively, the corresponding covariance matrix transformation uses \(H = J\) and \(H = [J \ 0 \ 0]\). (The context shall specify the details.)

### 2.3 The Basic Tracking Equations

In this section the linear Kalman filter (LKF) equations that are commonly used in radar tracking are provided – such are taken from [7]. To expedite the presentation, the form of the detection shall be \((\bar{y}; Y)\), where \(\bar{y}\) is a measurement of \(y\), with \(y = Hx\) (a linear measurement model is being used for now). And so here \((H; H) : (x; X) \rightarrow (y; Y)\). First, the basic tracking algorithm is outlined, and then the details are provided.

Given the first few detections that are believed to be associated with a new object, a “batch” estimator (weighted least-squares, or the BLUE) is used to determine its initial track, denoted \((\hat{x}_0; \hat{X}_0)\) at \(t_0\). The subsequent detections are then reindexed as \(t_n, n = 1, 2, \cdots, N\). Next, given \((\hat{x}_{n-1}; \hat{X}_{n-1})\) at \(t_{n-1}\), a model of motion (an assumption) is used to predict the track at \(t_n\), denoted \((\hat{x}_n^-; \hat{X}_n^-)\). And a predicted detection at \(t_n\) is determined,

\[
(H; H) : (\hat{x}_n^-; \hat{X}_n^-) \rightarrow (\hat{y}_n^-; \hat{Y}_n^-). \tag{41}
\]

In practice, a hypothesis test [52] may also be needed to chose the next detection – something like the Mahalanobis distance [53], with some decision threshold \(\gamma\),

\[
(\hat{y}_n^- - \bar{y}_n^-)^T (\hat{Y}_n^- + \bar{Y}_n^-)^{-1} (\hat{y}_n^- - \bar{y}_n^-) \geq \gamma^2_{H_a}, \tag{42}
\]

where \(H_a\) is the null-hypothesis (associated) and \(H_U\) is the alternative one (unassociated). Under \(H_a\), \((\bar{y}_n^-; \bar{Y}_n^-)\) is used to update \((\hat{x}_n^-; \hat{X}_n^-)\), which yields \((\hat{x}_n; \hat{X}_n)\).
2.3.1 The Prediction Step

In the Kalman approach to tracking the possible positions and motions of the object are modeled as a mean-squared continuous random process, having the form \[54\]

\[
\frac{d}{dt} X_t = FX_t + V_t, \quad X_0 = X(0)
\]  

(for convenience, \(0 \leq t \leq T\), and the initial conditions are given at \(t_0 = 0\)). The above equation is called the system model, and the random process \(V_t\) is called the system noise, assumed to satisfy \(\mathbb{E}V_t = 0\), with \(\mathbb{E}VV_t = S_v(t, \tau)\delta(t - \tau)\), where \(\delta(t - \tau)\) is the Dirac delta function. Also, \(V_t\) is assumed to be independent of \(X_0\) for all \(t \in (0,1]\), with \(S_v(t) = S_v(t,t)\) positive semi-definite (in the literature \(S(t)\) is called the spectral density matrix of the system noise). Note that (43) is analogous to the deterministic case, \[44\]

\[
\dot{x}(t) = f(t) + f(t), \quad x(0) = x_0.
\]

Now under the above assumptions,

\[
\dot{\mu}_x(t) = F\mu_x(t) \quad \text{and} \quad \dot{\Sigma}_x(t) = F\Sigma_x(t) + \Sigma_x(t)F^T + S(t)
\]  

(45)

And so,

\[
\mu_x(t) = \Phi(t)\mu_x(0) \quad \text{and} \quad \Sigma_x(t) = \Phi(t)\Sigma_x(0)\Phi^T(t) + \Sigma_v(t)
\]  

(46)

where

\[
\Sigma_v(t) \equiv \int_0^t \Phi(t - \tau)S(\tau)\Phi^T(t - \tau)d\tau.
\]  

(47)

Such defines the prediction step. For convenience, it shall be written symbolically as

\[
(\mu_x; \Sigma_x) \xrightarrow{(P,S)} (\hat{x}; \hat{X}^-),
\]  

(48)

with the context providing the necessary details (i.e., the kinematic order, number of geometric degrees-of-freedom, and the coordinate representation).
As an example, consider the one degree-of-freedom rectangular CV case. Let

\[ \mathbf{x}^T = (x, \dot{x}) \]; and let the initial conditions to (45) be

\[
\hat{\mathbf{x}}_0 = \begin{bmatrix} \mu_x \\ \mu_x \end{bmatrix} \quad \text{and} \quad \hat{\mathbf{X}}_0 = \begin{bmatrix} \sigma_x^2 & \sigma_{xx} \\ \sigma_{xx} & \sigma_{\dot{x}}^2 \end{bmatrix}.
\] (49)

And let

\[
\mathbf{S}(t) = \begin{bmatrix} 0 & 0 \\ 0 & s \end{bmatrix},
\] (50)

with \( s \) a non-negative constant. The prediction at time \( t \) is then

\[
\hat{\mathbf{x}}(t^-) = \Phi(t)\hat{\mathbf{x}}_0 = \begin{bmatrix} \mu_x + t\mu_{\dot{x}} \\ \mu_{\dot{x}} \end{bmatrix}
\] (51)

and

\[
\hat{\mathbf{X}}(t^-) = \Phi(t)\hat{\mathbf{X}}_0 + \Sigma_v(t),
\] (52)

with

\[
\Phi(t)\hat{\mathbf{X}}_0\Phi^T(t) = \begin{bmatrix} \sigma_x^2 + 2t\sigma_{xx} + t^2\sigma_{\dot{x}}^2 & \sigma_{xx} + t\sigma_{\dot{x}}^2 \\ \sigma_{xx} + t\sigma_{\dot{x}}^2 & \sigma_{\dot{x}}^2 \end{bmatrix}
\] (53)

and

\[
\Sigma_v(t) = s \begin{bmatrix} t^3/3 & t^2/2 \\ t^2/2 & t \end{bmatrix}.
\] (54)

### 2.3.2 The Update Step

The estimation equations of the Kalman filter are basically those of the Best Linear Unbiased Estimator (BLUE) – a derivation of such, also taken from [7], is given in the Appendix. The basic assumptions are that the underlying random vectors of the detection’s measurement and the predicted track are mutually independent.
Given a predicted track and measurement at time $t$, respectively $(\hat{x}^-; \hat{X}^-)$ and $(\bar{y}; \bar{Y})$, with $\bar{Y}$ and $\hat{X}^−$ positive definite, the standard Kalman update equations are
\[
\dot{x} = \hat{x}^- + K(\bar{y} - H \hat{x}^-) \quad \text{and} \quad \dot{X} = (I - KH) \hat{X}^-,
\] (55)
where
\[
K = \hat{X}^−H^T\left(H\hat{X}^−H^T + \bar{Y}\right)^{-1}.
\] (56)

An alternate form is [55]
\[
\dot{x} = \hat{X}\left[(\hat{X}^-)^{-1}\hat{x}^- + H^T\bar{Y}^{-1}\bar{y}\right] \quad \text{and} \quad \dot{X}^{-1} = (\hat{X}^-)^{-1} + H^T\bar{Y}^{-1}H.
\] (57)
with
\[
K = \hat{X}H^T\bar{Y}^{-1}.
\] (58)

Note that, using (58) in the first expression of (55), and letting $\hat{y}^- = H\hat{x}^-$, leads to
\[
\dot{x} = \hat{x}^- + \hat{X}H^T\bar{Y}^{-1}(\bar{y} - \hat{y}^-).
\] (59)

This alternate form is more amenable to analysis. For example, the difference $\bar{y} - \hat{y}^-$ is called the (prediction) residual; and $\bar{Y}^{-1}(\bar{y} - \hat{y}^-)$ and $\hat{X}^{-1}(\hat{x} - \hat{x}^-)$ are called (normalized) innovations. Using (59), the innovations are readily seen to be related as
\[
\hat{X}^{-1}(\hat{x} - \hat{x}^-) = H^T\bar{Y}^{-1}(\bar{y} - \hat{y}^-),
\] (60)
which has the form of a differential. For convenience, the update shall be written symbolically as
\[
(\hat{x}^-; \hat{X}^-) \xrightarrow{\bar{y}, \bar{Y}} (\dot{x}; \dot{X}),
\] (61)
with the details provided by the context.
As an example, consider the one degree-of-freedom rectangular CV case, where \( \mathbf{x}^T = (x, \dot{x}) \). Here the predicted track is \( (\hat{x}^-, \hat{X}^-) \). In component form,

\[
\hat{x}^- = \begin{bmatrix} \hat{x}^- \\ \hat{\dot{x}}^- \end{bmatrix} \quad \text{and} \quad \hat{X}^- = \begin{bmatrix} (\sigma_{xx}^-)^2 & (\sigma_{xx}^-)^2 \\ (\sigma_{xx}^-)^2 & (\sigma_{xx}^-)^2 \end{bmatrix}.
\] (62)

And the “detection” is simply \( (\overline{y}; \sigma_y^2) \), where \( \overline{y} = \mathbf{Hx} + \tilde{y} \) with \( \mathbf{H} = (1, 0) \). In which case, using (59), the updated estimate is \( (\hat{x}; \hat{X}) \), where

\[
\hat{x} = \begin{bmatrix} \hat{x} \\ \hat{\dot{x}} \end{bmatrix} = \begin{bmatrix} \hat{x}^- \\ \hat{\dot{x}}^- \end{bmatrix} + \begin{bmatrix} \sigma_{xx}^2 / \sigma_y^2 \\ \sigma_{xx}^2 / \sigma_y^2 \end{bmatrix} (\overline{y} - \tilde{y})
\] (63)

and

\[
\hat{X}^{-1} = \begin{bmatrix} \sigma_{xx}^2 & \sigma_{xx}^2 \\ \sigma_{xx}^2 & \sigma_{xx}^2 \end{bmatrix}^{-1} = \begin{bmatrix} (\sigma_{xx}^-)^2 & (\sigma_{xx}^-)^2 \\ (\sigma_{xx}^-)^2 & (\sigma_{xx}^-)^2 \end{bmatrix}^{-1} + \begin{bmatrix} 1/\sigma_y^2 & 0 \\ 0 & 0 \end{bmatrix}.
\] (64)

Here the gain matrix, (58), is

\[
\mathbf{K} = \frac{1}{(\sigma_{xx}^-)^2 + \sigma_y^2} \begin{bmatrix} (\sigma_{xx}^-)^2 \\ (\sigma_{xx}^-)^2 \end{bmatrix} = \begin{bmatrix} \sigma_{xx}^2 / \sigma_y^2 \\ \sigma_{xx}^2 / \sigma_y^2 \end{bmatrix}.
\] (65)

### 2.4 A Special Case

In the sequel, when evaluating the results of the analyses, certain closed form solutions to the tracking problem will be used as accuracy references. In this section those references are derived (such are the Cramer-Rao lower bounds of the linear-gaussian case having zero system noise). In particular, let \( \mathbf{x}^T = (x, \dot{x}) \), with \( \ddot{x} = 0 \) and \( s = 0 \). And let the detections provide measurements directly on \( x \), written \( (\overline{x}_n; \sigma_x^2) \), and given at equispaced intervals of time, \( \tau \equiv t_n - t_{n-1}, n = 2, 3, \ldots, N \). In which case the propagation and update equations can be combined into a single “batch” form as follows.
First, define $\mathbf{y}^T \equiv (\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_N)$ and $\Sigma_y \equiv \sigma^2_x \mathbf{I}$ (the random vectors of the measurement errors are orthogonal). And define the “batch detection” to be $(\mathbf{y}; \Sigma_y)$, along with the batch measurement model, $\mathbf{y} = \mathbf{H}x_N + \tilde{y}$, where

$$
\mathbf{H}^T \equiv \begin{bmatrix} 1 & \cdots & 1 & 1 \\
-(N-1)\tau & \cdots & -\tau & 0 \end{bmatrix}.
$$

Then, given $(\mathbf{y}; \Sigma_y)$ with $N \geq 2$, the (batch) BLUE equations are

$$\hat{\mathbf{x}}(t_N) = \hat{\mathbf{X}}(t_N)\mathbf{H}^T\Sigma_y^{-1}\mathbf{y} \quad \text{and} \quad \hat{\mathbf{y}}(t_N) = (\mathbf{H}^T\Sigma_y^{-1}\mathbf{H})^{-1}.$$  

But, since $\Sigma_y = \sigma^2_x \mathbf{I}$, this simplifies to

$$\hat{\mathbf{x}}(t_N) = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y} \quad \text{and} \quad \hat{\mathbf{y}}(t_N) = \sigma^2_x (\mathbf{H}^T\mathbf{H})^{-1},$$

with

$$\mathbf{H}^T\mathbf{H} = \sum_{n=1}^N \begin{bmatrix} 1 & -(n-1)\tau \\
-(n-1)\tau & (n-1)^2\tau^2 \end{bmatrix}.$$  

Of course,

$$\sum_{n=0}^{N-1} 1 = N, \quad \sum_{n=1}^{N-1} n = \frac{(N-1)N}{2}, \quad \sum_{n=0}^{N-1} n^2 = \frac{(N-1)N(2N-1)}{6}.$$  

And so

$$\mathbf{H}^T\mathbf{H} = \begin{bmatrix} N & -(N-1)N\tau/2 \\
-(N-1)N\tau/2 & (N-1)N(2N-1)\tau^2/6 \end{bmatrix}.$$  

and

$$\det \mathbf{H}^T\mathbf{H} = (N-1)N^2(N+1)\tau^2/12.$$  

Thus, for $N \geq 2$,

$$\hat{\mathbf{X}}(t_N) = \sigma^2_x \begin{bmatrix} 2(2N-1)/N(N+1) & 6/N(N+1)\tau \\
6/N(N+1)\tau & 12/(N^2-1)\tau^2 \end{bmatrix}.$$
Now, if a subsequent detection is given at \( t_{N+1} \), the prediction \((\hat{x}_{N+1}^-, \hat{x}_{N+1}^-)\) may be updated using \((\bar{x}_{N+1}; \bar{x}_{N+1}^-)\), to determine \((\hat{x}_{N+1}^+; \hat{x}_{N+1}^+)\). In particular,

\[
\hat{x}_{N+1}^+ = \hat{x}_{N+1}^- + K_{N+1}(\bar{x}_{N+1} - \hat{x}_{N+1}^-) \quad \text{and} \quad \hat{x}_{N+1}^+ = (I - K_{N+1}G)\hat{x}_{N+1}^-,
\]

with \( G = [1 \ 0] \) and

\[
K_{N+1} = \hat{x}_{N+1}^+ G^T \sigma_{\hat{x}}^{-2} = \begin{bmatrix} \alpha_{N+1} \\ \beta_{N+1}/\tau \end{bmatrix},
\]

where, from (72) and the definition of \( G \),

\[
\alpha_N = 2(2m - 1)/(m^2 + m) \quad \text{and} \quad \beta_N = 6/(m^2 + m),
\]

with \( m = N + 2, \ N > 0 \). (This special case is the theoretical basis for the so-called alpha-beta filter, which is perhaps the most commonly used radar tracking method [5].)

Note that the variance of the position estimate in (72) is asymptotically \( 4\sigma_{\hat{x}}^2/N \). In the CP case it is exactly \( \sigma_{\hat{x}}^2/N \). And in the CA case it is asymptotically \( 9\sigma_{\hat{x}}^2/N \).

Indeed, if a \( p \)-th kinematic order model were used, satisfying the above assumptions, the variance of the position estimate would be asymptotically \( \sigma_{\hat{x}}^2 (p + 1)^2/N \) [56, 57].

### 2.5 The Basic Estimation Cases and Notation

In the sequel several different estimators shall be used. For convenience, the basic ones are summarized here (with time not explicitly denoted) – the next Chapter shall provide the details. All are predictor-corrector loops as defined above: given a prior track, the propagation equations are used to determine a prediction; and, given a predicted track and a detection, the update equations are used to determine a correction.

- **The Linear Kalman Filter (LKF).** Here the radar detections are used directly to determine the track in radar coordinates. The prediction and update steps are simply
\[ (\hat{f}; \hat{R}) \xrightarrow{(F,S)} (\hat{f}^-; \hat{R}^-) \quad \text{and} \quad (\hat{r}; \hat{R}) \xrightarrow{(r;R)} (\hat{r}; \hat{R}). \] (76)

- **The Pseudo-LKF (PLKF).** This case uses the LKF, but with the detections first converted into rectangular coordinates. That is, given \((\bar{f}; \bar{R})\), first determine
\[ (h^{-1}; J^{-1}): (\bar{f}; \bar{R}) \mapsto (x'; \bar{X}), \] (77)
called a pseudo-detection. And then use
\[ (\hat{x}; \hat{X}) \xrightarrow{(F,S)} (\hat{x}^-; \hat{X}^-) \quad \text{and} \quad (\hat{x}^-; \hat{X}^-) \xrightarrow{(r;X)} (\hat{x}; \hat{X}). \] (78)
Alternatively, the update step may be written
\[ (\hat{x}^-; \hat{X}) \xrightarrow{(h^{-1}-\cdot J^{-1}); (\bar{r}; \bar{R}) \mapsto (\bar{X}; \bar{X})} (\hat{x}; \hat{X}). \] (79)

- **The Converted-LKF (CLKF).** Here the track is predicted in rectangular coordinates and updated in radar coordinates. In particular, before and after each update the track is respectively transformed as
\[ (h; H): (\hat{x}^-; \hat{X}^-) \mapsto (\hat{r}^-; \hat{R}^-) \quad \text{and} \quad (h^{-1}; H^{-1}): (\hat{r}; \hat{R}) \mapsto (\hat{x}; \hat{X}). \] (80)
And the prediction and update steps are
\[ (\hat{x}; \hat{X}) \xrightarrow{(F,S)} (\hat{x}^-; \hat{X}^-) \quad \text{and} \quad (\hat{r}^-; \hat{R}^-) \xrightarrow{(r;R)} (\hat{r}; \hat{R}). \] (81)
A variation of this case is the **Radar Principal Cartesian Coordinates (RPCC) method** [58]. There the prediction update steps are
\[ (\hat{x}; \hat{R}) \xrightarrow{(F,S)} (\hat{x}^-; \hat{R}^-) \quad \text{and} \quad (\hat{r}^-; \hat{R}^-) \xrightarrow{(r;R)} (\hat{r}; \hat{R}), \] (82)
with \((h; I): (\hat{x}^-; \hat{R}^-) \mapsto (\hat{r}^-; \hat{R}^-) \quad \text{and} \quad (h^{-1}; I): (\hat{r}; \hat{R}) \mapsto (\hat{x}; \hat{R}). \)

- **The Extended Kalman Filter (EKF).** This case uses the radar detections directly to update the track in rectangular coordinates. Its prediction and upstate steps are
\[ (\hat{x}; \hat{X}) \xrightarrow{(F,S)} (\hat{x}^-; \hat{X}^-) \quad \text{and} \quad (\hat{x}^-; \hat{X}^-) \xrightarrow{(r;R)} (\hat{x}; \hat{X}). \] (83)
Finally, the basic notation is summarized as follows. Non-bold italic lowercase symbols shall denote real scalars; and non-bold italic uppercase symbols shall denote either Euclidean points or random variables (the context will specify). Bold symbols such as $\xi, x, X, X$ shall denote objects that are referenced to rectangular coordinates; while $\rho, r, R, R$ shall denote the corresponding ones that are referenced to radar coordinates. In particular, column vectors of scalars will be denoted by symbols such as $x$ and $r$; when they are realizations of random vectors, those functions will be denoted as $X$ and $R$; and symbols such as $X$ and $R$ will denote the corresponding covariance matrices. There will be exceptions to these rules: for example, the domain of the independent variable, time, a closed real interval, written $[0, T]$, with “$T$” a real number; $X, Y, R, A$ may also denote random variables (the context shall specify); and $K, M, N, L$ shall always denote integers.
3 Illustration of the Problem

In this Chapter the problem to be analyzed is illustrated by using a simple example that Julier and Uhlmann employed to motivate their Unscented Kalman Filter (UKF) [19]. A set of unbiased and independent and identically distributed (iid) gaussian radar measurements on $P \in \mathbb{R}$ are averaged to estimate $(r, a)$; and the corresponding rectangular pseudo-measurements are averaged to estimate $(x, y)$. The former is optimal in the unbiased and minimum variance sense, but the latter is biased and is more noisy. Various linear Kalman filters are then used, and similar problems occur. Finally, the extended Kalman filter is employed: it is less biased and less noisy, but it has some convergence issues. The results are summarized in the concluding Section of this Chapter.

3.1 The Basic Estimation Bias Problem

In the aforementioned “exemplar” of Julier and Uhlmann, the object was located at the point $P \in \mathbb{R}$, having rectangular and radar coordinates $(x, y) = (0, 1)$ and $(r, a) = (1, \pi/2)$. A sequence of mutually independent radar measurements on $P$ was given, $\tilde{r}_n = r + \tilde{a}_n$ and $\tilde{a}_n = a + \tilde{a}_n$, $n = 1, 2, \cdots, N$, whose errors, $\tilde{r}_n$ and $\tilde{a}_n$, were distributed as $\mathcal{N}(0; \sigma^2_r)$ and $\mathcal{N}(0; \sigma^2_A)$, with $\sigma_r = 0.02$ meters and $\sigma_A = \pi/12$ radians. And the corresponding sequence of rectangular measurements on $P$ was obtained as

$$\bar{x}_n = \tilde{r}_n \cos \tilde{a}_n \quad \text{and} \quad \bar{y}_n = \tilde{r}_n \sin \tilde{a}_n.$$  

(84)

Then, to respectively estimate $(r, a)$ and $(x, y)$, the averages of the two sets of measurements were determined,

$$\hat{r}_n = \left[ \begin{array}{c} \hat{r}_n \\ \hat{a}_n \end{array} \right] = \frac{1}{n} \sum_{m=1}^{n} \left[ \begin{array}{c} \tilde{r}_m \\ \tilde{a}_m \end{array} \right] \quad \text{and} \quad \hat{x}_n = \left[ \begin{array}{c} \hat{x}_n \\ \hat{y}_n \end{array} \right] = \frac{1}{n} \sum_{m=1}^{n} \left[ \begin{array}{c} \bar{x}_m \\ \bar{y}_m \end{array} \right].$$

(85)
Here, to facilitate comparisons between the two cases, the point $P$ is chosen such that $(x, y) = (r, a)$. Specifically, let $x = r = 1$ (meters) and let $y = a = 0$ (meters and radians).

And, to make the effects of the nonlinearities more pronounced, let $\sigma_A = \pi/6$ (twice the value that Julier and Uhlmann used). In which case the underlying random vector of the radar measurements is distributed according to

$$p_R(r) = \frac{1}{2\pi(\det \Sigma_R)^{1/2}} \exp \left[ -\frac{1}{2} (r - \mu_R)^T \Sigma_R^{-1} (r - \mu_R) \right],$$

with

$$\mu_R = \begin{bmatrix} \mu_R \\ \mu_A \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{and} \quad \Sigma_R = \begin{bmatrix} \sigma_R^2 & \sigma_{RA} \\ \sigma_{AR} & \sigma_A^2 \end{bmatrix} = \begin{bmatrix} .004 & 0 \\ 0 & \pi^2/36 \end{bmatrix}. \quad (87)$$

Figure 2 illustrates the ensuing radar measurements and rectangular pseudo-measurements for $N = 10,000$. Also shown is the sequence of rectangular averages, $(\hat{x}_n', \hat{y}_n')$, $n = 1, 2, \ldots, N$. The sequence of $(\hat{r}, \hat{a})$'s is obscured by the measurements.

![Figure 2 The basic estimation bias problem](image)

In above figure the two sets of measurements are shown as coordinate-points. The $(\hat{r}, \hat{a})$'s are plotted against axes that are linear in range and azimuth; and the $(\hat{x}', \hat{y}')$'s
er overlaid using those axes as range and cross-range. (The right-hand side provides an expanded view.) There effects of the nonlinear coordinate transformation are clearly seen: the gaussian set of radar measurements has an elliptical shape; while the corresponding set of rectangular coordinates has a circular shape. Also, the sequence of \((\hat{x}', \hat{y}')\)'s appears to be biased – it does not converge to \((1,0)\).

3.1.1 The Scalar-Weight Case

Here the effects of the nonlinear coordinate transformation upon the above estimates are further illustrated. But, in anticipation of the sequel, the recursive form of the weighted average will now be used. Recall that in the scalar-weight case the coordinate form of a detection is written \((\hat{r};\hat{I}/\hat{w})\) or \((\hat{x}';\hat{I}/\hat{w})\), with \(\hat{w} > 0\); and the estimates are written \((\hat{r};\hat{I}/\hat{w})\) and \((\hat{x}';\hat{I}/\hat{w})\), with \(\hat{w} > 0\).

Let a sequence of distinct radar detections be given, \((\hat{r}_n;\hat{I}_n/\hat{w}_n)\), \(n = 1,2,\ldots,N\) (the measurements are distinct; the weights may all be the same). Their weighted average is

\[
\hat{r}_N = \frac{1}{\hat{w}_N} \sum_{n=1}^{N} \hat{w}_n \hat{r}_n \quad \text{and} \quad \hat{w}_N = \sum_{n=1}^{N} \hat{w}_n. \tag{88}
\]

Note that (85) and (88) become same when all the weights are equal. Indeed,

\[
\frac{1}{\hat{w}_n} \sum_{m=1}^{n} \hat{w}_m \hat{r}_m = \frac{1}{n} \sum_{m=1}^{n} \hat{r}_m \quad \text{and} \quad \hat{w}_n = \sum_{m=1}^{n} \hat{w}_m = n\hat{w}. \tag{89}
\]

The recursive fusion form of (88) is (see the Appendix)

\[
\hat{r}_n = (\hat{w}_{n-1}/\hat{w}_n) \hat{r}_{n-1} + (\hat{w}_n/\hat{w}_n) \hat{r}_n \quad \text{and} \quad \hat{w}_n = \hat{w}_{n-1} + \hat{w}_n, \tag{90}
\]

\(n = 1,2,\ldots,N\), with \((\hat{r}_1;\hat{I}_1/\hat{w}_1) = (\hat{r}_1;\hat{I}_1/\hat{w}_1)\). And the corresponding update form is

\[
\hat{r}_n = \hat{r}_{n-1} + (\hat{w}_n/\hat{w}_n)(\hat{r}_n - \hat{r}_{n-1}) \quad \text{and} \quad \hat{w}_n = \hat{w}_{n-1} + \hat{w}_n. \tag{91}
\]

These two recursive forms shall be written symbolically as
\[
(\hat{r}_{n-1}; I/\hat{w}_{n-1}) \xrightarrow{(\bar{r}, \bar{a}/\bar{w})} (\hat{r}_n; I/\hat{w}_n).
\]

(92)

with \((\hat{r}_i; I/\hat{w}_i) = (\bar{r}_i; I/\bar{w}_i)\). In the rectangular (pseudo) measurement case the equations have the same form as those given above. And the recursive update is written

\[
(\hat{x}'_{n-1}; I/\hat{w}_{n-1}) \xrightarrow{(\bar{x}, \bar{a}/\bar{w})} (\hat{x}'_n; I/\hat{w}_n),
\]

(93)

with \((\hat{x}'_i; I/\hat{w}_i) = (\bar{x}_i; I/\bar{w}_i)\).

To illustrate further the effects of the coordinate transformation upon the estimates, let \(\bar{w}_m = \bar{w}, n = 1, 2, \ldots, N\), and partition the \((\bar{r}, \bar{a})\)'s in Figure 2 into \(M\) disjoint subsets of length \(L\) each, with \(LM = N\). And then, for \(m = 1, 2, \ldots, M\), determine

\[
\hat{r}_{m,l} = \frac{1}{L} \sum_{k=1}^{L} \bar{r}_{m,k} \quad \text{and} \quad \hat{x}'_{m,l} = \frac{1}{L} \sum_{k=1}^{L} \bar{x}_{m,k}, \quad l = 1, 2, \ldots, L.
\]

(94)

Figure 3 shows the case where \(M = 50\) and \(L = 200\). Here such sequences shall be called estimation paths (of length \(L\)). More formally, they are mutually independent Monte Carlo trials, with each trial a Markov chain starting at a given random draw.
Of course, the radar averages may be converted into rectangular coordinates to estimate \((x, y)\). That is, after determining each \((\hat{r}_{m,l}, \hat{a}_{m,l})\) in (94), transform them using

\[
\hat{x}_{m,l} = \hat{r}_{m,l} \cos \hat{a}_{m,l} \quad \text{and} \quad \hat{y}_{m,l} = \hat{r}_{m,l} \sin \hat{a}_{m,l}.
\]

(95)

Figure 4 shows the sequences of these converted estimates, denoted by \(\hat{x} = \hat{r} \cos \hat{a}\) and \(\hat{y} = \hat{r} \sin \hat{a}\), along with the original \((\hat{x}', \hat{y}')\)'s of Figure 3.

Similarly, the \((\hat{x}', \hat{y}')\)'s may be converted back into radar coordinates to provide estimates of \((r, a)\). That is, given \((\hat{x}'_{m,l}, \hat{y}'_{m,l})^T = (\hat{x}_{m,l}', \hat{y}_{m,l}')\) from the second expression of (94), determine

\[
\hat{r}'_{m,l} = \sqrt{\hat{x}'^2_{m,l} + \hat{y}'^2_{m,l}} \quad \text{and} \quad \hat{a}'_{m,l} = \arctan \left( \frac{\hat{y}'_{m,l}}{\hat{x}'_{m,l}} \right).
\]

(96)

Figure 5 shows the sequences of these converted-back estimates, denoted \(\hat{r}' = \sqrt{\hat{x}'^2 + \hat{y}'^2}\) and \(\hat{a}' = \arctan(\hat{y}', \hat{x}')\), together with the original \((\hat{r}, \hat{a})\)'s of Figure 3.
3.1.2 The Matrix-Weight Case

In this section the update equations of the Linear Kalman Filter (LKF) are used to determine the estimates. Here the measurements and estimates have associated covariance matrices, and \((\mathbf{h}^{-1}; \mathbf{J}^{-1}): (\mathbf{f}; \mathbf{R}) \mapsto (\mathbf{x}'; \mathbf{X}')\), where \(\mathbf{R} = \mathbf{W}^{-1}\) with

\[
\mathbf{W} = \begin{bmatrix}
    w_r & 0 \\
    0 & w_a
\end{bmatrix}.
\] (97)

Given \((\mathbf{f}_n; \mathbf{R}_n), n = 1, 2, \ldots, N\), the batch form of the BLUE is (see the Appendix)

\[
\hat{r}_n = \hat{R}_n \sum_{k=1}^{n} \hat{R}_k^{-1} \hat{r}_k \quad \text{and} \quad \hat{R}_n^{-1} = \sum_{k=1}^{n} \hat{R}_k^{-1}.
\] (98)

Its recursive fusion form is

\[
\hat{r}_n = \hat{R}_n \hat{r}_n^{-1} \hat{r}_{n-1} + \hat{R}_n \hat{R}_n^{-1} \hat{r}_n \quad \text{and} \quad \hat{R}_n^{-1} = \hat{R}_{n-1} + \hat{R}_n^{-1},
\] (99)

and its recursive update form is

\[
\hat{r}_n = \hat{r}_{n-1} + \hat{R}_n \hat{R}_n^{-1} (\mathbf{f}_n - \hat{r}_{n-1}) \quad \text{and} \quad \hat{R}_n^{-1} = \hat{R}_{n-1} + \hat{R}_n^{-1}.
\] (100)
These recursive forms are written symbolically as

\[(\hat{r}_{n-1}; \hat{R}_{n-1}) \xrightarrow{(r_n; R_n)} (\hat{r}_n; \hat{R}_n). \] (101)

Similarly, determine

\[
\hat{x}' = \hat{x}' + \sum_{k=1}^{n} \hat{x}'_{k-1} \hat{x}_k \quad \text{and} \quad \hat{x}'_{n-1} = \sum_{k=1}^{n} \hat{x}'_{k-1}, \] (102)

where \((h^{-1}; J) : (\bar{r}_n; \bar{R}_n) \mapsto (\bar{x}_n'; \bar{X}_n'), \ n = 1, 2, \cdots, N\). That is,

\[\bar{x}' = h^{-1}(\bar{r}_n) \quad \text{and} \quad \bar{X}_n' = J^{-1}(\bar{r}_n) \bar{R}_n J^{-T}(\bar{r}_n). \] (103)

Expressions similar to those in (99) and (100) follow. Also,

\[(\hat{x}'_{n-1}; \hat{X}'_{n-1}) \xrightarrow{(x_n; X_n)} (\hat{x}_n'; \hat{X}_n'). \] (104)

or, equivalently,

\[(\hat{x}'_{n-1}; \hat{X}'_{n-1}) \xrightarrow{(h^{-1}; J) : (\bar{r}_n; \bar{R}_n) \mapsto (\bar{x}_n'; \bar{X}_n')} (\hat{x}_n'; \hat{X}_n'). \] (105)

Note that, when \(\bar{R}_n = \Sigma_R, \ n = 1, 2, \cdots, N\), see (87), the matrix-weight estimates determined by (98) are the same as those determined in the scalar-weight case,

\[\hat{r}_n = \frac{1}{N} \sum_{n=1}^{N} \Sigma^{-1}_R \bar{r}_n = \frac{1}{N} \sum_{n=1}^{N} \bar{r}_n \quad \text{and} \quad \hat{R}_n = \Sigma_R / N. \] (106)

But for \(n = 2, 3, \cdots, N\), the rectangular estimates determined by (102) are not the same as the ones in the corresponding scalar-weight case: the \(\bar{r}\)’s in (103) are all distinct, and so the \(\bar{X}'\)’s are also distinct.

Figure 6 illustrates the estimates determined by (102) for the equi-weighted case, \(\bar{R}_n = \Sigma_R, \ n = 1, 2, \cdots, N\), denoted by \(\hat{x}^{(m)}\) and \(\hat{y}^{(m)}\) – the superscript “\((m)\)” indicates the matrix-weight case. Here the same subsets of measurements used in Figure 3 are being reused, and so the estimates shown there are repeated, but now they are labeled \(\hat{x}'^{(s)}\) and
\( \hat{y}^{(s)} \) – the superscript “\((s)\)” indicates the *scalar-weight* case. Also shown in Figure 6 are the corresponding estimates of \((x, y)\) determined by (95). Figure 7 provides the estimates in Figure 6 converted back into radar coordinates.

**Figure 6** The scalar- and matrix-weight cases (rectangular coordinates)

**Figure 7** The scalar- and matrix-weight cases (radar coordinates)
Note that, as in the scalar-weight case, the matrix-weight estimation equations, (101) and (104), have the same form – both are LKF’s. But in the sequel, just the one that uses the radar measurements directly, (101), shall be called an LKF. The other case, (104), which uses the rectangular pseudo-measurements, shall be called a Pseudo-LKF (PLKF). Also, the case where the LKF estimates of \((r, a)\) are converted into rectangular coordinates shall be called a Converted LKF (C-LKF); and the case where PLKF estimates of \((x, y)\) are converted back into radar coordinates shall be called a Converted Back PLKF (CB-PLKF).

### 3.2 The Extended Kalman Filter

The LKF and PLKF defined above determine their estimates in the same coordinate system as the measurements they were given. That is,

\[
(\hat{r}_{n-1}; \hat{R}_{n-1}) \rightarrow (\hat{r}_n; \hat{R}_n) \quad \text{and} \quad (\hat{x}_{n-1}; \hat{X}_{n-1}) \rightarrow (\hat{x}_n; \hat{X}_n).
\]

In contrast, the Extended Kalman Filter (EKF) uses the radar measurements directly to estimate \((x, y)\). Symbolically, such is written

\[
(\hat{x}_{n-1}^*; \hat{X}_{n-1}^*) \rightarrow (\hat{x}_n^*; \hat{X}_n^*). \tag{107}
\]

In particular, given a prior track in rectangular coordinates, \((\hat{x}_{n-1}^*; \hat{X}_{n-1}^*)\), and given a subsequent detection in radar coordinates, \((\hat{r}_n; \hat{R}_n)\), the EKF update is [7]

\[
\hat{x}_n^* = \hat{x}_{n-1}^* + K_{n-1} (\hat{r}_n - h(\hat{x}_{n-1}^*)) \quad \text{and} \quad \hat{X}_n^* = [I - K_{n-1} J(\hat{x}_{n-1}^*)] \hat{X}_{n-1}^*, \tag{108}
\]

\[
K_n \equiv \hat{X}_{n-1}^* J'(\hat{x}_{n-1}^*) \left[ J(\hat{x}_{n-1}^*) \hat{X}_{n-1}^* J'(\hat{x}_{n-1}^*) + \hat{R} \right]^{-1}. \tag{109}
\]

To begin this recursion, let \((\hat{x}_{i}^*; \hat{X}_{i}^*) = (\hat{x}_i; \hat{X}_i)\).

Figure 8 illustrates the EKF estimates, along with the corresponding scalar- and
matrix-weight PLKF cases that were shown earlier – the same sets of detections used above are being reused here. Figure 9 shows the corresponding converted back case, CB-EKF, together with the CB-PLKF.
3.3 Discussion of the Basic Estimation Problem

The above results are summarized as follows. The LKF and C-LKF estimates seem to be unbiased and have the least variance. The PLKF and CB-PLKF estimates seem to be biased and noisier than the corresponding LKF and C-LKF ones. And the biases of the scalar- and matrix-weight PLKF estimates have the opposite sense. The EKF seems to be less biased and less noisy than the PLKF, but it appears to have convergence issues.

Now $\sigma_A = \pi/6$ in the cases shown above, but Julier and Uhlmann used $\sigma_A = \pi/12$. And so repeat the above cases using their smaller $\sigma_A$. Figure 10 provides those PLKF and EKF estimates; and Figure 11 provides the corresponding CB-PLKF and CB-EKF cases. Note that the EKF now appears to be unbiased, and the PLKF seems to be less biased. Also, the EKF convergence problem has been mitigated somewhat.

![Figure 10 PLKF and EKF rectangular estimates ($\sigma_A = \pi/12$)](image)
Finally, Figure 12 and Figure 13 respectively compare the sample means for the rectangular and radar coordinate sets of cases that have been shown in this chapter. And a comparison of the sample standard deviations is given below in Figure 14 and Figure 15. But, for the sake of comparison, they are normalized by \( \sigma_X = \sigma_R = .02 \) and by \( \sigma_Y = \sigma_A \), with either \( \sigma_A = \pi/6 \) or \( \sigma_A = \pi/12 \) (respectively the solid and dashed curves). In these figures the solid curves denote the \( \sigma_A = \pi/6 \) sub-cases, and the dashed ones denote the \( \sigma_A = \pi/12 \) sub-cases. The (normalized) optimal case is also shown, the smooth curves, \( 1/\sqrt{n} \). Also, the component forms of the underlying random vectors for \((\hat{x}'_r, \hat{y}'_r)\) and \((\hat{r}', \hat{a}')\) are respectively

\[
\hat{X}'^T(l) = \left( \hat{X}'(l), \hat{Y}'(l) \right) \quad \text{and} \quad \hat{R}'^T(l) = \left( \hat{R}'(l), \hat{A}'(l) \right)
\]  

(110)

– with those for the other estimates are written similarly.
Figure 12 Comparison of the sample means (rectangular cases)

Figure 13 Comparison of the sample means (radar cases)
Figure 14 Normalized sample standard deviations (rectangular cases)

Figure 15 Normalized sample standard deviations (radar cases)
For later reference, here the sample means and sample covariance matrices for the LKF estimates are respectively determined as

\[
\hat{\mu}_{R(l)} = \begin{bmatrix} \hat{\mu}_{R(l)} \\ \hat{\mu}_{\lambda(l)} \end{bmatrix} = \frac{1}{M} \sum_{m=1}^{M} \begin{bmatrix} \hat{r}_{m,l} \\ \hat{a}_{m,l} \end{bmatrix} = \frac{1}{M} \sum_{m=1}^{M} \begin{bmatrix} \hat{r}_{m,l} \\ \hat{a}_{m,l} \end{bmatrix}
\]

(111)

and

\[
\Sigma_{R(l)} = \begin{bmatrix} \hat{\sigma}^2_{R(l)} & \hat{\sigma}_{R(l)\lambda(l)} \\ \hat{\sigma}_{R(l)\lambda(l)} & \hat{\sigma}^2_{\lambda(l)} \end{bmatrix} = \frac{1}{M-1} \sum_{m=1}^{M} \begin{bmatrix} \hat{r}_{m,l} - \hat{\mu}_{R(l)} \\ \hat{a}_{m,l} - \hat{\mu}_{\lambda(l)} \end{bmatrix} \left[ \begin{bmatrix} \hat{r}_{m,l} - \hat{\mu}_{R(l)} \\ \hat{a}_{m,l} - \hat{\mu}_{\lambda(l)} \end{bmatrix} \right]^T.
\]

(112)

– with those for the other estimates are written similarly.
4 Analysis of the PLKF Update

In the previous Chapter the PLKF was defined to be an LKF that uses rectangular pseudo-measurements to estimate \((x, y)\). And there two PLKF cases were distinguished: scalar-weight and matrix-weight. Both PLKF’s were seen to be biased, and usually worse than simply converting the (optimal) LKF estimates of \((r, a)\) using \(\hat{x} = \hat{r} \cos \hat{a}\) and \(\hat{y} = \hat{r} \sin \hat{a}\) – the C-LKF case. Here those estimators are discussed more formally, and certain analytic results are obtained for use in the sequel.

4.1 The Pseudo-Measurement Errors

Recall that the canonical measurement model used in radar tracking is \(\bar{r} = r + \tilde{r}\), where \(r = h(x)\), with the measurement error, \(\tilde{r}\), tacitly assumed to be independent of \(x\).

Similarly, the model of the rectangular pseudo-measurements is \(\bar{x}' = x + \tilde{x}'\). But, unlike the radar measurements, the components of \(\tilde{x}'\) are not independent of \(x\). Indeed, substituting \(r + \tilde{r}\) and \(a + \tilde{a}\) into \(\bar{x}' = \bar{r} \cos \bar{a}\) and \(\bar{y}' = \bar{r} \sin \bar{a}\) yields

\[
\begin{align*}
\bar{x}' &= r \cos a \cos \tilde{a} - r \sin a \sin \tilde{a} + \frac{\tilde{r}}{r} (r \cos a \cos \tilde{a} - r \sin a \sin \tilde{a}) \\
\bar{y}' &= r \sin a \cos \tilde{a} + r \cos a \sin \tilde{a} + \frac{\tilde{r}}{r} (r \sin a \cos \tilde{a} + r \cos a \sin \tilde{a}),
\end{align*}
\]

which simplify to

\[
\begin{bmatrix}
\bar{x}' \\
\bar{y}'
\end{bmatrix} = (1 + \tilde{r}/r) \begin{bmatrix}
\cos \tilde{a} & -\sin \tilde{a} \\
+\sin \tilde{a} & \cos \tilde{a}
\end{bmatrix} \begin{bmatrix}
x \\
y
\end{bmatrix} + \begin{bmatrix}
\cos \tilde{a} \\
\sin \tilde{a}
\end{bmatrix} \tilde{r}.
\] (113)

That is, the errors of the pseudo-measurements depend nonlinearly upon both \(x\) and \(\tilde{r}\),

\[
\begin{align*}
\bar{x}' &= \bar{x} - x \\
\bar{y}' &= \bar{y} - y
\end{align*}
\]

\[
\begin{align*}
\bar{x}' &= (\cos \tilde{a} - 1) x - \sin \tilde{a} y + \frac{\tilde{r}}{r} \cos \tilde{a} \\
\bar{y}' &= +\sin \tilde{a} x + (\cos \tilde{a} - 1) y + \frac{\tilde{r}}{r} \sin \tilde{a}.
\end{align*}
\] (114)
4.1.1 The Expected Value of the Pseudo-Measurements

The underlying random vectors of \( \mathbf{r} = \mathbf{r} + \tilde{\mathbf{r}} \) are related as \( \mathbf{R} = \mathbf{r} + \tilde{\mathbf{R}} \). Let \( \tilde{\mathbf{R}}' = (\tilde{\mathbf{R}}, \tilde{\mathbf{A}}) \), with \( \tilde{\mathbf{R}} \) and \( \tilde{\mathbf{A}} \) independent and distributed as \( \tilde{\mathbf{R}} \sim \mathcal{N}(0; \sigma^2_{\tilde{R}}) \) and \( \tilde{\mathbf{A}} \sim \mathcal{N}(0; \sigma^2_{\tilde{A}}) \). The underlying random vector of the rectangular pseudo-measurements is \( \mathbf{X}' = \mathbf{h}^{-1}(\mathbf{R}) \), and its components are

\[
X' = R \cos A \quad \text{and} \quad Y' = R \sin A.
\]

But since \( R \) and \( A \) are independent, the expected values of these expressions factor as

\[
\mu_{X'} = E(X') = (E)R(E \cos A) \quad \text{and} \quad \mu_{Y'} = E(Y') = (E)R(E \sin A).
\]

And, since \( E = r \) and \( A = a + \tilde{A} \),

\[
\begin{bmatrix}
\mu_{X'} \\
\mu_{Y'}
\end{bmatrix}
= r
\begin{bmatrix}
\cos a & -\sin a \\
\sin a & \cos a
\end{bmatrix}
\begin{bmatrix}
E \cos \tilde{A} \\
E \sin \tilde{A}
\end{bmatrix}.
\]

By symmetry, \( E \sin \tilde{A} = 0 \). Thus, \( E X' = xE \cos \tilde{A} \), where

\[
E \cos \tilde{A} = \int_{-\infty}^{\infty} \cos(\alpha) p_{\tilde{A}}(\alpha) d\alpha.
\]

This integral can be evaluated using a technique outlined in [46]. Let

\[
I(t) = \int_{-\infty}^{\infty} [\cos(\alpha t)] p_{\tilde{A}}(\alpha) d\alpha.
\]

Since \( \cos(\alpha t) \) is continuous at any finite \( \alpha \) and \( t \),

\[
\frac{d}{dt} I(t) = \int_{-\infty}^{\infty} \left[ \frac{\partial}{\partial t} \cos(\alpha t) \right] p_{\tilde{A}}(\alpha) d\alpha = -\int_{-\infty}^{\infty} \alpha \sin(\alpha t) p_{\tilde{A}}(\alpha) d\alpha.
\]

Integration by parts, \( \int u dv = uv - \int v du \), where \( u = \sin(\alpha t) \) and \( v = \sigma^2_{\tilde{A}} p_{\tilde{A}}(\alpha) \), yields

\[
\frac{d}{dt} I(t) = \sin(\alpha t) \sigma^2_{\tilde{A}} p_{\tilde{A}}(\alpha) \bigg|_{t=\infty}^{t=0} - t \sigma^2_{\tilde{A}} \int_{-\infty}^{\infty} \cos(\alpha t) p_{\tilde{A}}(\alpha) d\alpha.
\]

In the above expression, since \( p_{\tilde{A}} \) is of order \( e^{-\alpha^2} \), the first summand is zero,
\[
\lim_{a \to \pm \infty} \sin(\alpha t) \sigma_A^2 p_A(\alpha) \bigg|_{\alpha = \pm a} = 0.
\]

And the second summand is
\[
t \sigma_A^2 \int_{-\infty}^{+\infty} \cos(\alpha t) p_A(\alpha) d\alpha = t \sigma_A^2 I(t).
\]

Thus,
\[
\frac{d}{dt} I(t) = -t \sigma_A^2 I(t).
\]

The solution to this differential equation is
\[
I(t) = I_0 \exp(-t^2 \sigma_A^2 / 2),
\]

where \( I_0 \) is an integration constant that is independent of \( t \). Let \( t = 0 \) in (118) to obtain \( I_0 = 1 \); and then let \( t = 1 \) with \( I_0 = 1 \) to obtain
\[
\mathcal{E} \cos \tilde{A} = \int_{-\infty}^{+\infty} \cos(\alpha) p_A(\alpha) d\alpha = e^{-\sigma_A^2 / 2}.
\]

Thus, the expected value of \( X' \) is
\[
\mu_{X'} = \begin{bmatrix} \mu_{X'} \\ \mu_{Y'} \end{bmatrix} = \begin{bmatrix} \cos a \\ \sin a \end{bmatrix} e^{-\sigma_A^2 / 2} = \begin{bmatrix} x' \\ y' \end{bmatrix} e^{-\sigma_A^2 / 2}.
\]

Note that \( \mu_{X'} = xe^{-\sigma_A^2 / 2} \neq x \) when \( \sigma_A \neq 0 \).

### 4.1.2 The Covariance Matrix of the Pseudo-Measurements

By definition, the covariance matrix of the rectangular pseudo-measurements is
\[
\Sigma_{X'} \equiv \mathcal{E}(X' - \mu_{X'})(X' - \mu_{X'})^T = \begin{bmatrix} \sigma_{X'}^2 & \sigma_{X'Y'} \\ \sigma_{X'Y'} & \sigma_{Y'}^2 \end{bmatrix}.
\]

Which is equivalent to
\[
\mathcal{E}(X' - \mu_{X'})(X' - \mu_{X'})^T = \mathcal{E}X'X'^{\top} - \mu_{X'}\mu_{X'}^{\top}.
\]

From (122),
\[ \mathbf{\mu}_X^T \mathbf{\mu}_X = \begin{bmatrix} \mu_X & \mu_X \mu_Y^T \\ \mu_Y & \mu_Y^T \end{bmatrix} = e^{-\sigma_A^2} \begin{bmatrix} x^2 & xy \\ yx & y^2 \end{bmatrix}. \]  

(125)

And so, to evaluate \( \Sigma \), just \( \mathcal{E}X^{\prime T} \mathcal{E} \) remains to be determined.

Using (116) and the mutual independence of \( R \) and \( A \),

\[ \mathcal{E}X^{\prime T} \mathcal{E} = \mathcal{E} \begin{bmatrix} X'^2 \ 
\end{bmatrix} \mathcal{E} = (\mathcal{E}R^2) \mathcal{E} \begin{bmatrix} \cos^2 A & \cos A \sin A \\ \cos A \sin A & \sin^2 A \end{bmatrix}. \]  

(126)

Of course,

\[ 2 \begin{bmatrix} \cos^2 A & \cos A \sin A \\ \cos A \sin A & \sin^2 A \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} +\cos 2A & \sin 2A \\ \sin 2A & -\cos 2A \end{bmatrix}, \]  

(127)

and

\[ \begin{bmatrix} +\cos 2A & \sin 2A \\ \sin 2A & -\cos 2A \end{bmatrix} = \cos 2\tilde{A} \begin{bmatrix} +\cos a & \sin a \\ \sin a & -\cos a \end{bmatrix} + \sin 2\tilde{A} \begin{bmatrix} -\sin a & \cos 2a \\ \cos a & \sin 2a \end{bmatrix}. \]  

(128)

By symmetry \( \mathcal{E} \sin 2\tilde{A} = 0 \). And using (121) with \( \beta = 2\alpha, \tilde{B} = 2\tilde{A}, \sigma_B^2 = 4\sigma_A^2 \),

\[ \mathcal{E} \cos 2\tilde{A} = \int_{-\infty}^{\infty} (\cos 2\alpha)p_A(\alpha) d\alpha = \int_{-\infty}^{\infty} (\cos \beta) \frac{2 \exp(-\beta^2/2\sigma_A^2)}{\sigma_B\sqrt{2\pi}} d\beta/2 = e^{-\sigma_A^2/2}. \]

Thus, since \( \sigma_B^2/2 = 2\sigma_A^2 \), the expected value of (128) is

\[ \mathcal{E} \begin{bmatrix} +\cos 2A & \sin 2A \\ \sin 2A & -\cos 2A \end{bmatrix} = e^{-2\sigma_A^2} \begin{bmatrix} +\cos 2A & \sin 2A \\ \sin 2A & -\cos 2A \end{bmatrix}. \]  

(129)

Substituting the above result into the expected value of (127) yields

\[ \mathcal{E} \begin{bmatrix} \cos^2 A & \cos A \sin A \\ \cos A \sin A & \sin^2 A \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + e^{-2\sigma_A^2} \begin{bmatrix} +\cos 2A & \sin 2A \\ \sin 2A & -\cos 2A \end{bmatrix}. \]  

(130)

And using this result in (130), along with \( x/r = \cos a \) and \( y/r = \sin a \), and the double angle identities for the cosine and sine functions, gives
\[
\mathcal{E}\left[\begin{array}{cc}
\cos^2 A & \cos A \sin A \\
\cos A \sin A & \sin^2 A
\end{array}\right] = \frac{1-e^{-2\sigma_i^2}}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{e^{-2\sigma_i^2}}{r^2} \begin{bmatrix} x^2 & xy \\ xy & y^2 \end{bmatrix}.
\tag{131}
\]

Also, \(\sigma_i^2 = \mathcal{E}R^2 - r^2\). And so the matrix defined by (126) is

\[
\mathcal{E}X'X'^T = (\sigma_i^2 + r^2) \left( \frac{1-e^{-2\sigma_i^2}}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{e^{-2\sigma_i^2}}{r^2} \begin{bmatrix} x^2 & xy \\ xy & y^2 \end{bmatrix} \right).
\tag{132}
\]

Therefore, using (125) and (132) in (124), the covariance matrix of \(X'\) is

\[
\Sigma_{X'} = (\sigma_i^2 + r^2) \left( \frac{1-e^{-2\sigma_i^2}}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{e^{-2\sigma_i^2}}{r^2} \begin{bmatrix} x^2 & xy \\ xy & y^2 \end{bmatrix} \right) - e^{-2\sigma_i^2} \begin{bmatrix} x^2 & xy \\ xy & y^2 \end{bmatrix}.
\tag{133}
\]

Note that \(\Sigma_{X'}\) is a function of \(x\). And so, more explicitly, write

\[
\Sigma_{X'}(x) = \begin{bmatrix}
(\Sigma_{X'}^-)_{xx}(x, y) & (\Sigma_{X'}^-)_{xy}(x, y) \\
(\Sigma_{X'}^-)_{yx}(x, y) & (\Sigma_{X'}^-)_{yy}(x, y)
\end{bmatrix}.
\tag{134}
\]

Alternatively, using \(x = h^{-1}(r)\), one may write

\[
\Sigma_{X'}(r) = \begin{bmatrix}
(\Sigma_{X'}^-)_{xx}(r, a) & (\Sigma_{X'}^-)_{xy}(r, a) \\
(\Sigma_{X'}^-)_{yx}(r, a) & (\Sigma_{X'}^-)_{yy}(r, a)
\end{bmatrix}.
\tag{135}
\]

In (134) and (135) the subscripts on the \(\Sigma_{X'}^-\)'s index the components of the matrix; and the arguments serve to specify both the parameterization and the point at which the matrices are being evaluated.

Both forms are needed for the sequel. And so, given (133), the component form of (135) is determined as follows. First, rewrite (132) as

\[
\frac{r^2\mathcal{E}X'X'^T}{e^{\sigma_i^2}(\sigma_i^2 + r^2)} = \frac{e^{\sigma_i^2}}{2} \begin{bmatrix} x^2 + y^2 & 0 \\ 0 & x^2 + y^2 \end{bmatrix} + \frac{e^{-\sigma_i^2}}{2} \begin{bmatrix} 2x^2 & 2xy \\ 2xy & 2y^2 \end{bmatrix} - \frac{e^{-2\sigma_i^2}}{2} \begin{bmatrix} x^2 + y^2 & 0 \\ 0 & x^2 + y^2 \end{bmatrix}.
\]

Then use \(\cosh \sigma = (e^{\sigma} + e^{-\sigma})/2\) and \(\sinh \sigma = (e^{\sigma} - e^{-\sigma})/2\) to obtain
\[
\frac{r^2EXX'^T}{e^{\sigma_x^2}(\sigma_x^2 + r^2)} = \chi^3 \begin{bmatrix}
\cosh \sigma_A^2 & 0 \\
0 & \sinh \sigma_A^2
\end{bmatrix} + \chi^3 \begin{bmatrix}
\sinh \sigma_A^2 & 0 \\
0 & \cosh \sigma_A^2
\end{bmatrix} + e^{-\sigma_i^2} \begin{bmatrix}
0 & xy \\
xy & 0
\end{bmatrix}.
\]

And use \( \cos a = x/r \) and \( \sin a = y/r \) to obtain the components of \( EXX'^T \) as

\[
EXX' = (\sigma_r^2 + r^2)e^{-\sigma_i^2}(\cos^2 a \cosh \sigma_A^2 + \sin^2 a \sinh \sigma_A^2)
\]
\[
EYY' = (\sigma_r^2 + r^2)e^{-\sigma_i^2}(\cos^2 a \sinh \sigma_A^2 + \sin^2 a \cosh \sigma_A^2)
\]
\[
EXY' = EYX' = (\sigma_r^2 + r^2)e^{-2\sigma_i^2} \cos a \sin a.
\]

Also, rewrite (125) as

\[
\mu_X \mu_X^T = r^2 e^{-\sigma_i^2} \begin{bmatrix}
\cos^2 a & \cos a \sin a \\
\cos a \sin a & \sin^2 a
\end{bmatrix}.
\]

Substitution of these last two expressions into \( \Sigma_X = EXX'^T - \mu_X \mu_X^T \) yields

\[
(\Sigma_X)_{xx}(r,a) = \sigma_r^2 e^{-\sigma_i^2}(\cos^2 a \cosh \sigma_A^2 + \sin^2 a \sinh \sigma_A^2)
\]
\[
+r^2 e^{-\sigma_i^2}[\cos^2 a(\cosh \sigma_A^2 - 1) + \sin^2 a \sinh \sigma_A^2]
\]
\[
(\Sigma_X)_{xy}(r,a) = (\Sigma_X)_{yx}(r,a) = [\sigma_r^2 + r^2(1-e^{+\sigma_i^2})]e^{-2\sigma_i^2} \cos a \sin a
\]
\[
(\Sigma_X)_{yy}(r,a) = \sigma_r^2 e^{-\sigma_i^2}(\cos^2 a \sinh \sigma_A^2 + \sin^2 a \cosh \sigma_A^2)
\]
\[
+r^2 e^{-\sigma_i^2}[\cos^2 a \sinh \sigma_A^2 + \sin^2 a(\cosh \sigma_A^2 - 1)].
\]

Note that (138) was given in [20]. Here its derivation has been provided, via (133).

### 4.2 The Estimation Errors of the PLKF and CLKF

The sample mean basically provides an unbiased estimate of the expected value of \( X' \), and the sample covariance matrix gives an estimate of the covariance matrix of \( X' \) [60].

But \( X' \) is biased as an estimator of \( x \). Indeed, by definition, the bias vector and mean-squared error matrix of \( X' \) as an estimator of \( x \) are respectively

\[
b_{X'} = \varepsilon(X' - x) \text{ and } \text{mse} X' = \varepsilon(X' - x)(X' - x)^T.
\]

Obviously, since \( \varepsilon X' = \mu_{X'} \),

\[
b_{X'} = \mu_{X'} - x
\]
And, since \( X' - x = X' - \mu_{x'} + \mu_{x'} - x = X' - \mu_{x'} + b_{x'} \),

\[
\text{mse } X' = \Sigma_{x'} + b_{x'}^T b_{x'}. \tag{141}
\]

Also, using \( \Sigma_{x'} = \mathcal{E} X' (X')^T - \mu_{x'} \mu_{x'}^T \),

\[
\text{mse } X' = \mathcal{E} X' (X')^T + b_{x'} b_{x'}^T - \mu_{x'} \mu_{x'}^T. \tag{142}
\]

Thus, when (122) is valid the bias vector of \( X' \) as an estimator of \( x \) is

\[
b_{x'} = (e^{-\sigma_{t}^2/2} - 1)x. \tag{143}
\]

And, using (142) with

\[
b_{x'} b_{x'}^T - \mu_{x'} \mu_{x'}^T = (1 - e^{-\sigma_{t}^2/2})^2 xx^T - e^{-\sigma_{t}^2} xx^T = (1 - 2e^{-\sigma_{t}^2/2}) xx^T,
\]

the mean squared error matrix of \( X' \) as an estimator of \( x \) is determined to be

\[
\text{mse } X' = (\sigma_{r}^2 + r^2)\left[\left(1 - e^{-2\sigma_{t}^2}\right)/2 + xx^T \left(e^{-2\sigma_{t}^2}/r^2\right)\right] + xx^T \left(1 - 2e^{-\sigma_{t}^2/2}\right). \tag{144}
\]

### 4.2.1 The Errors of the PLKF

Given the above bias vector and mean-squared error matrix of \( X' \) as an estimator of \( x \), the corresponding ones of the PLKF are determined as follows. (For now, just the equi-weighted scalar-weight case shall be used, and the superscripts “(s)” will be dropped.)

First, consider the sequence of estimates, \( \hat{x}'_n = (1/n) \sum_{m=1}^{n} \hat{x}'_m \); and let \( \hat{X}'_n \) and \( \hat{X}'(n) \), \( n = 1, 2, \cdots, N \), respectively denote the underlying random vectors of \( \hat{x}'_n \) and \( \hat{x}' \). The mean and variance of the \( \hat{X}'(n) \) are [60]

\[
\mu_{\hat{X}_n} = \mu_{x'} \quad \text{and} \quad \Sigma_{\hat{X}_n} = \Sigma_{X'}/n \tag{145}
\]

(since the underlying random vectors of the radar measurements are mutually independent, the \( \hat{X}_n = h^{-1} (\hat{R}_n) \)'s are also mutually independent). Thus, using (141) as an identity, the bias vector and mean-squared error matrix of \( \hat{X}'(n) \) as an estimator of \( x \) are
\[ \mathbf{b}_{\hat{X}^{(n)}} = \mathbf{b}_X \] and \[ \text{mse } \hat{X}^{(n)} = \Sigma_X/n + \mathbf{b}_X \mathbf{b}_X^T. \] (146)

Note that the PLKF remains biased as an estimator of \( \mathbf{x} \) when \( n \to \infty \).

### 4.2.2 The Errors of the C-LKF

Next, consider the C-LKF as an estimator of \( \mathbf{x} \). And let \( \hat{\mathbf{R}}(n) \) denote the underlying random vector of \( \hat{\mathbf{r}}_n = (1/n) \sum_{m=1}^{n} \mathbf{r}_m \). The corresponding random vector of the C-LKF is then \( \hat{\mathbf{X}}(n) = \mathbf{h}^{-1}(\hat{\mathbf{R}}(n)) \). But since \( \hat{\mathbf{R}}(n) \) is distributed as \( \mathcal{N}(\mathbf{r}; \Sigma_{\mathbf{R}}/n) \) [60], the distribution of \( \hat{\mathbf{X}}(n) \) has the same form as the distribution of \( \mathbf{X}' \), except with \( \sigma_R^2 \) and \( \sigma_A^2 \) replaced by \( \sigma_{R(n)}^2 = \sigma_R^2/n \) and \( \sigma_{A(n)}^2 = \sigma_A^2/n \). In particular, given (122) and (133),

\[ \mathbf{\mu}_{\hat{X}^{(n)}} = e^{-\sigma_A^2/2n} \mathbf{x} \] (147)

and

\[ \Sigma_{\hat{X}^{(n)}} = \left( r^2 + \sigma_R^2/n \right) \begin{pmatrix} 1 - e^{-2\sigma_A^2/n} & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & x^2 & x y & x y \\ 0 & 0 & x y & y^2 & x y \\ 0 & 0 & x y & x y & y^2 \end{pmatrix} - e^{-\sigma_A^2/n} \begin{pmatrix} x^2 & x y & x y \\ y x & y^2 & y x \\ x y & y^2 & y^2 \end{pmatrix}. \] (148)

Therefore, the estimation bias vector and mean-squared error matrix of the C-LKF as an estimator of \( \mathbf{x} \) are

\[ \mathbf{b}_{\hat{X}^{(n)}} = (e^{-\sigma_A^2/2n} - 1) \mathbf{x} \]

and

\[ \text{mse } \mathbf{X}' = (r^2 + \sigma_R^2/n) \left[ (1 - e^{-2\sigma_A^2/n})/2 + \mathbf{x} \mathbf{x}^T (e^{-\sigma_A^2/n})/r^2 \right] + \mathbf{x} \mathbf{x}^T (1 - 2e^{-\sigma_A^2/2n}). \] (149)

Note that, like the PLKF, the C-LKF is a biased estimator of \( \mathbf{x} \). But, unlike the PLKF, the C-LKF is unbiased as \( n \to \infty \).
4.3 The Popular PLKF “Debiasing” Methods

In the seminal paper on the Debiased Consistent Converted Measurements (DCCM) method [20] it was proposed that the rectangular pseudo-measurements be first “debiased” before being used in a PLKF. And there an additive correction was specified. Later (for reasons to be given below) a multiplicative one was recommended by others [22, 23], called the Unbiased Consistent Converted Measurements (UCCM) method. Those DCCM and UCCM proponents also recommended that the true covariance matrix of the rectangular pseudo-measurement be used, $\Sigma_{X'}$ instead of $J^{-1}(\bar{r})\Sigma_{R}J^{-T}(\bar{r})$.

The DCCM and the UCCM “debiasing” operations are basically

\[
\bar{x}'_{n}(b) \equiv \bar{x}'_{n} - b_{X'} \quad \text{and} \quad \bar{x}'_{n}(\lambda) \equiv (1/\lambda)\bar{x}'_{n}
\]  

(here the argument “$b$” serves to denote the DCCM case, and the argument “$\lambda$” serves to denote the UCCM case). And, using (150), the “debiased” PLKF estimates are

\[
\hat{x}'_{n}(s)(b) \equiv \frac{1}{n} \sum_{m=1}^{n} \bar{x}'_{m}(b) \quad \text{and} \quad \hat{x}'_{n}(s)(\lambda) \equiv \frac{1}{n} \sum_{m=1}^{n} \bar{x}'_{m}(\lambda).
\]  

Figure 16 illustrates the effectiveness of the DCCM and UCCM methods for the scalar-weight case, using $\sigma_{\lambda} = \pi/6$ in $b_{X'} = (\lambda - 1)x$ and $\lambda = e^{-\sigma_{\lambda}/2}$. (Here the same measurement sets of the previous Chapter are being reused, partitioned into 50 mutually independent Monte Carlo trials.) And Figure 17 provides the sample means and “confidence intervals.” Note that the corresponding sample means are mostly indistinguishable; and that the UCCM standard deviations are slightly larger than the DCCM ones – given (150), the true covariance matrices of the “debiased” measurements in the DCCM and UCCM cases are respectively $\Sigma_{X'}$ and $(1/\lambda^{2})\Sigma_{X'}$, where $\lambda = e^{-\sigma_{\lambda}/2}$. 

55
Figure 16 “Debiased” scalar-weight PLKF estimates

Figure 17 Accuracy of the “debiased” scalar-weight PLKF estimates
Unfortunately, such simplistic “debiasing” can lead to worse results in the matrix-weight case. Recall the definition of the matrix-weight PLKF:

\[
\hat{x}^{(m)}_n = \hat{X}^{(m)}_n \sum_{m=1}^{n} \bar{X}^{-1}_m \bar{x}_m \quad \text{and} \quad \hat{X}^{(m)}_n = \left( \sum_{m=1}^{n} \bar{X}^{-1}_m \right)^{-1},
\]

(152)

where \( \bar{x} = h^{-1}(\bar{r}) \) and \( \bar{X} = J^{-1}(\bar{r}) \Sigma_J J^T(\bar{r}) \). If \( \bar{x}'(b) = \bar{x} - b \) is used in \( \bar{X}' \), that is,

\[
\bar{X}'_n(b) \equiv J^{-1}(\bar{x}'(b)) \Sigma_J J^T(\bar{x}'(b)) ,
\]

(153)

then

\[
\hat{x}^{(m)}_n(b) = \hat{X}^{(m)}_n(b) \sum_{m=1}^{n} \bar{X}^{-1}_m(b) \bar{x}'(b) \tag{154}
\]

and

\[
\hat{X}^{(m)}_n(b) = \left( \sum_{m=1}^{n} \bar{X}^{-1}_m(b) \right)^{-1}.
\]

(155)

And if \( \bar{x}'(\lambda) \equiv (1/\lambda)\bar{x}' \) is used in \( \bar{X}' \), that is,

\[
\bar{X}'_n(\lambda) \equiv J^{-1}(\bar{x}'(\lambda)) \Sigma_J J^T(\bar{x}'(\lambda)) ,
\]

(156)

then

\[
\hat{x}^{(m)}_n(\lambda) = \hat{X}^{(m)}_n(\lambda) \sum_{m=1}^{n} \bar{X}^{-1}_m(\lambda) \bar{x}'(\lambda) \tag{157}
\]

and

\[
\hat{X}^{(m)}_n(\lambda) = \left( \sum_{m=1}^{n} \bar{X}^{-1}_m(\lambda) \right)^{-1}.
\]

(158)

Figure 18 illustrates the sample means for the two “debias” matrix-weight PLKF cases, respectively defined by (154) and (155), and by (156) and (157). (The corresponding C-LKF estimates are also shown for reference).
Figure 18 “Debiased” matrix-weight PLKF biases

Of course, the DCCM and UCCM proponents also say that the true covariance matrices should be used, $\Sigma_{\chi',n}$, not $J^{-1}(\bar{r}_n)\Sigma_{\chi}J^{-T}(\bar{r}_n)$. Unfortunately, $\Sigma_{\chi',n}$ is a function of the unknown being estimated, $\chi$, and so in practice some approximation must be used instead. For example: given $\bar{r}$, use $\Sigma_{\chi'}(\bar{r})$; or, given $\hat{\chi}$, use $\Sigma_{\chi'}(\hat{\chi})$. Here these two cases shall be distinguished by respectively calling them measurement-based and estimate-based PLKF’s, written MB-PLKF and the EB-PLKF. (Note that since the basic PLKF uses $J^{-1}(\bar{r})\Sigma_{\chi}J^{-T}(\bar{r})$, it too is an MB-PLKF.) More unfortunate, however, the $b_{\chi',n}$ that the DCCM uses also depends upon $\chi$ – the ideal DCCM was actually being illustrated above, with $b_{\chi',n} = (\lambda - 1)\chi$. In contrast, the $\lambda$ in the UCCM depends only on $\sigma_A$. And so only the more practical UCCM shall be considered further in the sequel.

Accordingly, let the MB-PLKF use $\bar{x}'_{\chi,n}(\lambda)$ in $\Sigma_{\chi',n}$, $n = 1, 2, \cdots, N$, and determine
\[ \hat{X}^{(m)}_n(\lambda) = \hat{X}^{(m)}_n(\lambda) \sum_{m=1}^{n} \Sigma_{X^{-1}}(\lambda) \Sigma_{X}^{-1}(\lambda) \] (159)

and

\[ \hat{X}^{(m)}_n(\lambda) = \left( \sum_{m=1}^{n} \Sigma_{X^{-1}}(\lambda) \right)^{-1}. \] (160)

And, alternatively, let the EB-PLKF use \( \hat{X}^{(m)}_{n-1}(\lambda) \) in \( \Sigma_{X'} \), \( n = 2,3,\ldots,N \), and determine

\[ \hat{X}^{(m)}_n(\lambda) = \hat{X}^{(m)}_n(\lambda) \left[ \left( \hat{X}^{(m)}_{n-1}(\lambda) \right)^{-1} \hat{X}^{(m)}_{n-1}(\lambda) + \Sigma_{X'}(\hat{X}^{(m)}_{n-1}(\lambda)) \Sigma_{X}^{-1}(\lambda) \right] \] (161)

and

\[ \hat{X}^{(m)}_n(\lambda) = \left[ \left( \hat{X}^{(m)}_{n-1}(\lambda) \right)^{-1} + \Sigma_{X'}(\hat{X}^{(m)}_{n-1}(\lambda)) \right]^{-1}, \] (162)

with \( \hat{x}^{(m)}_i(\lambda) = \hat{x}^{(m)}_i(\lambda) \) and \( \hat{X}^{(m)}_i(\lambda) = \hat{X}^{(m)}_i(\lambda). \) [Between (159) and (160), the common factor \( 1/\lambda^2 \) has been tacitly canceled – also between (161) and (162).] Figure 19 shows these two “debiased and consistent” PLKF cases, along with “biased” C-LKF case.

Figure 19 Matrix-weight UCCM with “consistent” covariance matrices
4.4 Discussion of the “debiased” PLKF

The above presentation is summarized as follows. The PLKF is a biased estimator of $\mathbf{x}$ as $N \to \infty$, while the C-LKF is asymptotically unbiased. The (ideal) DCCM “debiases” the PLKF by subtracting the true estimation bias from the (unbiased) rectangular pseudo-measurements; and the (practical) UCCM “debiases” the PLKF by making those measurements noisier. And both “debiased” PLKF’s are much noisier than the C-LKF.

Figure 20 shows the sample means and “confidence intervals” for the estimates that were shown earlier in Figure 17, along with the true standard deviations of the DCCM and UCCM cases – the dashed curves. There the DCCM standard deviations are seen to be smaller than those of the UCCM – the DCCM covariance matrix is $\Sigma_{\mathbf{x}'}$, given by (133), while the UCCM covariance matrix is $(1/\lambda^2)\Sigma_{\mathbf{x}'}$. The corresponding curves for the C-LKF are also shown in the figure (the darker curves).

![Figure 20 Accuracies of the “debiased” PLKF’s (scalar-weight cases)](image-url)
4.5 The Residual Biases of the “Debiased and Consistent” PLKF

Unfortunately, the “debiased and consistent” matrix-weight PLKF’s have residual biases. The reason is the same as why the bias of the matrix-weight PLKF has the opposite sense to the bias of the scalar-weight PLKF.

Consider the basic matrix-weight PLKF, but written as

\[ \hat{X}_{n}^{-1}\tilde{x}_{n} = \sum_{m=1}^{n} \tilde{X}_{m}^{-1}\tilde{x}_{m} \quad \text{and} \quad \hat{X}_{n}^{-1} = \sum_{m=1}^{n} \tilde{X}_{m}^{-1}. \]  

(163)

Dropping the adornments, the functional form of the product \( X^{-1}x = J^T(x)\Sigma^{-1}_R J(x)h^{-1}(r) \) is determined as follows. First, the component form of the product \( J(x)h^{-1}(r) \) is

\[
\begin{bmatrix}
\cos a & \sin a \\
-sin a/r & \cos a/r
\end{bmatrix}
\begin{bmatrix}
r cos a \\
r sin a
\end{bmatrix} = \begin{bmatrix}
r \\
0
\end{bmatrix}.
\]

(164)

And, given \( \Sigma_R \), symmetric and positive definite, the component form of \( \Sigma^{-1}_R Jh^{-1} \) is

\[
\begin{bmatrix}
1/\sigma_R^2 & 0 \\
0 & 1/\sigma_A^2
\end{bmatrix}
\begin{bmatrix}
\cos a & \sin a \\
-sin a/r & \cos a/r
\end{bmatrix}
\begin{bmatrix}
r cos a \\
r sin a
\end{bmatrix} = \begin{bmatrix}
r/\sigma_R^2 \\
0
\end{bmatrix}.
\]

(165)

Multiplying this result by \( J^T \) yields

\[
\begin{bmatrix}
\cos a & -\sin a/r \\
\sin a & \cos a/r
\end{bmatrix}
\begin{bmatrix}
r/\sigma_R^2 \\
0
\end{bmatrix} = \frac{1}{\sigma_R^2} \begin{bmatrix}
r cos a \\
r sin a
\end{bmatrix}.
\]

(166)

Thus,

\[ X^{-1}x = (1/\sigma_R^2)x. \]

(167)

Note that in (167) the effective weight of \( X^{-1}x \) is independent of \( \sigma_A^2 \). But \( X^{-1} \) is

\[
(J^{-1}\Sigma_R J^T)^{-1} = J^T\Sigma^{-1}_R J = O^T D^{-1} \Sigma^{-1}_R D^{-1} O
\]

(168)

– see (17) and (18) in Chapter 2. And its component form is
\[
X^{-1} = \begin{bmatrix}
\cos^2 a / \sigma^2_R + \sin^2 a / (r \sigma_A)^2 & + \cos a \sin a / \sigma^2_R - \cos a \sin a / (r \sigma_A)^2 \\
+ \cos a \sin a / \sigma^2_R - \cos a \sin a / (r \sigma_A)^2 & \sin^2 a / \sigma^2_R + \cos^2 a / (r \sigma_A)^2
\end{bmatrix},
\]

which, for convenience, is written as

\[
X^{-1} = X_R^{-1} + X_A^{-1},
\]

with

\[
X_R^{-1} \equiv \frac{1}{\sigma^2_R} \begin{bmatrix}
\cos^2 a & \cos a \sin a \\
\cos a \sin a & \sin^2 a
\end{bmatrix}
\]

and

\[
X_A^{-1} \equiv \frac{1}{r^2 \sigma^2_A} \begin{bmatrix}
\sin^2 a & -\cos a \sin a \\
-\cos a \sin a & \cos^2 a
\end{bmatrix}
\]

(the notation here is a contrivance – both \(X_R^{-1}\) and \(X_A^{-1}\) are singular). Thus the effective weight matrix of \(X^{-1}x\) is inconsistent with the weight of \(X^{-1}\).

Consider the recursive update form of the basic MB-PLKF (without “debiasing”),

\[
\hat{x}'_n = \hat{x}'_{n-1} + \hat{X}'_n \tilde{X}'_{n-1} (\tilde{x}'_n - \hat{x}'_{n-1}), \tag{169}
\]

where

\[
\tilde{X}'_n \equiv \tilde{X}'(\tilde{r}_n) = J^{-1}(\tilde{r}_n) \Sigma_R J^{-T}(\tilde{r}_n). \tag{170}
\]

Given (167) with \(\tilde{X}'_n = \tilde{X}'(\tilde{r}_n)\), the weighted measurement is \(\tilde{X}'_{n-1} \tilde{x}'_n = (1/\sigma^2_R) \tilde{X}_n\), and so in (169) the product of the gain matrix and the measurement, \(\hat{X}'_n \tilde{X}'_{n-1} \tilde{x}'_n\), is \(\hat{X}'_n \tilde{x}'_n / \sigma^2_R\). But the product of the MB-gain matrix and the prior estimate is \(\hat{X}'_n \tilde{x}'_{n-1} \hat{x}'_{n-1}\). Similarly, in the basic EB-PLKF, since \(\tilde{X}'_n \equiv \tilde{X}'(\tilde{x}'_{n-1}) = J^{-1}(\tilde{x}'_{n-1}) \Sigma_R J^{-T}(\tilde{x}'_{n-1})\), the product of the gain matrix and the prior estimate is \(\hat{X}'_n \tilde{x}'_{n-1} / \sigma^2_R\), while the product of the gain matrix and the measurement is \(\hat{X}'_n \tilde{x}'_{n-1} \tilde{x}'_{n-1}\). Thus, in both the MB-PLKF and EB-PLKF, the effective
gains that respectively operate on the given measurement and the prior estimate are “inconsistent” because of the functional dependency between the realization and its transformed associated covariance matrix.

Fortunately, a simple remedy for the unmodeled correlation problem exists when $n > 1$: instead of $\mathbf{X}'(\mathbf{x}_n') = J^{-1}(\mathbf{x}_{n-1}')\Sigma_n J^{-T}(\mathbf{x}_n')$, simply use

$$\mathbf{X}'_n \equiv \mathbf{X}'(\mathbf{x}_{n-1}') = J^{-1}(\mathbf{x}_{n-1}')\Sigma_n J^{-T}(\mathbf{x}_{n-1}') \quad (171)$$

That is, let

$$\hat{\mathbf{x}}'_n = \hat{\mathbf{x}}'_n - \sum_{m=2}^{n} \left[ \mathbf{X}'(\mathbf{x}_{m-1}') \right]^{-1} \mathbf{x}_m \quad (172)$$

and

$$\hat{\mathbf{x}}'_{n-1}^{-1} = \left[ \mathbf{X}'(\mathbf{x}_n') \right]^{-1} + \sum_{m=2}^{n} \left[ \mathbf{X}'(\mathbf{x}_{m-1}') \right]^{-1} \quad (173)$$

In which case the effective gains used on the measurement and prior estimate become “consistent” as $n$ increases – for small $n$ there is still a problem, because $\hat{\mathbf{x}}_i = \mathbf{x}_i'$.

Figure 21 illustrates the effectiveness of using (171) in the basic PLKF matrix-weight update, along with the basic scalar-weight case. Also shown in the figure are $\mu_{x'}$ and $\mu_{y'}$, the expected values of the rectangular pseudo-measurements (the dark lines). Figure 22 provides the corresponding sample means and “confidence intervals” (there the solid curves are the matrix-weight case, and the dashed curves are the scalar-weight case).

This shows that the remedy given by (171) comes at a cost: the estimates appear to be more noisy. Obviously, since in (172), when $m > 1$ each summand is the product of a random matrix and an independent random vector. Recall that the variance of a product
of mutually independent random variables is the sum of the variances of the respective variables. And so these PLKF estimates are noisier than the basic matrix-weight ones.

Figure 21 Comparison of the “prior-weight” MB-PLKF with the basic scalar-weight PLKF

Figure 22 Accuracies of prior-matrix weight and scalar-weight PLKF’s
The functional dependency between certain realizations and their transformed associated covariance matrices in the MB-PLKF and EB-PLKF cases is a correlation that is not modeled in the DCCM and UCCM methods. In the “debiased and consistent” MB-PLKF UCCM case \( \Sigma_{X'}^{-1}(\tilde{x}_n' (\lambda)) \) and \( \tilde{x}_m' (\lambda) \) are correlated; and in the “debiased and consistent” EB-PLKF UCCM case \( \Sigma_{X'}^{-1}(\tilde{x}_{n,-1}' (\hat{\lambda})) \) and \( \tilde{x}_m' (\hat{\lambda}) \) are correlated. But, using the above remedy, \( \overline{X}'^{-1}(\tilde{x}_{n,-1}') = \Sigma_{X'}^{-1}(\tilde{x}_{n,-1}' (\hat{\lambda})) \), the effective gains that operate upon the measurement and prior estimate can become the same (as \( n \) increases). Of course, this (asymptotic) remedy comes at a cost: the UCCM estimates, which were already noisier by \( 1/\lambda \), are made even noisier. In the sequel the prior-weight “debiased” and now consistent UCCM measurement-based case shall be used for the illustrations.
5 Analysis of the EKF Update

In Chapter 3 it was shown that the EKF can perform better than a PLKF. Here the EKF is discussed more formally; and some analytic results are obtained for use in the sequel.

In this Chapter the alternate form of the EKF update equations shall be used, and the primes and indices on $\hat{x}_n'$ and $\hat{X}_n'$ will be dropped. That is, given a prior estimate of $x$, and its associated covariance matrix, written $(\hat{x}^-; \hat{X}^-)$, the EKF update equations are

$$\hat{x} = \hat{x}^- + K [\bar{r} - h(\hat{x}^-)] \quad \text{and} \quad \hat{X} = [(\hat{X}^-)^{-1} + J^T(\hat{x}^-)\hat{X}^-J(\hat{x}^-)]^{-1},$$

with

$$K = \hat{X}J^T(\hat{x}^-)\hat{X}^-.$$  \hfill (175)

As before, $R$ is the underlying random vector of $\bar{r}$, distributed as $\mathcal{N}(\bar{r}; \Sigma_R)$; and the underlying random variables of the range and azimuth measurements, $R$ and $A$, are mutually independent. But now an underlying random vector for $\hat{x}^-$ is also invoked, denoted $X^-$. Its distribution is arbitrary, except for its covariance matrix, which is assumed to exist and be positive definite; and $X^-$ and $R$ are tacitly assumed to be mutually independent.

Note that $R$ and $A$ mutually independent implies $\Sigma = \Sigma_R + \Sigma_A$

$$\Sigma_R = \begin{bmatrix} \sigma_R^2 & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \Sigma_A = \begin{bmatrix} 0 & 0 \\ 0 & \sigma_A^2 \end{bmatrix}, \hfill (176)$$

and that

$$\bar{X} = J^{-1}(\hat{r}^-)(\Sigma_R + \Sigma_A)J^T(\hat{r}^-) = \bar{X}_R + \bar{X}_A,$$ \hfill (177)

where

$$\bar{X}_R \equiv J^{-1}(\hat{r}^-)\Sigma_R J^T(\hat{r}^-) \quad \text{and} \quad \bar{X}_A \equiv J^{-1}(\hat{r}^-)\Sigma_A J^T(\hat{r}^-).$$ \hfill (178)
5.1 The EKF Update Errors

Now given $(\hat{x}^-; \hat{X}^-)$ and $R$, determine $\hat{X}$ by the second expression in (174). And then define the conditional random vector

$$X_{(\hat{x}^-; \hat{X}^-)} = \hat{x}^- + \hat{X}^T(\hat{x}^-)\Sigma_R^{-1} \left[ R - h(\hat{x}^-) \right].$$  

(179)

Note that $\hat{X}$ is conditionally deterministic: it depends only on $(\hat{x}^-; \hat{X}^-)$ and $\Sigma_R$. Thus, $X_{(\hat{x}^-; \hat{X}^-)}$ is conditionally gaussian [47]. Indeed, $X_{(\hat{x}^-; \hat{X}^-)} \sim \mathcal{N}(\mu_{x(\hat{x}^-; \hat{X}^-)}; \Sigma_{x(\hat{x}^-; \hat{X}^-)})$, where

$$\mu_{x(\hat{x}^-; \hat{X}^-)} = \hat{x}^- + \hat{X}^T(\hat{x}^-)\Sigma_R^{-1} \left[ \mu_R - h(\hat{x}^-) \right]$$

(180)

and

$$\Sigma_{x(\hat{x}^-; \hat{X}^-)} = \hat{X}^T(\hat{x}^-)\Sigma_R^{-1} I(\hat{x}^-) \hat{X}^T.$$  

(181)

This last expression follows from $X_{(\hat{x}^-; \hat{X}^-)} - \mathcal{E}X_{(\hat{x}^-; \hat{X}^-)} = \hat{X}^T(\hat{x}^-)\Sigma_R^{-1} (R - \mathcal{E}R)$.

Unfortunately, as an estimator of $x$, (179) has an unmodeled linearization error. For example, consider the scalar-weight EKF update, defined here as

$$\hat{x}^{(s)} = \hat{x}^- + \frac{\hat{w}}{\hat{w}} J^{-1}(\hat{r}^-) \left( \hat{r}^- - h(\hat{x}^-) \right) \quad \text{and} \quad \hat{w} = \hat{w}^- + \hat{w},$$

(182)

where $\hat{X}^- \equiv I/\hat{w}^-$ and $\hat{X} \equiv I/\hat{w}$. This update is a linear (affine) relation in $\hat{r}$, with a nonlinear dependency upon $\hat{r}^-$. Together with $\hat{r}^- = h(\hat{x}^-)$, it leads to

$$\hat{w}(\hat{x}^{(s)} - \hat{x}^-) = \hat{w} J^{-1}(\hat{r}^-) \left( \hat{r}^- - \hat{r}^- \right),$$

(183)

a relation between weighted differentials.

Using $\hat{r}^- = [(\hat{x}^-)^2 + (\hat{y}^-)^2]^{1/2}$ and $\hat{a}^- = \arctan(\hat{x}^-, \hat{y}^-)$, in component form (182) is

$$\begin{bmatrix} \hat{x}^{(s)} \\ \hat{y}^{(s)} \end{bmatrix} = \begin{bmatrix} \hat{x}^- \\ \hat{y}^- \end{bmatrix} + \frac{\hat{w}}{\hat{w}} \begin{bmatrix} \cos \hat{a}^- & -\hat{r}^- \sin \hat{a}^- \\ \hat{r}^- \sin \hat{a}^- & \cos \hat{a}^- \end{bmatrix} \begin{bmatrix} \hat{r}^- - \hat{r}^- \\ \hat{a}^- - \hat{a}^- \end{bmatrix}.$$  

(184)
Equivalently,

\[
\begin{bmatrix}
\hat{x}^{(s)} \\
\hat{y}^{(s)}
\end{bmatrix} = \begin{bmatrix}
\hat{x}^- \\
\hat{y}^-
\end{bmatrix} + \frac{r_\Delta}{\bar{r}} \begin{bmatrix}
\hat{x}^- \\
\hat{y}^-
\end{bmatrix} + a_\Delta \begin{bmatrix}
-\hat{y}^- \\
+\hat{x}^-
\end{bmatrix},
\]

where

\[
\begin{align*}
\frac{r_\Delta}{\bar{r}} &= \frac{\bar{w}}{w} (\bar{r} - \hat{r}^-) \quad \text{and} \quad a_\Delta &= \frac{\bar{w}}{w} (\bar{a} - \hat{a}^-) .
\end{align*}
\]  

Using \( \hat{x}^{(s)} = \hat{r}^{(s)} \cos \hat{a}^{(s)} \) and \( \hat{y}^{(s)} = \hat{r}^{(s)} \sin \hat{a}^{(s)} \) in (182),

\[
\hat{r}^{(s)} \cos \hat{a}^{(s)} = (\hat{r}^- + r_\Delta) \cos \hat{a}^- - a_\Delta \hat{r}^- \sin \hat{a}^-
\]

and

\[
\hat{r}^{(s)} \sin \hat{a}^{(s)} = (\hat{r}^- + r_\Delta) \sin \hat{a}^- + a_\Delta \hat{r}^- \cos \hat{a}^- .
\]

leads to

\[
\hat{r}^{(s)} = \sqrt{(\hat{r}^- + r_\Delta)^2 + (a_\Delta \hat{r}^-)^2}
\]

and

\[
\hat{a}^{(s)} = \arctan \left[ \frac{(\hat{r}^- + r_\Delta) \sin \hat{a}^- + a_\Delta \hat{r}^- \cos \hat{a}^-}{(\hat{r}^- + r_\Delta) \cos \hat{a}^- - a_\Delta \hat{r}^- \sin \hat{a}^-} \right] .
\]

Now let \( a_\Delta = 0 \) in (189) and (190), with \( r_\Delta \) arbitrary. In which case

\[
\hat{r}^{(s)}_{a=\bar{a}} = \hat{r}^- + r_\Delta \quad \text{and} \quad \hat{a}^{(s)}_{a=\bar{a}} = \hat{a}^- .
\]

Note that this range update is exactly what the (optimal) LKF would determine – see Chapter 3. More important, it does not change the azimuth estimate, which is appropriate since the mutual independence of \( R \) and \( A \) implies that \( \bar{r} \) contains no information on \( a \). Alternatively, let \( r_\Delta = 0 \) with \( a_\Delta \) arbitrary. In which case (189) and (190) become

\[
\hat{r}^{(s)}_{\bar{r}} = \hat{r}^- \sqrt{1 + a_\Delta^2} \quad \text{and} \quad \hat{a}^{(s)}_{\bar{r}} = \arctan \left( \frac{\hat{y}^- + a_\Delta \hat{x}^-}{\hat{x}^- - a_\Delta \hat{y}^-} \right) .
\]
Here the range component is affected, which is inappropriate since \( \bar{a} \) contains no information on \( r \). Indeed, in (192), the range update is seen to be approximately \( \hat{r} + a_{\Delta} \hat{r}^{-1} \); and, rewriting the second expression in (192) as

\[
\hat{\theta}_{r=r}^{(x)} = \arctan \left( \frac{\sin \hat{\theta} + a_{\Delta} \cos \hat{\theta}}{\cos \hat{\theta} - a_{\Delta} \sin \hat{\theta}} \right),
\]

(193)

and then using the identity \( \alpha + \arctan b = \arctan [(\sin \alpha + b \cos \alpha)/(\cos \alpha - b \sin \alpha)] \), the azimuth update is seen to be approximately \( \hat{\theta} + a_{\Delta} + a_{\Delta}^3/3 \).

For the matrix-weight EKF case, use \( \bar{X}_R^{-1} \equiv J^T (\hat{x}^-) \Sigma_R^{-1} J(\hat{x}^-) \) and the identity \( J^T (\hat{r}^-) \Sigma_R^{-1} = J^T (\hat{x}^-) \Sigma_R^{-1} J(\hat{x}^-) J^{-1}(\hat{x}^-) \) to rewrite the first expression in (174) as

\[
\hat{x}^{(m)} = \hat{x}^- + \hat{X} \bar{X}_R^{-1} J^{-1}(\hat{r}^-) (\bar{r} - \hat{r}^-).
\]

(194)

As with (183), this update also determines a relation between weighted differentials,

\[
\hat{X}^{-1} (\hat{x}^{(m)} - \hat{x}^-) = \bar{X}_R^{-1} J^{-1}(\hat{r}^-) (\bar{r} - \hat{r}^-).
\]

(195)

Letting \( \hat{w}^- = \text{tr}(\hat{X}^-)^{-1} \) and \( \bar{w} \equiv \text{tr} \bar{X}_R^{-1} \), use the second expression in (174) to define

\[
\hat{w} \equiv \text{tr} \hat{X}^{-1} = \text{tr}(\hat{X}^-)^{-1} + \text{tr} \bar{X}_R^{-1}.
\]

(196)

That is, \( \hat{w} = \hat{w}^- + \bar{w} \). And then rewrite (194) as

\[
\hat{x}^{(m)} = \hat{x}^- + G \frac{\bar{w}}{\hat{w}} J^{-1}(\hat{r}^-) (\bar{r} - \hat{r}^-),
\]

(197)

where

\[
G \equiv (\hat{w}/\bar{w}) \hat{X} \bar{X}_R^{-1}.
\]

(198)

In which case, using (186),

\[
G^{-1} (\hat{x}^{(m)} - \hat{x}^-) = \frac{r_{\Delta}}{\hat{r}^-} \begin{bmatrix} \hat{x}^- \end{bmatrix} + a_{\Delta} \begin{bmatrix} -\hat{y}^- \+ \hat{x}^- \end{bmatrix}.
\]

(199)
Finally, comparing (185) and (199),

$$\mathbf{G}^{-1} (\hat{x}^{(m)} - \hat{x}^-) = \hat{x}^{(r)} - \hat{x}^-.$$  \hspace{1cm} (200)

But \( \mathbf{G} \) is independent of \( \mathbf{r} \). Thus, in both the scalar-weight and matrix-weight cases the EKF update introduces certain linearization errors that are not explicitly modeled by (179). When \( a_\Delta = 0 \) they are zero. But when \( a_\Delta \neq 0 \), the unmodeled range error is approximately \(|a_\Delta \hat{r}^-|\) and the unmodeled azimuth error is approximately \(a_\Delta^3 / 3\).

However, as seen in the first expression of (192), the range linearization errors are systematic (they are all positive). In contrast, the sign of the azimuth linearization errors is random – the expected value of their underlying random variable is zero.

5.2 The Preferred Ordering Theorem for the EKF

The findings of the last section lead directly to the Preferred Ordering Theorem (POT): given independent range and azimuth measurements, when using the EKF to update a radar track in rectangular coordinates, the measurements should be used recursively in the order azimuth first and range last [1, 31]. Such is outlined in the next two sections, and there some more notation is defined.

5.2.1 The POT in the Scalar-Weight EKF Case

Consider the EKF update defined by (184), but written as

$$\hat{x}_r^{(r)} = \hat{x}^- + \Delta_r^{(-)}(\mathbf{r})$$  \hspace{1cm} (201)

with

$$\Delta_r^{(-)}(\mathbf{r}) = \frac{\bar{w}}{w} \begin{bmatrix} \cos \hat{a}^- & -\hat{r}^- \sin \hat{a}^- \\ +\sin \hat{a}^- & \hat{r}^- \cos \hat{a}^- \end{bmatrix} \begin{bmatrix} \hat{r} - \hat{r}^- \\ \hat{a} - \hat{a}^- \end{bmatrix}. \hspace{1cm} (202)$$

Here the subscript "r" is used to indicate that the components of the measurement vector
are being used together – concurrently, not sequentially – and so (201) with (202) shall be
dubbed a vector-update. Of course, (201) and (202) may also be written together as
\[
\hat{x}_r^{(s)} = \hat{x}^- + \Delta_r^{(-)}(\bar{r}) + \Delta_a^{(-)}(\bar{a}) ,
\]
where
\[
\Delta_r^{(-)}(\bar{r}) = \frac{\bar{w}}{w} \left[ \cos \hat{\alpha}^- \right] (\bar{r} - \hat{r}^-) \quad \text{and} \quad \Delta_a^{(-)}(\bar{a}) = \frac{\bar{w}}{w} \left[ \frac{-\hat{r}^- \sin \hat{\alpha}^-}{\hat{r}^- \cos \hat{\alpha}^-} \right] (\bar{a} - \hat{a}^-) .
\]
Here the subscripts “\(r\)” and “\(a\)” are used to indicate that the scalar measurement
components are being used individually.

Next, define two scalar-updates as follows: either use \((\bar{r};1/\bar{w})\) first to update \(\hat{x}^-\), or
use \((\bar{a};1/\bar{w})\) first to update \(\hat{x}^-\). That is, determine either
\[
\hat{x}_r^{(s)} = \hat{x}^- + \Delta_r^{(-)}(\bar{r}) \quad \text{or} \quad \hat{x}_a^{(s)} = \hat{x}^- + \Delta_a^{(-)}(\bar{a}) .
\]
And then use the other measurement component, respectively as
\[
\hat{x}_r^{(s)} = \hat{x}_r^{(s)} + \Delta_a^{(-)}(\bar{a}) \quad \text{and} \quad \hat{x}_a^{(s)} = \hat{x}_a^{(s)} + \Delta_r^{(-)}(\bar{r}) .
\]
These are the two sequential scalar-updates corresponding to the above vector update:
(206) with (205) provides the same results as (203). If the above scalar-updates were
used recursively, however, the results would have been different [1, 31].

Define the range-first azimuth-last recursive update as follows. First use \((\bar{r};1/\bar{w})\)
with the first expression in (205), that is, determine \(\hat{x}_r^{(s)T} = (\hat{x}_r^{(s)}, \hat{x}_r^{(s)})\), which yields \(\hat{r}_r^{(s)}\)
and \(\hat{a}_r^{(s)}\). And then use \((\bar{a};1/\bar{w})\) with \(\hat{r}_r^{(s)}\) and \(\hat{a}_r^{(s)}\) in the second expression of (204),
\[
\Delta_a^{(-)}(\bar{a}) = \frac{\bar{w}}{w} \left[ \frac{-\hat{r}_r^{(s)} \sin \hat{a}_r^{(s)}}{\hat{r}_r^{(s)} \cos \hat{a}_r^{(s)}} \right] (\bar{a} - \hat{a}_r^{(s)}) ,
\]
to update \(\hat{x}_r^{(s)}\) as
\[ \hat{x}_{ra}^{(s)} = \hat{x}_r^{(s)} + \Delta_{ra}^{(s)}(\vec{a}) . \] (208)

This yields \( \hat{x}_{ra}^{(s)T} = (\hat{x}_{ra}^{(s)}, \hat{y}_{ra}^{(s)}) \), which determines \( \hat{r}_{ra}^{(s)} \) and \( \hat{a}_{ra}^{(s)} \). (The subscript “ra” is used to denote these outcomes as recursive range-first azimuth-last updates.)

Similarly, define the azimuth-first range-last recursive update as follows. First use \((\vec{a}; 1/\vec{w})\) with the second expression in (205), that is, \( \hat{x}_a^{(s)T} = (\hat{x}_a^{(s)}, \hat{y}_a^{(s)}) \), which yields \( \hat{r}_a^{(s)} \) and \( \hat{a}_a^{(s)} \). And then use \((\vec{r}; 1/\vec{w})\) with \( \hat{r}_a^{(s)} \) and \( \hat{a}_a^{(s)} \) in the first expression of (204),

\[ \Delta_{ar}^{-}(\vec{r}) = \frac{\vec{w}}{\vec{w}} \left[ \cos \hat{a}_a^{-} \right] (\vec{r} - \hat{r}_a^{-}) \] (209)

to update \( \hat{x}_a^{(s)} \) as

\[ \hat{x}_a^{(s)} = \hat{x}_a^{(s)} + \Delta_{ar}^{(s)}(\vec{r}) . \] (210)

This yields \( \hat{x}_{ar}^{(s)T} = (\hat{x}_{ar}^{(s)}, \hat{y}_{ar}^{(s)}) \), which in turn provides \( \hat{r}_{ar}^{(s)} \) and \( \hat{a}_{ar}^{(s)} \). Here the subscript “ar” is used to denote these outcomes as azimuth-first range-last recursive updates.

### 5.2.2 The POT in the Matrix-Weight EKF Case

Here the POT equations for the matrix-weight case are presented – and some new notation is defined for use in the sequel. Specifically, let

\[ e_{\parallel}(a) \equiv \begin{bmatrix} \cos a \\ \sin a \end{bmatrix} \quad \text{and} \quad e_{\perp}(a) \equiv \begin{bmatrix} \cos a \\ -\sin a \end{bmatrix} , \] (211)

and write the Jacobian matrices that were defined in Chapter 2 as

\[ J(p) = \begin{bmatrix} e_{\parallel}(a) \\ r^{-1}e_{\perp}(a) \end{bmatrix} \quad \text{and} \quad J^{-1}_r(p) = [e_{\parallel}(a) \quad re_{\perp}(a)] . \] (212)

And so write the matrix-weight EKF vector-update of \( \hat{x} \) as

\[ \hat{x}_r = \hat{x} - \hat{x}_r e_{\parallel}(\hat{a}) \left( \frac{\vec{r} - \hat{r}}{\sigma_r^2} \right) + \hat{r} \hat{x}_r e_{\perp}(\hat{a}) \left( \frac{\vec{a} - \hat{a}}{\sigma_a^2} \right) . \] (213)
Note that the differential form of the above expression is

$$\hat{X}_r^{-1}(\hat{x}_r - \hat{x}^-) = e_{\parallel}(\hat{a}^-) \left( \frac{\tilde{r} - \hat{r}}{\sigma_R^2} \right) + \hat{r} e_{\perp}(\hat{a}^-) \left( \frac{\tilde{a} - \hat{a}}{\sigma_\Delta^2} \right). \quad (214)$$

In the **sequential** matrix-update case, given \((\hat{x}^-; \hat{X}^-)\), either use \((\tilde{r};\sigma_R^2)\) first,

$$\hat{x}_r = \hat{x}^- + \hat{X}_r e_{\parallel}(\hat{a}^-) \left( \frac{\tilde{r} - \hat{r}}{\sigma_R^2} \right) \quad \text{and} \quad \hat{X}_r = \left[ (\hat{X}^-)^{-1} + \frac{1}{\sigma_R^2} e_{\parallel}(\hat{a}^-) e_{\parallel}^T(\hat{a}^-) \right]^{-1}. \quad (215)$$

Or, given \((\hat{x}^-; \hat{X}^-)\), use \((\tilde{a};\sigma_\Delta^2)\) first,

$$\hat{x}_a = \hat{x}^- + \hat{r} \hat{X}_a e_{\perp}(\hat{a}^-) \left( \frac{\tilde{a} - \hat{a}}{\sigma_\Delta^2} \right) \quad \text{and} \quad \hat{X}_a = \left[ (\hat{X}^-)^{-1} + \frac{(\hat{r})^2}{\sigma_\Delta^2} e_{\perp}(\hat{a}^-) e_{\perp}^T(\hat{a}^-) \right]^{-1}. \quad (216)$$

These two updates respectively imply

$$\hat{X}_r^{-1}(\hat{x}_r - \hat{x}^-) = e_{\parallel}(\hat{a}^-) \left( \frac{\tilde{r} - \hat{r}}{\sigma_R^2} \right) \quad \text{and} \quad \hat{X}_a^{-1}(\hat{x}_a - \hat{x}^-) = \hat{r} e_{\perp}(\hat{a}^-) \left( \frac{\tilde{a} - \hat{a}}{\sigma_\Delta^2} \right). \quad (217)$$

And the respective sums of these weighted differentials equals the one in (214). That is,

$$\hat{X}_r^{-1}(\hat{x}_r - \hat{x}^-) = \hat{X}_r^{-1}(\hat{x}_r - \hat{x}^-) + \hat{X}_a^{-1}(\hat{x}_a - \hat{x}^-). \quad (218)$$

Thus,

$$\hat{x}_r = \hat{x}^- + K_r \left( \hat{x}_r - \hat{x}^- \right) + K_a \left( \hat{x}_a - \hat{x}^- \right), \quad (219)$$

where

$$K_r = \hat{X}_r \hat{X}_r^{-1} \quad \text{and} \quad K_a = \hat{X}_a \hat{X}_a^{-1}. \quad (220)$$

That is, as in the scalar-weight case, the sequential-scalar update and the vector update provide the same result.
For the recursive matrix-update case, given either \( \hat{x}_r; \hat{X}_r \) and \( \hat{x}_a; \hat{X}_a \) as determined above, (215) and (216), respectively use the other measurement component,

\[
\hat{x}_{ra} = \hat{x}_r + \hat{X}_{ra} e_{\parallel}(\hat{a}_r) \left( \frac{\bar{a}_r - \hat{a}_r}{\sigma^2_A} \right) \quad \text{and} \quad \hat{x}_{ar} = \hat{x}_a + \hat{X}_{ar} e_{\perp}(\hat{a}_a) \left( \frac{\bar{a}_a - \hat{a}_a}{\sigma^2_R} \right),
\]

(221)

where

\[
\hat{X}_{ra} = \left[ \hat{X}_r^{-1} + \frac{1}{\sigma^2_A} e_{\parallel}(\hat{a}_r) e_{\parallel}^T(\hat{a}_r) \right]^{-1} \quad \text{and} \quad \hat{X}_{ar} = \left[ \hat{X}_a^{-1} + \frac{\hat{P}_a^2}{\sigma^2_R} e_{\parallel}(\hat{a}_a) e_{\parallel}^T(\hat{a}_a) \right]^{-1}.
\]

(222)

The updates in (215) and (216) shall be written symbolically as

\[
(\hat{x}; \hat{X}) \xrightarrow{\left( \pi, \sigma^2_A \right)} (\hat{x}_r; \hat{X}_r) \quad \text{and} \quad (\hat{x}; \hat{X}) \xrightarrow{\left( \pi, \sigma^2_A \right)} (\hat{x}_a; \hat{X}_a).
\]

(223)

And the updates in (221) and (222) shall be written symbolically as

\[
(\hat{x}; \hat{X}) \xrightarrow{\left( \bar{a}; \sigma^2_A \right)} (\hat{x}_{ra}; \hat{X}_{ra}) \quad \text{and} \quad (\hat{x}; \hat{X}) \xrightarrow{\left( \bar{a}; \sigma^2_A \right)} (\hat{x}_{ar}; \hat{X}_{ar}).
\]

(224)

Accordingly, these two EKF recursive update cases shall be represented by

\[
(\hat{x}; \hat{X}) \xrightarrow{\left( \bar{a}; \sigma^2_A \right)} (\hat{x}_r; \hat{X}_r) \xrightarrow{\left( \pi, \sigma^2_A \right)} (\hat{x}_{ra}; \hat{X}_{ra})
\]

(225)

and

\[
(\hat{x}; \hat{X}) \xrightarrow{\left( \bar{a}; \sigma^2_A \right)} (\hat{x}_a; \hat{X}_a) \xrightarrow{\left( \pi, \sigma^2_A \right)} (\hat{x}_{ar}; \hat{X}_{ar}).
\]

(226)

### 5.2.3 Illustration of the POT

Below, the EKF and the POT are illustrated, using the case in Chapter 3 where \( \sigma_A = \pi/12 \). Figure 23 provides the sample means and sample standard deviations of the estimates of \( x \) and \( y \) – the solid and dashed curves are respectively those of the EKF and the POT. And Figure 24 provides the corresponding converted back cases. Note that the POT does not appear to be very effective for this (very) short range case. (In Chapter
6 a longer range case is illustrated, and there the POT is seen to be very effective.) In these figures, track initialization transients are also apparent.

Figure 23 Effectiveness of the POT in rectangular coordinates

Figure 24 Effectiveness of the CB-POT (converted back into radar coordinates)
5.3 The Basic Extended-POT

Given the ineffectiveness of the POT at very short ranges (illustrated above), the POT shall now be extended as follows – a further extension will be given in the sequel. After updating the estimate using the azimuth measurement pair, \((\alpha^-; \sigma^2_{\alpha})\), instead of implicitly removing just a portion of that spurious linearization error in range (by using the range update after the azimuth update), explicitly remove all of it. In particular, after the azimuth-first update, simply restore the estimated range to its prior value; and then update that estimate with \((\overline{r}^-; \sigma^2_{\overline{r}})\).

Let \(\hat{x}^\alpha\) be the outcome of the azimuth-first update defined above. The range it determines is \(\hat{r}_a = \sqrt{\hat{x}^2_a + \hat{y}^2_a}\). But since \(\overline{r}^- = \sqrt{(\hat{x}^-)^2 + (\hat{y}^-)^2}\), the prior range corresponding to \(\hat{x}^-\) is known; and the spurious linearization error in range is also known – it is exactly \(\hat{r}_a - \overline{r}^-\). Therefore, adjust the estimate after the azimuth-first update, either as

\[
\hat{x}^{(\alpha)}_a = \begin{bmatrix}
\hat{r}^{(\alpha)}_a \cos \hat{\alpha}^{(\alpha)}_a \\
\hat{r}^{(\alpha)}_a \sin \hat{\alpha}^{(\alpha)}_a 
\end{bmatrix} = \begin{bmatrix}
\hat{r}_a \cos \hat{\alpha}_a \\
\hat{r}_a \sin \hat{\alpha}_a 
\end{bmatrix} - \begin{bmatrix}
(\hat{r}_a - \overline{r}^-) \cos \hat{\alpha}_a \\
(\hat{r}_a - \overline{r}^-) \sin \hat{\alpha}_a 
\end{bmatrix} = \begin{bmatrix}
\overline{r}^- \cos \hat{\alpha}_a \\
\overline{r}^- \sin \hat{\alpha}_a 
\end{bmatrix},
\]

or as

\[
\hat{x}^{(\alpha)}_a = \begin{bmatrix}
\hat{r}^{(\alpha)}_a \cos \hat{\alpha}^{(\alpha)}_a \\
\hat{r}^{(\alpha)}_a \sin \hat{\alpha}^{(\alpha)}_a 
\end{bmatrix} = \begin{bmatrix}
\hat{r}_a / \overline{r}_a \\
\hat{r}_a / \overline{r}_a 
\end{bmatrix} \left[\begin{bmatrix}
\hat{r}_a \cos \hat{\alpha}_a \\
\hat{r}_a \sin \hat{\alpha}_a 
\end{bmatrix} - \begin{bmatrix}
\overline{r}^- \cos \hat{\alpha}_a \\
\overline{r}^- \sin \hat{\alpha}_a 
\end{bmatrix}\right].
\]

And then use \((\overline{r}^-; \sigma^2_{\overline{r}})\) to update that adjusted estimate instead. That is, determine

\[
\hat{x}^{(\alpha)}_{ar} = \hat{x}^{(\alpha)}_a + \hat{\mathbf{x}}^{(\alpha)}_a \mathbf{e}_\perp(\hat{\alpha}_a) \left(\frac{\overline{r}^- - \hat{r}_a}{\sigma^2_{\overline{r}}}ight) \quad \text{and} \quad \hat{\mathbf{X}}^{(\alpha)}_{ar} = \left[\begin{bmatrix}
\hat{\mathbf{x}}^{-1}_a + \frac{(\overline{r}^-)^2}{\sigma^2_{\overline{r}}} \mathbf{e}_\parallel(\hat{\alpha}_a) \mathbf{e}_\parallel^T(\hat{\alpha}_a) 
\end{bmatrix}\right].
\]

Figure 25 illustrates the effectiveness of this version of the POT (the solid curves), now dubbed the (basic) Extended-POT (EPOT), along with the POT results of Figure 23 (the dashed curves). And Figure 26 provides the corresponding converted-back cases.
Figure 25 Effectiveness of the EPOT in rectangular coordinates

Figure 26 Effectiveness of the CB-EPOT (converted back into radar coordinates)
6 Application to Radar Tracking

The results of the analyses presented above in Chapters 5 and 6 will now be applied to a more stressing tracking problem, the one that has been used in the DCCM/UCCM literature over the years to show the superiority of those “debiasing” methods [20, 26].

The DCCM/UCCM “exemplar” is summarized as follows. The object is initially at 70 kilometers range and 45 degrees azimuth, and its motion has constant velocity, 15 meters per second due north (parallel to the Y-axis). The radar is fixed at the coordinate origin and provides 50 detections, which are at 60 second intervals. The measurements are all unbiased and mutually independent, and have gaussian distributions. The range standard deviation is $\sigma_r = 50$ meters. Azimuth has two cases: here $\sigma_A = 1.5$ degrees shall be used (2.5 will be used in the next Chapter). Finally, in the literature the system noise is 0.01 meters/seconds-squared in each coordinate, but here it is zero in all cases.

First, the basic CV tracking equations that were derived in Chapter 2 are extended to the two degree-of-freedom case, with everything in rectangular coordinates. And then the various forms of the estimators for the CV case are given and illustrated. It will be seen that the EKF with the POT is much better than the “debiased” and “consistent” PLKF. But it shall also be seen that the EPOT as defined in the previous Chapter is not as effective. Here that version is dubbed the basic-EPOT (B-EPOT) – an additional will be provided in the next Chapter.

6.1 The Basic 2DOF CV Tracking Equations

Let the state vector be $\mathbf{x}^T = (x, y, \dot{x}, \dot{y})$. Such shall also be written as $\mathbf{x}^T = (\xi^T, \dot{\xi}^T)$, with $\xi^T = (x, y)$ and $\dot{\xi}^T = (\dot{x}, \dot{y})$. 
Given a prior track at \( t_{n-1}, (\hat{x}_{n-1}, \hat{X}_{n-1}) \), the predicted track at \( t_n \) is
\[
\hat{x}_n = \Phi(\tau)\hat{x}_{n-1} \quad \text{and} \quad \hat{X}_n = \Phi(\tau)\hat{X}_{n-1}\Phi^T(\tau) + S(\tau),
\]
(230)
\( \tau = t_n - t_{n-1} \geq 0 \). And the transition and system noise matrices are
\[
\Phi(\tau) = I \otimes \begin{bmatrix} 1 & \tau I \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad S(\tau) = \delta I \otimes \begin{bmatrix} \tau^3/3 & \tau^2/2 \\ \tau^2/2 & \tau \end{bmatrix}
\]
(231)
(here, and in all the examples, \( \delta = 0 \)). Then, given a “rectangular detection” at \( t_n \),
\( (\xi_n; \Sigma_n) \), where \( \xi_n = Hx(t_n) + \hat{\xi}_n \) with \( H = [I \quad 0] \equiv I \otimes (1, 0) \), the update is
\[
\hat{x}_n = \hat{x}_n + K_n (\xi_n - H\hat{x}_n) \quad \text{and} \quad \hat{X}_n = (I - K_n H)\hat{X}_n,
\]
(232)
with
\[
K_n = \hat{X}_n H^T (H\hat{X}_n H^T + \Sigma_n)^{-1}.
\]
(233)
The fusion form of the update is
\[
\hat{x}_n = \hat{X}_n \left[ (\hat{X}_n)^{-1}\hat{x}_n + H^T \Sigma^{-1}\xi_n \right] \quad \text{and} \quad \hat{X}_n = \left[ (\hat{X}_n^{-1}) + H^T \Sigma_n H \right]^{-1}.
\]
(234)
And, using \( K_n = \hat{X}_n H^T \Sigma^{-1}_n \), the alternate form of the state vector update equation is
\[
\hat{x}_n = \hat{x}_n + \hat{X}_n H^T \Sigma^{-1}_n (\xi_n - H\hat{x}_n).
\]
(235)
Now to initialize a CV track at least two detections at distinct times are needed. Here such are said to occur at \( t_U \) and \( t_V \), with \( t_V > t_U \) (an initially unassociated one followed by an associated one, which verifies that the object exists). In particular, given \( (\xi_U; \Sigma_U) \) and \( (\xi_V; \Sigma_V) \), with \( \tau = t_V - t_U > 0 \), use the fusion form to determine
\[
\hat{x}_V = \hat{X}_V \left[ G_{-\tau}^T \Sigma_{-\tau}^{-1}\xi_U + G_{0}^T \Sigma_{0}^{-1}\xi_V \right] \quad \text{and} \quad \hat{X}_V^{-1} = G_{-\tau}^T \Sigma_{-\tau}^{-1} G_{-\tau} + G_{0}^T \Sigma_{0}^{-1} G_{0},
\]
(236)
with \( G_{\tau} \equiv H\Phi(\tau) \). Note that
\[
G_{\pm \tau} = \begin{bmatrix}
1 & 0 & \pm \tau & 0 \\
0 & 1 & 0 & \pm \tau
\end{bmatrix} \equiv I \otimes (1, \pm \tau).
\] (237)

Also, \( G_0 = H \). Now

\[
G_{\pm \tau}^T \Sigma^{-1} G_{\pm \tau} = \begin{bmatrix} I & \Sigma^{-1} \pm I \end{bmatrix} \equiv \Sigma^{-1} \otimes \begin{bmatrix} 1 & \pm \tau \end{bmatrix} \cdot (238)
\]

And so

\[
\hat{x}_v = \hat{x}_v \begin{bmatrix} \Sigma^{-1} \xi_v + \Sigma^{-1} \xi_v \pm \Sigma^{-1} \xi_v \\
- \tau \Sigma^{-1} \xi_v \pm \Sigma^{-1} \xi_v
\end{bmatrix}
\]

and

\[
\hat{x}_v^{-1} = \begin{bmatrix} 2 \Sigma^{-1} \pm \tau \Sigma^{-1} \\
- \tau \Sigma^{-1} \pm \tau \Sigma^{-1}
\end{bmatrix} \cdot (239)
\]

Thus,

\[
\hat{x}_v = \hat{x}_v \begin{bmatrix} \xi_v \\
\xi_v
\end{bmatrix} = \begin{bmatrix} \xi_v \\
(\xi_v - \xi_v) / \tau
\end{bmatrix}
\]

and

\[
\hat{x}_v = \Sigma \otimes \begin{bmatrix} 1 & 1 / \tau \\
1 / \tau & 2 / \tau^2
\end{bmatrix} \cdot (240)
\]

For convenience, write \((\hat{x}_0, \hat{X}_0) = (\hat{x}_v, \hat{X}_v)\) and \( t_0 = t_v \); and reindex the subsequent detections as \( n = 1, 2, \ldots, N \). Then, using the special case given in Chapter 2, for \( n = 1, 2, \ldots, N \), the updated “covariance” matrices are

\[
\hat{X}_n = \Sigma \otimes \begin{bmatrix} 2(2m-1)/m(m+1) & 6/(m+1) \tau \\
6/(m+1) \tau & 12/(m^2-1)mr^2
\end{bmatrix} \cdot (241)
\]

\( m = n + 2 \). In which case, the corresponding gain matrix is

\[
K_n = \hat{X}_n H^T \Sigma^{-1} = I \otimes \begin{bmatrix} 2(2m-1)/m(m+1) \\
6(m+1) \tau
\end{bmatrix} \cdot (242)
\]

6.1.1 The 2DOF CV LKF Case

Now in the CV LKF case the above equations are used formally with everything in radar coordinates. The state vector is \( \textbf{r}^T = (\rho^T, \phi^T) \), with \( \rho^T = (r, a) \) and \( \phi^T = (\dot{r}, \dot{a}) \). And
the measurement model is \( \bar{\rho}_n = Hr(t_n) + \tilde{\rho}_n \), with \( H = [I \ 0] \). For initialization, \( (\tilde{\rho}_v; \Sigma_R) \) are given, and the initial track is determined as

\[
\begin{bmatrix}
\hat{\rho}_0 \\
\hat{\rho}_0
\end{bmatrix}
= \begin{bmatrix}
\bar{\rho}_v \\
(\bar{\rho}_v - \tilde{\rho}_v) / \tau
\end{bmatrix}
\quad \text{and} \quad
\hat{\Sigma}_0 = \Sigma_R \otimes \begin{bmatrix}
1/\tau & 1/\tau \\
1/\tau & 2/\tau^2
\end{bmatrix}.
\] (243)

Then, given \( (\hat{\rho}_{n-1}; \hat{\Sigma}_{n-1}) \), \( n = 1, 2, \cdots, N \), the predicted LKF track at \( t_n \) is

\[
\begin{bmatrix}
\hat{\rho}_n \\
\hat{\rho}_n
\end{bmatrix}
= \Phi(\tau)\hat{\rho}_{n-1} \quad \text{and} \quad
\hat{\Sigma}_n = \Phi(\tau)\hat{\Sigma}_{n-1}\Phi^T(\tau) + S(\tau),
\] (244)

with \( \Phi \) and \( S \) as in (231). Note that here “CV” is a misnomer: this model defines a spiral in \( \mathbb{E} \). (In practice, this model is used for tracking incoming objects that are at very long ranges, and for “tracking” clutter.) Given \( (\bar{\rho}_n; \Sigma_R) \), the track is updated as

\[
\begin{bmatrix}
\hat{\rho}_n \\
\hat{\rho}_n
\end{bmatrix}
= \hat{\rho}_n + K_n (\bar{\rho}_n - H\hat{\rho}_n) \quad \text{and} \quad
\hat{\Sigma}_n = (I + K_n H_n)\hat{\Sigma}_n,
\] (245)

where

\[
K_n = \hat{\Sigma}_n H^T \left( H\hat{\Sigma}_n H^T + \Sigma_R \right)^{-1},
\] (246)

with the components of this \( K_n \) the same as those in (242).

Figure 27 illustrates this LKF case with 100 Monte Carlo trials. The left-hand side shows the radar measurements plotted in \( \mathbb{E} \), where the problem is defined. And the right-hand side shows the LKF estimates of position, again in \( \mathbb{E} \). The darker curves down the centers are respectively the sample means of the measurements and estimates. Figure 28 provides the position errors of these LKF estimates; and there the dark curves represent the sample means of the respective sets of errors – the top two plots provide the errors in radar coordinates, and the bottom two plots provide the errors in rectangular coordinates. Note that this LKF is biased – because its model of motion defines a spiral. An improvement will be given in the next Section.
Figure 27 Radar measurements and LKF position estimates (in Euclidean space)

Figure 28 LKF position estimation errors (2DOF CV case)
6.1.2 The 2DOF CV C-LKF Case

Here the Radar Principal Cartesian Coordinates (RPCC) method is used to illustrate the C-LKF case [58]. The updates are determined in radar coordinates, and the covariance matrix is propagated in radar coordinates, but the estimated state vector is propagated in rectangular coordinates. For the sake of comparison with the previous results, however, the same LKF initial track used above will be reused here, \((\hat{\mathbf{r}}_0; \hat{\mathbf{R}}_0) \equiv (\hat{\mathbf{r}}_n; \hat{\mathbf{R}}_n)\).

First, to propagate the estimated state vectors in rectangular coordinates, given
\[
\begin{align*}
\dot{\mathbf{r}}_{n-1}^T &= (\rho_{n-1}^r, \hat{\rho}_{n-1}^r), \\
\hat{\mathbf{r}}_{n-1}^T &= (\hat{\rho}_{n-1}^r, \hat{\rho}_{n-1}^a)
\end{align*}
\]
respectively as \(\dot{\xi}_{n-1} = \mathbf{h}^{-1}(\hat{\rho}_{n-1})\) and \(\dot{\xi}_{n-1} = \mathbf{J}^{-1}(\hat{\rho}_{n-1})\hat{\rho}_{n-1}\). That is,
\[
\begin{align*}
\dot{\xi} &= \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} r \cos a \\ r \sin a \end{bmatrix} \\
\dot{\xi} &= \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos a & -r \sin a \\ +r \sin a & r \cos a \end{bmatrix} \begin{bmatrix} \dot{r} \\ \dot{a} \end{bmatrix}.
\end{align*}
\]
(247)
And then form \(\hat{\mathbf{x}}_{n-1}^T = (\hat{\xi}_{n-1}^T, \hat{\xi}_{n-1}^T)\). In which case the initial conditions have the form
\[
(\hat{\mathbf{x}}_{n-1}; \hat{\mathbf{R}}_{n-1}).
\]
And the propagation equations are respectively
\[
\dot{\mathbf{x}}_{n} = \Phi(\tau)\hat{\mathbf{x}}_{n-1} \quad \text{and} \quad \dot{\hat{\mathbf{R}}}_{n} = \Phi(\tau)\hat{\mathbf{R}}_{n-1}\Phi(\tau) + \mathbf{S}(\tau).
\]
(248)
For the updates, the predicted estimate in rectangular coordinates, \(\hat{\mathbf{x}}_{n}^-\), is transformed back into radar coordinates as \(\hat{\mathbf{p}}_{n}^- = \mathbf{h}(\hat{\xi}_{n}^-)\) and \(\hat{\mathbf{p}}_{n}^+ = \mathbf{J}(\hat{\xi}_{n})\hat{\xi}_{n}\). That is,
\[
\begin{align*}
\hat{\mathbf{p}}_{n}^- &= \begin{bmatrix} \sqrt{\hat{\xi}_{n}^x + \hat{\xi}_{n}^y} \\ \arctan(\hat{\xi}_{n}^x, \hat{\xi}_{n}^y) \end{bmatrix} \\
\hat{\mathbf{p}}_{n}^+ &= \begin{bmatrix} \cos \hat{\alpha}_{n} & \sin \hat{\alpha}_{n} \\ -\left(1/\hat{r}_{n}\right) \sin \hat{\alpha}_{n} & \left(1/\hat{r}_{n}\right) \cos \hat{\alpha}_{n} \end{bmatrix} \begin{bmatrix} \hat{\xi}_{n}^x \\ \hat{\xi}_{n}^y \end{bmatrix}.
\end{align*}
\]
(249)
And \((\hat{\mathbf{r}}_{n}; \hat{\mathbf{R}}_{n})\), where \((\hat{\mathbf{r}}_{n})^T = (\hat{\mathbf{p}}_{n}^-)^T, (\hat{\mathbf{p}}_{n}^+)^T\), is updated using \((\hat{\mathbf{p}}_{n}; \Sigma_{n})\) with (245) and (246). Figure 29 illustrates these C-LKF estimates, along with the corresponding LKF ones shown above. Figure 30 provide the position errors.
Figure 29 Comparison of the LKF and C-LKF (in Euclidean space)

Figure 30 C-LKF position estimation errors (2DOF CV case)
6.1.3 The 2DOF CV Scalar-Weight PLKF Case

Here the “debiased” S-PLKF is illustrated, using the UCCM measurements,
\[ \overline{z}_n = (1/\lambda)h^{-1}(\overline{p}_n), \quad \text{with} \quad \lambda = e^{-\sigma^2/2}. \]
But now the initial track, \((\hat{x}_0; \hat{X}_0)\) at \(t_0\), is determined directly in rectangular coordinates, as
\[
\hat{x}_0 = \begin{bmatrix} \hat{\xi}_v \\ \hat{\xi}_n \end{bmatrix} = \begin{bmatrix} \overline{\xi}_v \\ (\overline{\xi}_n - \overline{p}_n) / \tau \end{bmatrix} \quad \text{and} \quad \hat{X}_0 = \overline{X}_0 \otimes \begin{bmatrix} 1/	au & 1/\tau \\ 1/\tau & 2/\tau^2 \end{bmatrix}.
\]

Recall that the S-PLKF uses \(\overline{X}_n = I/\overline{w}\) for the measurements’ covariance matrices.

And so for \(n = 1, 2, \cdots, N\), with \(m = n + 2\), the gain matrices are the same as in the previous two examples,
\[
K_n = I \otimes \begin{bmatrix} (2m-1)/m(m+1) \\ 6/m(m+1) \tau \end{bmatrix}.
\]

Figure 31 illustrates this “debiased” S-PLKF case along with the C-LKF results shown above. The estimation position errors are provided by Figure 32.

![Figure 31 Comparison of the C-LKF and S-PLKF (in Euclidean space)](image-url)
6.1.4 The 2DOF CV Matrix-Weight PLKF Case

Here the prior-weight UCCM M-PLKF is illustrated. That is, \( \bar{X}_n = \Sigma_x \left( \bar{\xi}_{n-1}(\lambda) \right) \), where

\[
\Sigma_x(x) = \left( \sigma_R^2 + r^2 \right) \left[ \begin{array}{cc} 1 - e^{-2\sigma^2} & 1 - e^{-2\sigma^2} \\ 0 & 1 - e^{-2\sigma^2} \end{array} \right] \left[ \begin{array}{c} x^2 \\ y^2 \end{array} \right] - e^{-\sigma^2} \left[ \begin{array}{c} x^2 \\ xy \end{array} \right].
\]  

(252)

As before, the propagation equations are

\[
\hat{x}_n = \Phi(\tau)\hat{x}_{n-1} \quad \text{and} \quad \hat{X}_n = \Phi(\tau)\hat{X}_{n-1}\Phi^T(\tau) + S(\tau),
\]  

(253)

and the update equations have the form

\[
\hat{x}_n = \hat{x}_n^- + K_n \left( \bar{\xi}_n - H\hat{x}_n^- \right) \quad \text{and} \quad \hat{X}_n = (I - K_n H_n)\hat{X}_n^-.
\]  

(254)

\[
K_n = \hat{X}_n^- H^T (H\hat{X}_n^- H^T + \bar{X}_n^-)^{-1}.
\]  

(255)

Figure 33 illustrates the estimates for this “debiased and consistent” M-PLKF, along with those of the “debiased” S-LKF case shown in the previous section. Figure 34 provides the corresponding estimation errors.
Figure 33 Comparison of the S-PLKF and M-PLKF (in Euclidean space)

Figure 34 M-PLKF position estimation errors (2DOF CV case)
6.1.5 Discussion of the 2DOV CV C-LKF, S-PLKF, and M-PLKF Cases

Figure 35 compares the sample mean-errors of the C-LKF, S-PLKF, and M-PLKF position estimates that were shown above (but here 500 Monte Carlo trials are used).

Figure 36 provides the corresponding sample standard deviations.

Figure 35 Sample means of C-LKF, S-PLKF, and M-PLKF errors (CV 2DOF)
In each figure the top sets of plots are in rectangular coordinates while the lower sets are in radar coordinates (the latter are the “converted-back” cases – those sample means and sample standard deviations being determined after transforming the estimates).
Note that the mean-errors in range of the “debiased and consistent” M-LKF appear to be worse than the “debiased” S-PLKF ones. However, for \( n > 20 \), the sample standard deviations of those M-PLKF estimates are better than the ones of the C-LKF and S-PLKF estimates. That is because the C-LKF and S-PLKF measurement-weights, \( \tilde{w} \), are independent of range and azimuth, while the M-PLKF ones scale as \( 1/(\sigma_r^2 + r_n\sigma_a^2) \).

### 6.2 The 2DOF CV EKF and POT Cases

Here the performance of the EKF is illustrated, first with and then without the POT. For the sake of comparison with the previous results, all the initial tracks are the same as the one used above in the (prior-weight UCCM) M-PLKF case given above. And the propagation equations are also the same as in that M-PLKF case.

#### 6.2.1 The 2DOF CV EKF without the POT

Recall that the (vector) EKF update equations are

\[
\hat{x} = \hat{x}^- + K \left( \tilde{p} - h(\hat{\xi}^-) \right) \quad \text{and} \quad \hat{X} = (I - KH^-)\hat{X}^- \quad (256)
\]

and

\[
K = \hat{X}^- H^T \left( HH^T + \Sigma_r \right)^{-1}. \quad (257)
\]

Figure 37 illustrates this EKF case with 100 Monte Carlo trials, along with the “debiased and consistent” M-PLKF ones shown earlier. Figure 38 provide the errors of these EKF estimates.
Figure 37 Comparison of the M-PLKF and EKF (in Euclidean space)

Figure 38 EKF position estimation errors (2DOF CV case)
6.2.2 The 2DOF CV EKF with the POT

Recall that the POT uses the measurement components of a radar detection recursively, \((\bar{a};\sigma_a^2)\) first and \((\bar{r};\sigma_r^2)\) last. In particular, given \((\hat{x}^-;\hat{X}^-)\) and \((\bar{a};\sigma_a^2)\), the azimuth-first scalar-update is

\[
\hat{x}_a = \hat{x}^- + K_a (\bar{a} - \hat{a}^-) \quad \text{and} \quad \hat{X}_a = \left( I - K_a H_a (\hat{x}^-) \right) \hat{X}^-
\]

\[
K_a = \hat{X}^- H_a^T (\hat{x}^-) \left( H_a (\hat{x}^-) \hat{X}^- H_a^T (\hat{x}^-) + \sigma_a^2 \right)^{-1},
\]

with \(H_a \equiv da/dx^T\). And, given \((\hat{x}_a;\hat{X}_a)\) and \((\bar{r};\sigma_r^2)\), the range-last scalar-update is

\[
\hat{x}_{ar} = \hat{x}_a + K_{ar} (\bar{r} - \hat{r}_a) \quad \text{and} \quad \hat{X}_{ar} = \left( I - K_{ar} H_r (\hat{x}_a) \right) \hat{X}_a
\]

and

\[
K_{ar} = \hat{X}_a H_r^T (\hat{x}_a) \left( H_r (\hat{x}_a) \hat{X}_a H_r^T (\hat{x}_a) + \sigma_r^2 \right)^{-1}.
\]

with \(H_r \equiv dr/dx^T\).

Figure 39 illustrates this EKF with the POT case, using the same 100 Monte Carlo trial data, along with the basic EKF estimates that were shown above. And Figure 40 provides the corresponding position errors. Note that the estimates determined by the EKF with the POT appear to be better than those determined in the previous section by the EKF. Also in the EKF without that POT case, there were a few “estimation paths” that did not seem to converge very well. Here that problem does not seem to appear.
Figure 39 Comparison of EKF and POT (in Euclidean space)

Figure 40 POT position estimation errors (2DOF CV case)
6.3 The 2DOF CV Case with the Basic-EPOT

Recall that in the previous Chapter the EPOT “extended” the POT as follows: after the azimuth update, the range of the track is restored to its prior value, and then the range update is determined. And so, given the scalar azimuth-first update, as determined under the POT, \((\hat{x}_a^*, \hat{X}_a^*)\), let \(\hat{\xi}_a^{(*)} = (r^- / r_a) \hat{\xi}_a\) and form \(\hat{x}_a^{(*)T} = (\hat{\xi}_a^{(*)T}, \hat{\xi}_a^{(*)})\). And then use the range measurement as

\[
\hat{x}_{ar}^{(*)} = \hat{x}_a^{(*)} + \mathbf{K}_{ar}^{(*)}(\hat{r}^* - \hat{r}^-) \quad \text{and} \quad \hat{X}_{ar}^{(*)} = \left( \mathbf{I} - \mathbf{K}_{ar}^{(*)} \mathbf{H}_{r}^{(*)}(\hat{\xi}_a^{(*)}) \right) \hat{\mathbf{X}}_a^{(*)}
\]  \hspace{1cm} (262)

\[
\mathbf{K}_{ar}^{(*)} = \hat{\mathbf{X}}_{ar}^{(*)} \mathbf{H}_{r}^{(*)T}(\hat{\xi}_a^{(*)}) \left( \mathbf{H}_{r}^{(*)}(\hat{\xi}_a^{(*)}) \hat{\mathbf{X}}_{ar}^{(*)} \mathbf{H}_{r}^{(*)T}(\hat{\xi}_a^{(*)}) + \mathbf{Z}_r^2 \right)^{-1}.
\]  \hspace{1cm} (263)

Note that \(\hat{r}^-\) appears in the first expression of (262) because \(\hat{\xi}_a^{(*)} = (r^- / r_a) \hat{\xi}_a\) implies that \(\hat{r}_a^{(*)} = \hat{r}^-\). Figure 39 illustrates the estimates of this case, along with those of the POT shown above. Figure 40 provides the position errors.

Figure 41 Comparison of POT and Basic EPOT (in Euclidean space)
6.4 Comparison of the 2DOV CV EKF, POT Cases, and basic EPOT

Figure 43 and Figure 44 provide the sample mean-errors and sample standard deviations of the M-PLKF, EKF, and POT position estimates that were shown above (but now 500 Monte Carlo trials are being used). And the sample mean-errors and sample standard deviations of the EKF, POT, and EPOT cases are shown in Figure 45 and Figure 46.

Note that in Figure 43 and Figure 44 the EKF estimates are generally the worst, while the EKF with the POT is generally the best. (In the DCCM/ UCCM literature, the EKF usually provides the worst performance – there the EKF with the POT is ignored.) However, in Figure 45 and Figure 46 the EPOT is the worst. This performance issue of the EPOT shall be analyzed in the next Chapter; and there a remedy will be provided – accordingly, the version used here is dubbed the Basic-EPOT (B-EPOT).
Figure 43 M-PLKF, EKF, and POT errors sample means (CV 2DOF)
Figure 44 M-PLKF, EKF, and POT sample standard deviations (CV 2DOF)
Figure 45 M-PLKF, POT, and Basic EPOT errors sample means (CV 2DOF)
Figure 46 M-PLKF, POT, and Basic EPOT sample standard deviations (CV 2DOF)
7 Analysis of the EPOT

In Chapter 5 the EPOT was seen to perform better than the POT (the CP case), but in
Chapter 6, where the object was moving (the CV case), the POT appeared to be better.
Here it is shown that in the latter case the EPOT updates of position and velocity were
inconsistent with one another; and a remedy for that inconsistency shall be provided.
And it will also be shown that this EPOT overcomes the fundamental limitation POT:
under it the measurement components of a detection may be used recursively in any order
to update a track, with virtually the same results. As in Chapter 5, the scalar-weight case
will be analyzed first, followed by the matrix-weight case. And, as in Chapter 6, the state
vector is \( x^T = (\xi^T, \dot{\xi}^T) \), with \( \xi^T = (x, y) \) and \( \dot{\xi}^T = (\dot{x}, \dot{y}) \); the coordinate transformations
are \( p = h(\xi) \) and \( \dot{\xi} = h^{-1}(p) \), with \( p^T = (r, a) \); and \( H(x) = J(\xi) \otimes (1, 0) \).

For convenience, the EKF update equations are summarized as follows. Given
\( (\hat{x}^-, \hat{X}^-) \) and \( (\hat{p}; \Sigma_R) \),
\[
\hat{x} = \hat{x}^- + K (\hat{p} - \hat{\rho}^-) \quad \text{and} \quad \dot{\hat{X}} = \left( I - KH(\hat{x}^-) \right) \hat{X}^- \tag{264}
\]
with
\[
K = \hat{X}^- H^T(\hat{x}^-) \left( H(\hat{x}^-) \hat{X}^- H^T(\hat{x}^-) + \Sigma_R \right)^{-1}. \tag{265}
\]
And the alternate form for updating the estimate is
\[
\hat{x} = \hat{x}^- + \hat{X} H^T(\hat{x}^-) \Sigma_R^{-1} \left( \hat{p} - \hat{\rho}^- \right). \tag{266}
\]
7.1 The Scalar-Weight Consistent EPOT

First, some more notation is defined. Recall that the scalar-weight EKF update is

\[
\hat{x} = \hat{x}^- + \frac{w}{\hat{w}} J^{-1}(\hat{r}^-) (\hat{\rho}^- - \hat{\rho}^-) \quad \text{and} \quad \hat{w} = \hat{w}^- + \hat{w}.
\]

(267)

Here, to emphasize the geometrical aspects of the update, that position update is written

\[
\hat{\xi} = \hat{\xi}^- + r_{\Delta} \mathbf{e}_\parallel (\hat{a}^-) + \hat{r}^- a_{\Delta} \mathbf{e}_\perp (\hat{a}^-)
\]

(268)

with \(\mathbf{e}_\parallel \equiv \mathbf{e}_r\) and \(\mathbf{e}_\perp \equiv \mathbf{e}_a\),

\[
\mathbf{e}_\parallel^T (a) \equiv (\cos a, \sin a) \quad \text{and} \quad \mathbf{e}_\perp^T (a) \equiv (-\sin a, \cos a),
\]

(269)

and with

\[
r_{\Delta} = \frac{\bar{w}}{\hat{w}} (\bar{r} - \hat{r}^-) \quad \text{and} \quad a_{\Delta} = \frac{\bar{w}}{\hat{w}} (\bar{a} - \hat{a}^-).
\]

(270)

Now in the CV case the scalar-weight EKF update is defined as

\[
\begin{bmatrix}
\hat{x}^{(s)}_{\text{CV}} \\
\hat{r}^{(s)}_{\text{CV}}
\end{bmatrix}
= \begin{bmatrix}
\hat{x}^- \\
\hat{r}^-
\end{bmatrix}
+ \mathbf{K}^{(s)}_{\text{CV}} \begin{bmatrix}
\bar{r} - \hat{r}^- \\
\bar{a} - \hat{a}^-
\end{bmatrix},
\]

(271)

where

\[
\mathbf{K}^{(s)}_{\text{CV}} = J^{-1}(\hat{\rho}^-) \otimes \begin{bmatrix}
\alpha \\
\beta/\tau
\end{bmatrix},
\]

(272)

with \(J^{-1}(\rho) = \begin{bmatrix} \mathbf{e}_\parallel (a) & \mathbf{e}_\perp (a) \end{bmatrix}\). The details for determining \(\alpha\) and \(\beta\) were given in Chapter 6. Specifically, using \(m = n + 2\), the gains were shown to be

\[
\alpha_n = 2(2m-1)/m(m+1) \quad \text{and} \quad \beta_n = 6/m(m+1)\tau.
\]

(273)

In (272) these gains are tacitly being used, but without the indices to simplify the notation. (For the remainder of this section the superscript “(s)” and subscript “CV” shall be dropped.)
Now write the scalar-weight EKF update determined by (271) and (272) as
\[
\hat{\xi} = \hat{\xi}^+ + \alpha \begin{bmatrix} \cos \hat{a}^- & -\hat{r}^- \sin \hat{a}^- \\ \sin \hat{a}^- & +\hat{r}^- \cos \hat{a}^- \end{bmatrix} \begin{bmatrix} \bar{r} - \hat{r}^- \\ \bar{a} - \hat{a}^- \end{bmatrix}
\]
(274)
and
\[
\hat{\xi} = \hat{\xi}^+ + (\beta / \tau) \begin{bmatrix} \cos \hat{a}^- & -\hat{r}^- \sin \hat{a}^- \\ \sin \hat{a}^- & +\hat{r}^- \cos \hat{a}^- \end{bmatrix} \begin{bmatrix} \bar{r} - \hat{r}^- \\ \bar{a} - \hat{a}^- \end{bmatrix}.
\]
(275)
And then let
\[
\Delta r^- \equiv \alpha(\bar{r} - \hat{r}^-) \quad \text{and} \quad \Delta a^- \equiv \alpha(\bar{a} - \hat{a}^-)
\]
(276)
and
\[
\Delta \hat{r}^- \equiv (\beta / \tau)(\bar{r} - \hat{r}^-) \quad \text{and} \quad \Delta \hat{a}^- \equiv (\beta / \tau)(\bar{a} - \hat{a}^-).
\]
(277)
Finally, write (274) and (275) together as
\[
\begin{bmatrix} \hat{\xi}^- \\ \hat{\xi}^+ \end{bmatrix} = \begin{bmatrix} \hat{\xi}^- \\ \hat{\xi}^+ \end{bmatrix} + \begin{bmatrix} \Delta r^- & \hat{r}^- \Delta a^- \\ \Delta \hat{r}^- & \hat{r}^- \Delta \hat{a}^- \end{bmatrix} \begin{bmatrix} e_{\parallel}(\hat{a}^-) \\ e_{\perp}(\hat{a}^-) \end{bmatrix}.
\]
(278)
That is, using \( \mathbf{x}^T = (\xi^T, \dot{\xi}^T) \),
\[
\mathbf{\dot{x}} = \mathbf{\dot{x}}^- + \Delta \hat{r}^- e_{\parallel}(\hat{a}^-) + \hat{r}^- \Delta \hat{a}^- e_{\perp}(\hat{a}^-).
\]
(279)
Next, recall the affine space \( \mathbf{A} \). Given \( \xi \in \mathbf{X} \) and \( \rho \in \mathbf{R} \), with \( \rho = \mathbf{h}(\xi) \), the column vectors \( \xi \) and \( \rho \) represent the same point in \( \mathbf{E} \); and their corresponding (abstract) vectors in \( \mathbf{A} \) are also the same, \( \xi = \rho \). That is, using (4) with \( r = \sqrt{x^2 + y^2} \) and \( a = \arctan(y, x) \),
\[
\xi = x e_{\parallel} + x e_{\perp} = r e_{\parallel}(a) + 0 e_{\perp}(a) = \rho.
\]
(280)
Thus, in \( \mathbf{A} \) the position and velocity updates given by (274) and (275) are simply
\[
\hat{\xi} = \hat{\xi}^- + \Delta r^- e_{\parallel} + \hat{r}^- \Delta a^- e_{\perp} \quad \text{and} \quad \hat{\xi} = \hat{\xi}^- + \Delta \hat{r}^- e_{\parallel} + \hat{r}^- \Delta \hat{a}^- e_{\perp},
\]
(281)
where, for convenience, $e^{-}_e \equiv e_e(\hat{a}^-)$ and $e^{-}_\perp \equiv e_\perp(\hat{a}^-)$. Finally, in $A$ the update equation corresponding to (271) and (278) is

$$
\begin{bmatrix}
\tilde{e}^-_e \\
\tilde{e}^-_c \\
\tilde{e}^-_r
\end{bmatrix} = 
\begin{bmatrix}
\tilde{\xi}^- \\
\tilde{\xi}^-_c \\
\tilde{\xi}^-_r
\end{bmatrix} +
\begin{bmatrix}
\Delta r^- & \hat{\Delta} a^- \\
\hat{\Delta} r^- & \hat{\Delta} a^- \\
\end{bmatrix}
\begin{bmatrix}
e^-_e \\
e^-_c \\
e^-_r
\end{bmatrix}.
$$

(282)

Figure 47 depicts the geometrical aspect of this EKF update of position in $A$, along with the corresponding LKF update. (The velocity update will be discussed shortly.) There the azimuth-first-range-last sequential EKF update is being used,

$$
\hat{\xi}^-_u = \tilde{\xi}^- + \hat{\Delta} a^- e^-_\perp \quad \text{and} \quad \hat{\xi}^-_{EKF} = \tilde{\xi}^- + \Delta r^- e^-_\parallel.
$$

(283)

The sequential range-first-azimuth-last update provides the same result, $\hat{\xi}^-_r = \hat{\xi}^- + \hat{\Delta} r^- r^- e^-_\parallel$ and $\hat{\xi}^-_{EKF} = \hat{\xi}^- + \hat{\Delta} a^- e^-_\perp$. Note that the LKF position update is simply the composition of rotating $\hat{\xi}^-$ by the angle $\Delta a^-$, and the translation along the resulting radial by $\Delta r^-$ (and these two operations of rotation and translation commute).
Figure 48 illustrates the corresponding updates determined by the POT and EPOT.

For the POT, the azimuth-first update is the same as in (283); but its recursive range-last update employs \( \hat{r}_a = \sqrt{\hat{x}_a^2 + \hat{y}_a^2} \) and \( \hat{a}_a = \arctan(\hat{y}_a, \hat{x}_a) \). That is,

\[
\hat{\xi}_{\text{POT}} = \hat{\xi}_a + \Delta r_a e_r^{(a)},
\]

where

\[
\Delta r_a \equiv \alpha(\hat{r} - \hat{r}_a) \quad \text{and} \quad e_r^{(a)} \equiv e_{\parallel}(\hat{a}_a).
\]

Similarly, given \( \hat{\xi}_a \), the range-last position update determined by the EPOT is

\[
\hat{\xi}_{\text{EPOT}} = \hat{\xi}_a + \Delta r_a e_r^{(a)},
\]

where \( \hat{\xi}_a^{(c)} \equiv (\hat{r} - \hat{r}_a)\hat{\xi}_a \), with \( \Delta r_a \) determined by (276).
Note that (274) and (275) are related by $\hat{\rho} - \bar{\rho}^{-}$, namely,

$$\hat{\xi} - \hat{\xi}^{-} = \hat{\xi}^{-} + \alpha J^{-1}(\hat{\rho}^{-})(\bar{\rho}^{-} - \hat{\rho}^{-}) \quad \text{and} \quad \hat{\xi} = \hat{\xi}^{-} + (\beta/\tau)J^{-1}(\hat{\rho}^{-})(\bar{\rho}^{-}) \quad \text{.} \tag{287}$$

In particular, given the residual $\bar{\rho}^{-} - \hat{\rho}^{-}$, the update determines a position differential, $\hat{\xi} - \hat{\xi}^{-}$, and a velocity differential, $\hat{\xi} - \hat{\xi}^{-}$. Figure 49 shows the respective position differentials for the LKF, EKF, POT, and EPOT updates that were illustrated above in Figure 47 and Figure 48.

Now solve for $\hat{\xi} - \hat{\xi}^{-}$ in the first expression in (287),

$$\bar{\rho}^{-} - \hat{\rho}^{-} = (1/\alpha)J(\hat{\xi}^{-})\left(\hat{\xi} - \hat{\xi}^{-}\right) \quad \text{.} \tag{288}$$

And then substitute this result into the second expression in (287). Also, recall that $I = J(\hat{\rho}^{-})J^{-1}(\hat{\rho}^{-})$ is an identity when $\hat{\rho}^{-} \neq 0$. Thus, the velocity differential determined by the second expression in (287) is

$$\hat{\xi} - \hat{\xi}^{-} = (\beta/\tau\alpha)\left(\hat{\xi} - \hat{\xi}^{-}\right) \quad \text{.} \tag{289}$$
This last expression defines the *position-velocity consistency condition* for the update. Such is simply a linear relation between differentials, \( \hat{\xi} - \hat{\xi}^- \) and \( \hat{\xi} - \hat{\xi}^- \). Indeed, in the scalar-weight case being used in this section, \( \hat{\xi} - \hat{\xi}^- \) and \( \hat{\xi} - \hat{\xi}^- \) are related by the scalar \((\beta/\tau \alpha)\), which is independent of both \( \overline{p} \) and \( \hat{\xi}^- \). The LKF, EKF, and POT cases inherently satisfy the position-velocity consistency condition defined above, but the “Basic-EPOT” does not (the Basic-EPOT modifies \( \hat{\xi} - \hat{\xi}^- \), without changing \( \hat{\xi} - \hat{\xi}^- \)).

Fortunately, the EPOT can easily be made to satisfy (289) by first updating the position estimate (using the Basic-EPOT), and then updating the velocity by using

\[
\hat{\xi}_{\text{EPOT}} \equiv \hat{\xi}^- + (\beta/\alpha \tau) \left( \hat{\xi}_{\text{EPOT}} - \hat{\xi}^- \right). \tag{290}
\]

Figure 50 and Figure 51 illustrate the performance of the EPOT with this *velocity-consistent* scalar-weight update, along with the “debiased” (UCCM) scalar-weight PLKF and the scalar-weight POT. Figure 50 provides the sample means of the estimation errors and Figure 51 provides the sample standard deviations of those errors. Here the same scenario used in Chapter 6 is being reused, but with \( \sigma_A = 2.5 \) instead of \( \sigma_A = 1.5 \). Note that the respective sample means are more or less the same, except for range – those of the “debiased” S-PLKF are worse than those of the S-POT and S-EPOT. Note also that the sample standard deviations of the S-PLKF, S-POT and S-EPOT position estimates are now indistinguishable, except for range: the S-PLKF range estimates are the worst while the S-EPOT range estimates are now the best.
Figure 50 PLKF, POT, and EPOT sample means (scalar-weight case)
Figure 51 PLKF, POT, and EPOT sample standard deviations (scalar-weight case)
7.2 The Matrix-Weight Consistent EPOT Case

In this Section the position-velocity consistency condition for the matrix-weight EPOT update is provided, and then illustrated. First, given the CV form of the updated estimate \( \hat{\mathbf{x}}^T = (\hat{\xi}^T, \hat{\xi}^T) \), write its associated covariance matrix as

\[
\mathbf{X} = \begin{bmatrix}
\mathbf{X}_{\xi\xi} & \mathbf{X}_{\xi\hat{\xi}} \\
\mathbf{X}_{\hat{\xi}\xi} & \mathbf{X}_{\hat{\xi}\hat{\xi}}
\end{bmatrix}.
\]

Here each subscripted "\( \mathbf{X} \)" is a \( 2 \times 2 \) sub-matrix; and \( \mathbf{X}_{\xi\xi} \) is positive definite since \( \hat{\mathbf{X}} \) is positive definite by assumption.

To analyze the matrix-weight case, an approach similar to that taken in the previous section will be followed. First, write the position and velocity updates respectively as

\[
\hat{\xi} - \hat{\xi}^- = \mathbf{X}_{\xi\xi} \mathbf{J}^T \mathbf{\Sigma}_r^{-1} (\hat{\mathbf{p}} - \hat{\mathbf{p}}^-) \quad \text{and} \quad \hat{\xi} - \hat{\xi}^- = \mathbf{X}_{\xi\hat{\xi}} \mathbf{J}^T \mathbf{\Sigma}_r^{-1} (\hat{\mathbf{p}} - \hat{\mathbf{p}}^-). \tag{291}
\]

As before, the first expression in (291) provides

\[
\hat{\mathbf{p}} - \hat{\mathbf{p}}^- = \mathbf{\Sigma}_r \mathbf{J}^T \mathbf{X}_{\xi\xi}^{-1} (\hat{\xi} - \hat{\xi}^-). \tag{292}
\]

And substituting this result into the second expression in (291) leads to

\[
\hat{\xi} - \hat{\xi}^- = \mathbf{X}_{\xi\hat{\xi}} \mathbf{X}_{\xi\xi}^{-1} (\hat{\xi} - \hat{\xi}^-). \tag{293}
\]

Such is the position-velocity consistency constraint for the matrix-weight case.

Figure 52 and Figure 53 provide the estimation errors of the consistent EPOT, defined in this section. Also shown are the corresponding errors M-PLKF and POT that were shown in Figures 26 and 27 of Chapter 6. Here, for the sake of comparison with those previous results, \( \sigma_A = 1.5 \). Now the mean-errors of the EPOT and POT are basically the same for range; and their sample standard deviations are mostly indistinguishable. The ones of the “debiased and consistent” M-PLKF are the worst.
Figure 52 M-PLKF, POT, and “CV EPOT” sample mean errors (CV 2DOF)
Figure 53 M-PLKF, POT, and “CV EPOT” sample standard deviations (CV 2DOF)


7.3 Extension to Higher-Order Models of Motion

The (consistent) CV matrix-weight EPOT is extended to the CA case as follows. First, write the CA state vector and measurement model matrix as

\[ \mathbf{x}^T = (\xi^T, \xi^T, \xi^T) \quad \text{and} \quad \mathbf{H}(\xi) = \mathbf{J}(\mathbf{x}) \otimes (1, 0, 0). \] (294)

And write the associated covariance matrix of the updated estimate as

\[
\mathbf{X} = \begin{bmatrix}
X_{\xi\xi} & X_{\xi\dot{\xi}} & X_{\xi\ddot{\xi}} \\
X_{\xi\dot{\xi}} & X_{\dot{\xi}\dot{\xi}} & X_{\dot{\xi}\ddot{\xi}} \\
X_{\xi\ddot{\xi}} & X_{\dot{\xi}\ddot{\xi}} & X_{\ddot{\xi}\ddot{\xi}}
\end{bmatrix}.
\]

The CA expression corresponding to the second expression in (291) is then

\[
\begin{bmatrix}
\mathbf{s}_{\xi\xi}^2 - \mathbf{s}_{\xi\dot{\xi}}^2 \\
\mathbf{s}_{\xi\dot{\xi}}^2 - \mathbf{s}_{\dot{\xi}\dot{\xi}}^2 \\
\mathbf{s}_{\xi\ddot{\xi}}^2 - \mathbf{s}_{\dot{\xi}\ddot{\xi}}^2
\end{bmatrix} = \begin{bmatrix}
X_{\xi\xi} & X_{\xi\dot{\xi}} \\
X_{\xi\dot{\xi}} & X_{\dot{\xi}\dot{\xi}}
\end{bmatrix} \mathbf{J}^T \Sigma_r^{-1} (\bar{\mathbf{p}} - \bar{\mathbf{p}}^\circ). (295)
\]

And the expression for the CA case corresponding to (293) is

\[
\begin{bmatrix}
\mathbf{s}_{\xi\xi}^2 - \mathbf{s}_{\xi\dot{\xi}}^2 \\
\mathbf{s}_{\xi\dot{\xi}}^2 - \mathbf{s}_{\dot{\xi}\dot{\xi}}^2 \\
\mathbf{s}_{\xi\ddot{\xi}}^2 - \mathbf{s}_{\dot{\xi}\ddot{\xi}}^2
\end{bmatrix} = \begin{bmatrix}
X_{\xi\xi} & X_{\xi\dot{\xi}} \\
X_{\xi\dot{\xi}} & X_{\dot{\xi}\dot{\xi}}
\end{bmatrix}^{-1} \begin{bmatrix}
\mathbf{s}_{\xi\xi} - \mathbf{s}_{\xi\dot{\xi}} \\
\mathbf{s}_{\xi\dot{\xi}} - \mathbf{s}_{\dot{\xi}\dot{\xi}}
\end{bmatrix}. (296)
\]

Finally, this position-velocity-acceleration consistency constraint readily generalizes to the \( p \)th-order case.

7.4 The Abolishment of the “Preferred Ordering” by the EPOT

The “consistent” EPOT abolishes the preferred ordering requirement in the POT. Figure 54 illustrates such using the sample mean-errors and “sample confidence intervals.” The corresponding results for the Basic-EPOT used in Chapter 6, are provided by Figure 55.
Figure 54 Comparison between the AR-EPOT and RA-EPOT (consistent case)
Figure 55 Comparison between the AR-EPOT and RA-EPOT (inconsistent case)
8 Concluding Remarks

In this Chapter the key results presented above are summarized; and two follow-on research activities are proposed.

8.1 Summary of this Research.
As mentioned in the introduction to this Dissertation, most radar tracking practitioners prefer to use some variation of the PLKF, and not the EKF. But the PLKF makes the estimates more noisy; and its tracks are biased. And it was shown here that the leading “debiasing” methods for the PLKF can make the tracks even more biased. Fortunately, the POT eliminates most of the EKF errors; and under it EKF tracks are generally better than the corresponding PLKF ones (less biased and less noisy). The POT, however, has an obvious limitation: by definition, its tracks must be updated using azimuth first and range last. Here it was shown here that the POT is not very effective at short ranges.

This Dissertation has provided a new version of the POT, dubbed the Extended-POT (EPOT). Not only is the EPOT more efficient than the POT at short ranges, it overcomes the fundamental limitation of the POT – the independent measurement components of a detection can now be used recursively by an EKF in any order, with virtually the same results. The key results are summarized as follows.

• Definition of the Problem. In Chapter 3 a variation of the Julier and Uhlmann CP “exemplar” (a more stressing one) was used to show that PLKF tracks tend to be biased, and more noisy than those of the LKF. Moreover, it was shown that two distinct PLKF bias cases exist, scalar- and matrix-weight, whose biases have the opposite sense to one another. The EKF performed better than the PLKF, but it was seen to have a convergence problem.
• **Analysis of the PLKF.** In Chapter 4 it was shown that the UCCM “debiasing” can make the PLKF biases worse. And it was also shown that the popular approximations for the “consistent” covariance matrix can lead to “inconsistent” gain matrices. But a remedy for that was provided – and that modified UCCM was seen to be less biased, albeit slightly more noisy.

• **Analysis of the EKF.** In Chapter 5, using the CP exemplar of Julier and Uhlmann, it was shown that the POT is not efficient at very short ranges. And so a new method was introduced, dubbed the *Extended-POT* (EPOT). That EPOT (the Basic-EPOT) was seen to perform much better than the POT, and better than the “debiased and consistent” PLKF, on the Julier and Uhlmann CP exemplar.

• **Application to Radar Tracking.** In Chapter 6 the CV exemplar of Lerro and Bar Shalom (from the DCCM/UCCM literature) was used to compare the effectiveness of the “debiased and consistent” PLKF with those of the EKF (with and without the POT and EPOT). There the EKF with the POT was best; and the Basic-EPOT was worse than the POT.

• **Analysis of the EPOT.** In Chapter 7 the various methods that were illustrated in the previous Chapters were analyzed further, and a certain position-velocity *consistency condition* was derived. The LKF, PLKF, EKF, and POT inherently satisfy that condition, while the CP version of the EPOT given in Chapter 5 does not. And so a remedy was given for the CV case (and for the CA case). Whereupon, the performance of the EPOT became comparable to that of the POT. Moreover, it was shown that that consistent EPOT overcomes the “preferred ordering” limitation of the POT.
8.2 Proposed Activities for Follow-On Research

Here two follow-on research topics are proposed. One addresses the efficacy of DCCM/UCCM “debiasing” when the observation geometry is nonstationary; and the other deals with the maximum likelihood aspect of radar tracking. In each case, first the problem is illustrated, and then the proposed follow-on activity is defined.

8.2.1 The Effect of Observation Geometry Non-Stationarity

Let the object in the UCCM CV exemplar be initially inbound, moving south (along an axis parallel to the $Y$-axis), such that its range at closest approach is one kilometer.

Figure 56 illustrates this case for the EPOT (with $\sigma_A = 1.5$ and zero system noise), using 500 Monte Carlo trials. The left-hand side shows the detections and tracks in $E$; and the right-hand side shows the detections and tracks converted back into radar coordinates. Note that the spread of the measurements shrinks in cross-range as the object approaches the radar, and expands in cross-range as the object moves away.

![Figure 56 EPOT estimates for air and missile defense scenario ($\sigma_A = 1.5$)](image)
Figure 57 provides the corresponding *rms-error* (rmse) of the estimated position radar coordinates for the M-PLKF, POT, and EPOT, with the “debiased” PLKF using the true covariance matrices of the rectangular pseudo-measurements. (Here the *rmse* is defined as the square-root of the sum of the sample mean-error squared plus the sample variance, \(\sqrt{\hat{\mu}^2 + \hat{\sigma}^2}\).) The corresponding results for \(\sigma_\alpha = 2.5\) degrees are provided in Figure 58.
Note that all the rmse’s are monotone decreasing as more detections are processed (except in the neighborhood of \( a = 0 \)). There the PLKF errors are monotone decreasing over the incoming segment \((a > 0)\), and they are monotone increasing over the outgoing segment \((a < 0)\). Note also that this example the rmse at \( a = 0 \) has a “spike: such is caused by the rotation of the covariance matrix as the object moves past the radar. In this example the principal axes of the measurements' covariance matrix rotate ninety degrees as the object approaches the radar, and they rotate another ninety degrees as the object moves away. Most of that rotation occurs from +60 to −60 degrees azimuth. And only four detection opportunities between those limits (overall, there are 156 detection opportunities).

Accordingly, it is conjectured that the efficacy of the PLKF “debiasing” methods wane as the covariance matrix of the measurements is not constant (the non-stationary case). The objective of this proposed research activity is to verify that loss exists, and to quantify its effects on the tracks.

8.2.2 Evaluation of the PLKF and EKF in the Maximum Likelihood Sense

The LKF is often said to be optimal in the unbiased and minimum variance sense. However, its behavior in the maximum likelihood sense should may also be considered. For example, consider CV exemplar, but with the object motionless: the object is fixed at \( r = 70 \) kilometers and \( a = 0 \) degrees, with \( \dot{r} = \dot{a} = 0 \); and its (constant) position and (zero) velocity are to be estimated. Figure 59 provides the sets of tracks that are determined by the PLKF, EKF, and the EKF with the POT and EPOT. The respective sets of tracks are overlaid upon the given measurements (the top plots are in \( \mathbb{E} \) (Euclidean space) and the lower plots are in \( \mathbb{R} \) (radar coordinate space).
Accordingly, it is conjectured that “debiased and consistent” PLKF tracks are more likely to be biased toward the radar; and that EKF tracks (without the POT or EPOT) are more likely to be biased away from the radar. It is also believed that the tracks determined under the POT and EPOT are less likely to be biased. Accordingly, the objective of this research activity is to provide a maximum likelihood analysis of the residual “debiased and consistent” PLKF biases, and the EKF (POT and EPOT) biases.
### 8.3 Contributions of this Work

The principal contribution of this work is the Extended-Preferred Ordering Theorem (EPOT) algorithm which enables the Extended Kalman Filter (EKF) to use radar measurements of range and azimuth in polar coordinates to update a track in rectangular coordinates, with significantly improved performance, as compared with direct use of EKF or the POT algorithm, which is current practice. An EKF azimuth update without the use of EPOT or POT introduces a spurious estimation error in range, which can be several orders of magnitude worse than the range measurement error.

Reference [38]:


The EPOT is an improvement over the current practice of using the Preferred Ordering Theorem (POT) for three reasons. One, the POT only removes some of the EKF’s spurious estimation errors in range. The EPOT removes all those errors. Two, POT must use the measurement components of detections in the order azimuth-first and range-last. (The opposite order, range-first and azimuth-last, is basically the same as the EKF update without the POT.) The EPOT can use either order; and, in fact, can eliminate the range measurement entirely – for example, angle-only measurements from passive sensors can now be used to update radar tracks without making them worse. Three, the EPOT is more effective than the POT when the object being tracked is at short range. Indeed, at very short ranges the POT offers no improvement over the EKF, while there the EPOT still removes all the spurious linearization errors in range.
Appendix

Here the Best Linear Unbiased Estimator (BLUE) is derived; and its various forms that are used in this work are provided: the batch form; and the recursive fusion and update forms. Such is taken from [7].

By definition, a linear estimate of \( y \) given \( z \) has the form \( \hat{y} =Az \), where \( A \) is a linear transformation that is independent of \( z \). Here \( y \in Y \) is some unknown to be determined, and \( z =Hy + \tilde{z} \), with \( z =Hy \) a linear transformation; and the measurement error, \( \tilde{z} \), is independent of \( z \). For example, when \( (H^TH)^{-1} \) exists, the linear least-squares estimation matrix is \( A = (H^TH)^{-1}H^T \). But, to define the best linear unbiased estimator, two random vectors are needed, say \( Y \) and \( Z \), where \( Z \) is a realization of \( Z \) and \( \hat{y} \) is a realization of \( Y \). In which case, \( Z =Hy + \tilde{Z} \), with \( \tilde{Z} \) is independent of \( Hy \).

The measurement is then said to be unbiased if \( E(Z) =Hy \), equivalently \( E(\tilde{Z}) =0 \); and the estimate is said to be unbiased if \( E(Y - y) =0 \). The estimator \( Y =AZ \) is the Best Linear Unbiased Estimator (BLUE): if \( Y \) is unbiased when \( Z \) is unbiased; and, if \( A \) is chosen such that the trace of \( \Sigma_Y = \text{cov}(Y) \) is minimized (over all possible nontrivial \( A \)'s).

Here a sequence of such \( Z \)'s is given, \( Z_n =H_ny + \tilde{Z}_n, \ n =1,2,\ldots,N \), and the linear estimator has the form

\[
Y_N = A_1Z_1 + A_2Z_2 + \cdots + A_NZ_N .
\]  

(297)

And so for \( Y_N \) to be unbiased,

\[
\mu_Y(N) \equiv E(Y_N) = \sum_{n=1}^{N} A_nE(Z_n) = \sum_{n=1}^{N} A_nH_ny = y .
\]
Which implies
\[ \sum_{n=1}^{N} A_n H_n = I. \] (298)

By definition, the covariance matrix of this estimate is
\[ \Sigma_N = E \left( \sum_{m=1}^{N} A_m Z_m - E \sum_{m=1}^{N} A_m Z_m \right) \left( \sum_{n=1}^{N} A_n Z_n - E \sum_{m=1}^{N} A_m Z_m \right)^T. \] (299)

For the BLUE, the \( A_n \)'s are then chosen such that the trace of (300) is minimized subject to (298) as a constraint.

For convenience, let the measurements be unbiased and mutually independent, with \( \Sigma_Z = \text{cov} Z_n, n = 1, 2, \cdots, N \), symmetric positive definite. The \( Z_n \)'s are then mutually orthogonal; and (299) becomes
\[ \Sigma_N = \sum_{n=1}^{N} A_n \Sigma_Z A_n^T. \] (300)

Now let \( \Lambda \) be a non-singular Lagrange multiplier matrix, and consider
\[ \lambda = \text{tr} \left( \sum_{n=1}^{N} A_n \Sigma_Z A_n^T \right) - 2 \text{tr} \left( \Lambda^T \sum_{n=1}^{N} A_n H_n \right). \] (301)

For each \( n = 1, 2, \cdots, N \), the derivative of this scalar \( \lambda \) with respect to the matrix \( A_n \) is
\[ \frac{\partial \lambda}{\partial A_n} = 2 A_n \Sigma_Z A_n^T - 2 \Lambda H_n^T. \]

Setting these expressions equal to the corresponding zero matrices yields
\[ A_n = \Lambda H_n^T \Sigma_Z^{-1}(n). \] (302)

And substituting the results back into (300), and noting that
\[ A_n \Sigma_Z A_n^T = \Lambda H_n^T A_n^T = H_n A_n^T \] (303)
(recall that \( \Sigma_Z \) is symmetric), leads to
\[ \Sigma_y(N) = \Lambda \sum_{n=1}^{N} H_n^T A_n = \sum_{n=1}^{N} A_n H_n \Lambda^T. \]  

(304)

But using (298) in (304) implies

\[ \Sigma_y(N) = \Lambda = \Lambda^T. \]  

(305)

And post-multiplying both sides of (302) by \( H_n \), and summing, yields

\[ \Lambda \sum_{n=1}^{N} H_n^T \Sigma^{-1}_Z(n) H_n = \sum_{n=1}^{N} A_n H_n = I. \]

Thus,

\[ \Sigma_y(N) = \left( \sum_{n=1}^{N} H_n^T \Sigma^{-1}_Z(n) H_n \right)^{-1} = \Lambda. \]  

(306)

Here \( \Sigma_y(N) \) is assumed to be nonsingular – see the discussion following (310).

The \( A_n \)'s that minimize the trace of (300) subject to (298) are then

\[ A_n = \left( \sum_{n=1}^{N} H_n^T \Sigma^{-1}_Z(n) H_n \right)^{-1} H_n^T \Sigma^{-1}_Z(n) = \Sigma_y(N) H_n^T \Sigma^{-1}_Z(n), \]  

(307)

and the BLUE is therefore

\[ \hat{Y}_N = \sum_{n=1}^{N} A_n Z_n = \left( \sum_{n=1}^{N} H_n^T \Sigma^{-1}_Z(n) H_n \right)^{-1} \sum_{n=1}^{N} H_n^T \Sigma^{-1}_Z(n) Z_n. \]  

(308)

Accordingly, the estimate and covariance matrix determined by the BLUE are

\[ \hat{y}_N = \Sigma_y(N) \sum_{n=1}^{N} H_n^T \Sigma^{-1}_Z(n) Z_n \quad \text{and} \quad \Sigma_y(N) = \left( \sum_{n=1}^{N} H_n^T \Sigma^{-1}_Z(n) H_n \right)^{-1}. \]  

(309)

The form of (309) is called the “batch” BLUE: all the measurements are being used at once, concurrently, in a single batch, to determine the estimate. However, the BLUE may also be determined recursively. For example, given the batch form of the BLUE in (309), determine the sequences
\[
\Sigma^{-1}_Y(n) \hat{y}_n = \sum_{k=1}^{n} H^T_k \Sigma^{-1}_Z(k) \bar{z}_k \quad \text{and} \quad \Sigma^{-1}_Y(n) = \sum_{k=1}^{n} H^T_k \Sigma^{-1}_Z(k) H_k , \quad (310)
\]

\( n = 1, 2, \ldots, N \). For convenience, let \( \Sigma^{-1}_Y(0) = 0 \), and use the second expression in (310) to formally define “\( \Sigma^{-1}_Y(n) \).” Whereupon, for \( n = 1, 2, \ldots, N \), determine (310) recursively as
\[
\Sigma^{-1}_Y(n) \hat{y}_n = \Sigma^{-1}_Y(n-1) \hat{y}_{n-1} + H^T_n \Sigma^{-1}_Z(n) \bar{z}_n \quad (311)
\]
and
\[
\Sigma^{-1}_Y(n) = \Sigma^{-1}_Y(n-1) + H^T_n \Sigma^{-1}_Z(n) H_n . \quad (312)
\]

But for the batch BLUE estimate to exist, \( \Sigma_Y(N) \) must be nonsingular. That is, a smallest \( n \leq N \) must exist, say \( m \), for which \( \Sigma_Y(m) \) is nonsingular. Accordingly, invert (312) at that \( m \), and then multiply \( \Sigma^{-1}_Y(m) \hat{y}_m \) by \( \Sigma_Y(m) \). In which case
\[
\hat{y}_m = \Sigma_Y(m) \left[ \Sigma^{-1}_Y(m-1) \hat{y}_{m-1} + H^T_n \Sigma^{-1}_Z(m) \bar{z}_m \right] . \quad (313)
\]
And then, for \( m < n \), substitute (312) as \( \Sigma^{-1}_Y(n-1) = H^T_n \Sigma^{-1}_Z(n) H_n - \Sigma^{-1}_Y(n) \) into (313), and simplify. Whereupon, the recursive form of the BLUE is obtained,
\[
\hat{y}_n = \hat{y}_{n-1} + \Sigma_Y(n) H_n^T \Sigma^{-1}_Z(n) \left( \bar{z}_n - H_n \hat{y}_{n-1} \right) \quad (314)
\]
and
\[
\Sigma_Y(n) = \left[ \Sigma^{-1}_Y(n-1) + H^T_n \Sigma^{-1}_Z(n) H_n \right]^{-1} . \quad (315)
\]

The more commonly used form of the recursive BLUE is
\[
\hat{y}_n = \hat{y}_{n-1} + K_n \left( \bar{z}_n - H_n \hat{y}_{n-1} \right) \quad \text{and} \quad \Sigma_Y(n) = \left( I - K_n H_n \right) \Sigma_Y(n-1) \quad (316)
\]
where
\[
K_n = \Sigma_Y(n-1) H_n^T \left[ H_n \Sigma_Y(n-1) H_n^T + \Sigma_Z(n) \right]^{-1} . \quad (317)
\]
These two recursive forms are related by the so-called matrix inversion lemma [59]. And the last expression, (317), is called the gain matrix— in (312) it is $\Sigma_y(n)H_n^T\Sigma_z^{-1}(n)$. The recursive form given by (314) and (315) is more amenable to analysis. The recursive form given by (316) and (317) is usually less computationally burdensome.
References


[38] D. M. Leskiw and H Wang, “The ‘Extended’ Preferred Ordering Theorem for Precision Acquisition, Tracking, and Pointing,” Proceedings of AeroSense:


VITA

NAME OF AUTHOR:  Donald Myron Leskiw

PLACE OF BIRTH:  Newark, New Jersey

DATE OF BIRTH:  October 25, 1952

GRADUATE AND UNDERGRADUATE SCHOOLS ATTENDED:

• Newark College of Engineering (NCE)
• New Jersey Institute of Technology (NJIT)
• New York University
• Harvard University

DEGREES AWARDED:

• Masters of Science in Electrical Engineering, 1977, NJIT
• Bachelor of Science in Electrical Engineering, 1974, NCE

AWARDS AND HONORS:

• US Patent on Single-Scan Track Initiation, March 2009
• NASA Group Achievement Award on Hypercube Program

PROFESSIONAL EXPERIENCE:

• Principal Member of Research Staff, Riverside Research Institute, New York (1977-1984)
• Self-employed as consultant and small business owner, with 50 employees (1985-2001)
• Senior Project Manager, NPAC, Syracuse University (1994-1997)
• Radar Systems Engineering Consultant to Sensis Corporation (1999-2001)
• Systems Engineer, Senior Staff, Lockheed Martin Corporation (2001-2010)
• Self-employed, Leskiw Associates, LLC (2010-Present)