UNIVERSITÀ DI PISA

Scuola di Dottorato in Ingegneria "Leonardo da Vinci"

Corso di Dottorato di Ricerca in
INGEGNERIA DELL'INFORMAZIONE

Tesi di Dottorato di Ricerca

This work has been supported by SESM-Finmeccanica and SELEX Sistemi Integrati

THEORETICAL ASPECTS AND REAL ISSUES IN AN INTEGRATED MULTIRADAR SYSTEM

Autore:

Stefano Fortunati _____

Relatori:

Prof. Fulvio Gini _____

Dr. Alfonso Farina _____

Prof. Maria Sabrina Greco _____

Anno 2011
Settore scientifico-disciplinare: ING-INF/03

To my family and to all my friends

Acknowledgments

I would like to express my sincere gratitude to Prof. Fulvio Gini and Dr. Alfonso Farina for the opportunity to write this thesis under their guidance. I'm also very grateful to them for their scientific support during the PhD course and for the numerous stimulating discussions.

I thank also Prof. Maria Sabrina Greco, Dr. Antonio Graziano and Dr. Sofia Giompapa for their help and for their comments since the beginning of this activity.

Contents

Preface…… VIII

Part I: Airborne radar tracking in a real environment……..... 1

Chapter 1: Tracking in the presence of atmospheric turbulences 2

1.1 Introduction…… 2

1.2 Geometry of the scenario…… 4

1.3 The Dryden model……... 6

1.3.1 Statistical characterization of the Dryden acceleration process…… 8

1.4 Target State Model…….. 9

1.4.1 Continuous-Time Target State Model……... 10

1.4.2 Discrete-Time Target State Model…… 11

1.5 Filter model…… 14

1.5.1 Performance bound for the ideal case……... 15

1.6 Simulation model and data generation…… 16

1.7 Simulation results…… 17

1.8 Summary…… 23

References…… 24

Chapter 2: Correction of the troposheric refraction effects……... 27

2.1 Introduction…… 27

2.2 Geometry of the scenario…… 29

2.3 Mathematical model for tropospheric propagation …… 30

2.3.1 Generation of the refracted ray path…….. 32

2.3.2 Generation of range and elevation errors…….. 34

2.4 Evaluation of the tropospheric errors…….. 35

2.4.1 Evaluation of tropospheric elevation error…… 35

2.4.2 Evaluation of tropospheric range error…….. 36

2.5. Modified KF for tropospheric error correction…….. 40

2.5.1 The discrete-time model for target motion and the measurement model….. 41

2.5.2 Modified Kalman Filter for tropospheric error correction…… 42

2.6 Simulation results…… 45

2.7 Summary…… 54

References…… 56

Appendix A……... 59

Appendix B……... 60

Part II: The grid-locking problem…….. 63

Chapter 3: The relative grid-locking problem……... 64

3.1 Introduction…… 64

3.2 The relative grid-locking problem……... 69

3.3 The measurements model…… 72

3.4 Target kinematic model……... 73

3.5 The linear least squares (LS) algorithm…….. 74

3.6 The Expectation-Maximization (EM) algorithm…… 78

3.6.1 The Expectation-Maximization algorithm: a brief outline..…….. 78

3.6.2 Application of the EM algorithm to the relative grid-locking problem…… 80
3.7 Performance bound……. 86
3.8 Numerical analysis…….. 90
3.9 Generalization to the multi-target scenario……. 106
3.9.1 Performance bound for the multi-target scenario…….. 108
3.9.2 Simulation results…….. 111
3.10 The identifiability problem…… 116
3.10.1 General formulation of the identifiability problem……. 116
3.10.2 Identifiability in presence of random nuisance parameters……. 119
3.10.3 Relationship among the identifiability conditions in presence of random nuisance parameters…… 122
3.10.4 Identifiability in the relative grid-locking problem……. 124
3.11 Summary……... 124
References……. 125

Chapter 4: The absolute grid-locking problem…….. 129
4.1 Introduction……. 129
4.2 Absolute grid-locking problem……... 129
4.2.1 Linear Least Squares estimator for the absolute grid-locking problem…… 133
4.3 The absolute grid-locking problem in the multi-sensor-multi-target scenario… 136
4.4 Performance bound for multi-sensor-multi-target scenario…… 138
4.5 Simulation results…… 141
4.6 Summary……. 149
References……. 149
Appendix C……... 151
Appendix D……... 152
Appendix E…… 155

Part III: Intrinsic covariance matrix estimation and its application to the radar target detection……... 158
Chapter 5: Intrinsic estimation in the manifold of the symmetric positive-definite matrices and its applications to the radar target detection……. 159
5.1 Introduction……. 159
5.2 Preliminaries: the exponential mapping…….. 160
5.3 Geometrical characterization of a Riemann manifolds……... 162
5.4 Manifold of the symmetric positive-definite matrices…… 164
5.5 Distance between two points on *P*…….. 165
5.6 Characterization of two intrinsic mean operators……... 166
5.6.1 The Karcher-Fréchet (KF) mean……... 166
5.6.2 The Log-Euclidean (LE) mean…….. 170
5.6.3 Comparison between the KF and the LE means……... 172
5.7 Radar target detection in the presence of additive clutter……... 172
5.7.1 Comparison between the classical decision criterion and the Riemann distance based criterion…….. 174
5.8 Comparison among covariance matrix estimation algorithms…… 178
5.8.1. Sample Covariance Matrix (SCM)……... 178
5.8.2 Matrix Means…… 179

5.8.3 Burg-based estimator……. 183
5.8.4 Intrinsic and Flat Cramér-Rao Lower Bounds…….. 183
5.8.5 Comparison among the covariance matrix estimators performance……. 184
5.9 Detection performance…… 188
5.10 Summary……... 192
References……. 192
Concluding remarks……... 195

Preface

In the last few years Homeland Security (HS) has gained a considerable interest in the research community. From a scientific point of view, it is a difficult task to provide a definition of this research area and to exactly draw up its boundaries. In fact, when we talk about the security and the surveillance, several problems and aspects must be considered. In particular, the following factors play a crucial role and define the complexity level of the considered application field: the number of potential threats can be high and uncertain; the threat detection and identification can be made more complicated by the use of camouflaging techniques; the monitored area is typically wide and it requires a large and heterogeneous sensor network; the surveillance operation is strongly related to the operational scenario, so that it is not possible to define a unique approach to solve the problem [1].

Information Technology (IT) can provide an important support to HS in preventing, detecting and early warning of threats. Even though the link between IT and HS is relatively recent, sensor integration and collaboration is a widely applied technique aimed to aggregate data from multiple sources, to yield timely information on potential threats and to improve the accuracy in monitoring events [2]. A large number of sensors have already been developed to support surveillance operations. Parallel to this technological effort in developing new powerful and dedicated sensors, interest in integrating a set of stand-alone sensors into an integrated multi-sensor system has been increasing. In fact, rather than to develop new sensors to achieve more accurate tracking and surveillance systems, it is more useful to integrate existing stand-alone sensors into a single system in order to obtain performance improvements

In this dissertation, a notional integrated multi-sensor system acting in a maritime border control scenario for HS is considered. In general, a border surveillance system is composed of multiple land based and moving platforms carrying different types of sensors [1]. In a typical scenario, described in [1], the integrated system is composed of a land based platform, located on the coast, and an airborne platform moving in front of the coast line. In this dissertation, we handle two different fundamental aspects.

In Part I, we focus on a single sensor in the system, i.e. the airborne radar. We analyze the tracking performance of such a kind of sensor in the presence of two different

atmospheric problems: the turbulence (in Chapter 1) and the tropospheric refraction (in Chapter 2). In particular, in Chapter 1, the losses in tracking accuracy of a turbulence-ignorant tracking filter (i.e. a filter that does not take into account the effects of the atmospheric turbulences) acting in a turbulent scenario, is quantified. In Chapter 2, we focus our attention on the tropospheric propagation effects on the radar electromagnetic (em) signals and their correction for airborne radar tracking. It is well known that the troposphere is characterized by a refractive index that varies with the altitude and with the local weather. This variability of the refractive index causes an error in the radar measurements. First, a mathematical model to describe and calculate the em radar signal ray path in the troposphere is discussed. Using this mathematical model, the errors due to the tropospheric propagation are evaluated and the corrupted radar measurements are then numerically generated. Second, a tracking algorithm, based on the Kalman Filter, that is able to mitigate the tropospheric errors during the tracking procedure, is proposed.

In Part II, we consider the integrated system in its wholeness to investigate a fundamental prerequisite of any data fusion process: the sensor registration process. The problem of sensor registration (also termed, for naval system, the *grid-locking* problem) arises when a set of data coming from two or more sensors must be combined. This problem involves a coordinate transformation and the reciprocal alignment among the various sensors: streams of data from different sensors must be converted into a common coordinate system (or frame) and aligned before they could be used in a tracking or surveillance system. If not corrected, registration errors can seriously degrade the global system performance by increasing tracking errors and even introducing ghost tracks. A first basic distinction is usually made between *relative* grid-locking and *absolute* grid-locking. The relative grid-locking process aligns remote data to local data under the assumption that the local data are bias free and that all biases reside with the remote sensor. The problem is that, actually, also the local sensor is affected by bias. Chapter 3 of this dissertation is dedicated to the solution of the relative grid-locking problem. Two different estimation algorithms are proposed: a linear Least Squares (LS) algorithm and an Expectation-Maximization-based (EM) algorithm. The linear LS algorithm is a simple and fast algorithm, but numerical results have shown that the LS estimator is not efficient for most of the registration bias errors. Such non-efficiency could be caused by the linearization implied by the linear LS algorithm. Then, in order to obtain a more efficient estimation

algorithm, an Expectation-Maximization algorithm is derived. In Chapter 4 we generalize our findings to the absolute grid-locking problem.

Part III of this dissertation is devoted to a more theoretical aspect of fundamental importance in a lot of practical applications: the estimate of the disturbance covariance matrix. Due to its relevance, in literature it can be found a huge quantity of works on this topic. Recently, a new geometrical concept has been applied to this estimation problem: the Riemann (or intrinsic) geometry. In Chapter 5, we give an overview on the state of the art of the application of the Riemann geometry for the covariance matrix estimation in radar problems. Particular attention is given for the detection problem in additive clutter. Some covariance matrix estimators and a new decision rule based on the Riemann geometry are analyzed and their performance are compared with the classical ones.

[1] Sofia Giompapa, "Analysis, modeling, and simulation of an integrated multi-sensor system for maritime border control", PhD dissertation, University of Pisa, April 2008.

[2] H. Chen, F. Y. Wang, and D. Zeng, "Intelligence and security informatics for Homeland Security: information, communication and transportation," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 5, no. 4, pp. 329-341, December 2004.

Part 1: Airborne radar tracking in a real environment

Chapter 1: Tracking in the presence of atmospheric turbulences

1.1 Introduction

In the last few years, the technology for identification, border security and controlled access to critical infrastructures has become a very important concern to prevent unexpected attacks. As pointed out in [1], [2], [3], and [4], a key sensor in an integrated control system is the airborne radar due to its capacity to cover a wide area. One of the problems for airborne radars is the atmospheric turbulence. The atmosphere is driven into intricate motion by Earth rotation and solar heating, and these phenomena give rise to a variety of complex thermodynamic, chemical and electromagnetic processes. The phenomenon of turbulence has been widely addressed in the aerospace literature, see e.g. [5], [6], [7]. In the aerospace field, the study of the turbulence effects is of fundamental importance in a lot of different aspects [5]: improvements the aerodynamic and structural analysis, prediction of the expected behaviour of an aircraft under various levels of turbulence, evaluation of the stability of onboard sensing equipment, and so on. Due to the extreme complexity of the turbulence phenomena and due to the huge variety of applications, there is not a unique full-comprehensive model for the atmospheric turbulence, but there exist a wide variety of different and simplified models. In [6], numerous turbulence models are enumerated and described. However, the most commonly adopted model to study the impact of the turbulent wind gust on the aircraft is the Dryden model ([5], [6], [7], [8]). According to this model, the atmospheric turbulence is modelled as a random velocity process added to the aircraft velocity vector described in a body-fixed Cartesian coordinate system. The turbulence velocity processes are assumed to be correlated, zero-mean, Gaussian-distributed random processes whose Power Spectral

Densities (PSDs) have been empirically found by fitting measured data. The Dryden model assumes that the turbulent gusts are homogeneous and isotropic. The assumption of homogeneity implies that the statistical properties of the turbulence are the same for each point of the air mass. The assumption of isotropy means that the statistical properties of the air mass do not depend upon the spatial orientation of the air mass itself. There is, however, a dependence upon the orientation of the aircraft because of its motion through the gust field [7]. Even if the turbulence problem is a well studied problem in the aerospace field, its application to the radar signal processing is still a challenging issue.

The aim of this Chapter is to quantify the losses in tracking accuracy of a turbulence-ignorant tracking filter (i.e. a filter that does not take into account the effects of the atmospheric turbulences) in a turbulent simulated scenario. First, we analytically derive both continuous-time and discrete-time target state model and the terms due to the Dryden disturbances are discussed and statistically characterized. Second, the problem of the generation of the data vector in the presence of atmospheric turbulence is addressed in order to simulate a turbulent scenario. Finally, a turbulent-ignorant tracking algorithm, based on the Kalman Filter (KF), is implemented in order to evaluate the loss in tracking accuracy with respect to the ideal case (absence of atmospheric turbulence). To this end, the error standard deviation for the estimate of each component of the target state vector is compared to the Posterior Cramér-Rao Lower Bound (PCRLB), evaluated for the ideal case. It is important to note that the possibility to correct the effects of the turbulences on the aircraft flight using some navigation devices (such as compass or GPS) is not taken into account here.

The airborne radar is assumed to be part of a multisensor system ([1]-[4]) for maritime and border surveillance. For this reason, the chosen racetrack course is a quasi-ellipsoidal trajectory near the surveillance area. The rest of the Chapter is organized as follows. In Section 1.2, a suitable scenario, based on [1], [2], [3], [4], and [9], is chosen as an example. The geometrical parameters of the aircraft course are given and the characteristics of both radar and target are described. Section 1.3 provides the analytical characterization of the Dryden model; the complete continuous-time and discrete-time target state models are derived in Section 1.4. The tracking filter is described in Section 1.5, where the derivation of the PCRLB for the ideal case is also addressed. In Section 1.6, the procedure to generate

the turbulence data is described. Some simulation results are presented in Section 1.7. Finally, our conclusions are reported in Section 1.8.

1.2 Geometry of the scenario

To derive a geometrical and kinematical description of both the airborne platform and target motion, we have to introduce two different reference systems. Assuming a flat Earth model, an inertial Cartesian reference system (x, y, z) is used to describe the kinematic equations of the aircraft and target. Since the airborne radar acquires the measurements in its local reference system, to describe the target motion in a suitable way to derive the target state model, we need a relative, aircraft-centred Cartesian coordinate system, named (x', y', z'). The inertial and the relative aircraft-centred coordinate systems are always parallel during the aircraft flight. Here, we consider a 3D Airborne Early Warning Radar (AEWR) [10] which measures the range, azimuth and elevation with the accuracy given by σ_ρ, σ_θ and σ_ε, respectively, a scan time of T seconds and a flight altitude h. The numerical values of all these parameters are given in Section VII. Without loss of generality, the chosen aircraft course is a quasi-ellipsoidal racetrack. In the following, taking into account fig. 1.1, the main geometrical parameters of the scenario are enumerated:

- $\mathbf{O}_a = \left(O_{a,x}, O_{a,y}, O_{a,z}\right)$: centre of the aircraft racetrack course, then $O_{a,z} = h$;
- d: length of the straight segment of the racetrack;
- R: radius of the circular segment of the racetrack;
- $\mathbf{A} = \left(O_{a,x} - d/2, O_{a,y} + R, h\right)$, $\mathbf{B} = \left(O_{a,x} + d/2, O_{a,y} + R, h\right)$, $\mathbf{C} = \left(O_{a,x} + d/2, O_{a,y} - R, h\right)$, $\mathbf{D} = \left(O_{a,x} - d/2, O_{a,y} - R, h\right)$;
- $\mathbf{r}(t)$: target position vector defined in the inertial reference system;
- $\mathbf{r}'(t)$: target position vector defined in the relative aircraft-centred reference system;
- $\mathbf{r}_O(t)$: aircraft position vector defined in the inertial reference system.

From this geometrical setting, it is easy to get the kinematic description of the aircraft motion in the inertial reference system. The aircraft position vector can be expressed as:

$$\mathbf{r}_O(t)=\begin{cases}\begin{pmatrix}A_x\\A_y\\h\end{pmatrix}+\begin{pmatrix}v_p t'\\0\\0\end{pmatrix} & t'\in[0,T_l]\\ \begin{pmatrix}B_x\\B_y-R\\h\end{pmatrix}+\begin{pmatrix}R\sin\left(\varpi\left(t'-T_l\right)\right)\\R\cos\left(\varpi\left(t'-T_l\right)\right)\\0\end{pmatrix} & t'\in[T_l,T_c+T_l]\\ \begin{pmatrix}C_x\\C_y\\h\end{pmatrix}+\begin{pmatrix}-v_p\left(t'-\left(T_l+T_c\right)\right)\\0\\0\end{pmatrix} & t'\in[T_l+T_c,T_c+2T_l]\\ \begin{pmatrix}D_x\\D_y+R\\h\end{pmatrix}-\begin{pmatrix}R\sin\left(\varpi\left(t'-2T_l-T_c\right)\right)\\R\cos\left(\varpi\left(t'-2T_l-T_c\right)\right)\\0\end{pmatrix} & t'\in[2T_l+T_c,2T_c+2T_l]\end{cases} \tag{1.1}$$

where $t'=\mathrm{mod}(t,2(T_l+T_c))$, v_p is the aircraft speed and T_l and T_c are the time intervals needed to cover the straight and circular segments of the racetrack course respectively. The aircraft angular speed ϖ can be expressed as $\varpi=\pi/T_c=v_p/R$. The aircraft instantaneous velocity $\mathbf{v}_{O'}(t)$ and acceleration $\mathbf{a}_{O'}(t)$ can be obtained straightforwardly by differentiation of eq. (1.1):

$$\mathbf{v}_O(t)=\dot{\mathbf{r}}_O(t), \tag{1.2}$$

$$\mathbf{a}_O(t)=\ddot{\mathbf{r}}_O(t). \tag{1.3}$$

As concerning the target kinematic model, for the ease of mathematical formulation of the problem involving the atmospheric turbulences, the target motion is assumed rectilinear and uniform. Then, the target position vector in the inertial reference system can be described by the following linear equation:

$$\mathbf{r}(t)=\mathbf{r}(0)+\mathbf{v}_t t, \tag{1.4}$$

where $\mathbf{r}(0)$ is the starting point of the track and $\mathbf{v}_t$ is the constant velocity vector of the target.

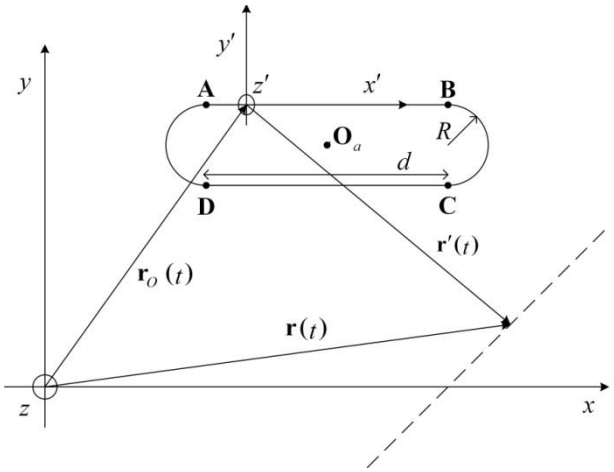

Figure 1.1 - Geometry of the scenario. Axes z and z' are orthogonal to the aircraft race plane.

1.3 The Dryden model

The Dryden model is one of the most useful and tractable models for the atmospheric turbulence. To define it, we need a body-fixed reference frame attached to the aircraft centre of gravity which moves with the aircraft. The *x* axis is positioned in the direction of motion, the *y* axis is positioned along the wings and the *z* axis is perpendicular to the aircraft plane. Then, the turbulence is modelled by adding some random components to the aircraft velocity defined in this body-fixed coordinate system. Such random terms are indicated by $v_u(t)$, $v_v(t)$, and $v_o(t)$ and represent the velocity components along the *x*, *y*, and *z* axes, respectively, of the body-fixed reference frame. In the Dryden model such continuous-time random processes are modelled as zero-mean, Gaussian-distributed processes whose PSDs have the analytic form shown in eqs. (1.5)-(1.7) ([5], [7]):

$$\Phi_u(\omega) = \sigma_u^2 \frac{L_u}{\pi V_0} \frac{1}{1+\left(\frac{L_u}{V_0}\omega\right)^2}, \tag{1.5}$$

$$\Phi_v(\omega)=\sigma_v^2\frac{L_v}{2\pi V_0}\frac{1+3\left(\frac{L_v}{V_0}\omega\right)^2}{\left[1+\left(\frac{L_v}{V_0}\omega\right)^2\right]^2}, \tag{1.6}$$

$$\Phi_o(\omega)=\sigma_o^2\frac{L_o}{2\pi V_0}\frac{1+3\left(\frac{L_o}{V_0}\omega\right)^2}{\left[1+\left(\frac{L_o}{V_0}\omega\right)^2\right]^2}, \tag{1.7}$$

where V_0 is the gust wind speed in the aircraft reference system, the parameters σ_u^2, σ_v^2 and σ_o^2 depend on the level of turbulence to be simulated and are selected accordingly [6], [8]. Parameters L_u, L_v and L_o are the scale lengths for the PSDs and depend on the flight altitude [6], [8]. Fig. 1.2 shows the PSDs of (1.5)-(1.7) for $\sigma_u=\sigma_v=\sigma_o=1\,\text{m/s}$, $L_u=L_v=L_o=533.54\,\text{m}$ and V_0 = 96 m/s. To reflect higher level of turbulence, the curves would be multiplied by the desired values of σ_u^2, σ_v^2 and σ_o^2.

Before the target state model analysis, some clarification about the body-fixed reference system has to be done. When the aircraft covers the straight segment of the racetrack course, such body-fixed Cartesian reference frame coincides (except for a negligible change of orientation) with the (x', y', z') defined in the previous section. When the aircraft manoeuvres to cover the circular segment of its course, the reference frame (x', y', z') and the body-fixed one are no longer the same. The body-fixed frame is not an inertial reference frame and then the Dryden velocities are modified by the centrifugal and Coriolis accelerations [7]. In order to make the analytical characterization of the problem tractable, these effects will be neglected in this paper. Under this assumption, the turbulence velocity terms can be defined directly in the (x', y', z') reference frame. Thus, the previous assumption can be summarized in the following way:

$$v_x(t)=v_{O'x}(t)+v_u(t), \tag{1.8}$$

$$v_y(t)=v_{O'y}(t)+v_v(t), \tag{1.9}$$

$$v_z(t) = v_{O'z}(t) + v_o(t), \tag{1.10}$$

where $v_x(t)$, $v_y(t)$, and $v_z(t)$, are the components of the aircraft velocity vector in the inertial reference system. From eqs. (1.8)-(1.10), it is clear that the atmospheric turbulences tends to take the aircraft off its nominal racetrack course.

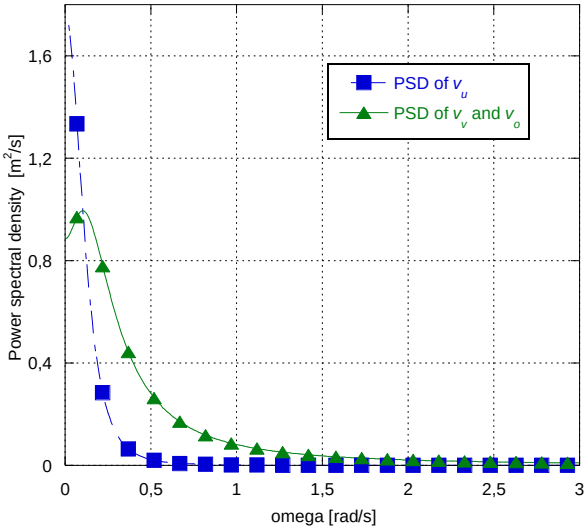

Figure 1.2 - PSD of the Dryden velocity processes.

1.3.1 Statistical characterization of the Dryden acceleration process

To the best of authors' knowledge, in the open literature there is no explicit statistical characterization of the random acceleration vector derived from the Dryden velocity vector. Since, as discussed in the next Section, knowledge of the statistics of the Dryden acceleration vector is needed to formulate a comprehensive signal model, in the following a complete characterization of this random vector is provided.

As usual, the Dryden acceleration vector can be defined as the first derivative of the Dryden velocity vector. Since the derivative is a linear operator, then the Dryden acceleration vector is itself a zero-mean, Gaussian-distributed random vector and then it is

fully characterized by its first and second order statistics. Using linear filter theory, it can be proved that the Autocorrelation Function (ACF) of the each component of the acceleration vector can be derived from the ACF of the corresponding component of the Dryden velocity vector as:

$$r_a(\gamma) = r_v(\gamma) \otimes \dot{\delta}(\gamma) \otimes \dot{\delta}(-\gamma) = -\ddot{r}_v(\gamma), \tag{1.11}$$

where $\dot{\delta}(\cdot)$ is the first derivative of a Dirac delta function [11] and $\gamma = \tau - \upsilon$. The ACFs of the Dryden velocity processes can be obtained as the inverse Fourier transform of the PSDs in eqs. (1.5)-(1.7) [7]:

$$r_{v_u}(\tau) = Ae^{-a|\gamma|}, \tag{1.12}$$

$$r_{v_v}(\tau) = r_{v_o}(\tau) = Be^{-b|\gamma|}\left(1 - \frac{b}{2}|\gamma|\right), \tag{1.13}$$

where: $A = \sigma_u^2 / 2\pi$, $B = \sigma_v^2 / 2\pi$, $a = V_0 / L_u$ and $b = V_0 / L_v$. Through double differentiation of (1.12) and (1.13) we can obtain:

$$r_{a_u}(\gamma) = -Aa^2 e^{-a|\gamma|} + 2Aa\delta(\gamma), \tag{1.14}$$

$$r_{a_v}(\gamma) = r_{a_o}(\gamma) = -Bb^2 e^{-b|\gamma|}\left(2 - \frac{b}{2}|\gamma|\right) + 3Bb\delta(\gamma). \tag{1.15}$$

1.4 Target State Model

The aim of this Section is to include the effects of the atmospheric turbulences in the signal model. This analysis is in general necessary to formulate both the filter model, i.e. the model that the tracking filter assumes for the signal, and the simulation model, i.e. the simulated signal. Although the optimal tracking performance can only be expected when the filter and the simulation model are one and the same model, in certain cases it is unfeasible to employ the simulation model in the filter. In such cases, a simplified model is assumed by the filter and this mismatch between the filter model and the simulated model leads to a loss in the tracking performances. As discussed ahead, this is exactly our case. In

fact, the analytical complexity makes the formulation of a turbulence-aware filter model intractable.

1.4.1 Continuous-Time Target State Model

In the following, we define as $\mathbf{r}'(t)$, $\mathbf{v}'(t)$ and $\mathbf{a}'(t)$ the target position, velocity, and acceleration vectors in the relative reference system. Since the inertial and the relative reference systems are considered always parallel during the flight, the coordinate transformation between them is linear. It is easy to show that the target position vector in the aircraft reference system is given by:

$$\mathbf{r}'(t) = \mathbf{r}(t) - \mathbf{r}_O(t), \tag{1.16}$$

where $\mathbf{r}(t)$ is the target position vector with respect to the inertial reference system, while $\mathbf{r}_O$(t) is defined in eq. (1.1). The relative velocity and acceleration vectors can be obtained by derivation, i.e. $\mathbf{v}'(t) = \dot{\mathbf{r}}'(t)$ and $\mathbf{a}'(t) = \ddot{\mathbf{r}}'(t)$. Defining the Dryden velocity vector as $\mathbf{v}_a(t) = (v_u(t) \quad v_v(t) \quad v_o(t))^T$, the target velocity vector in the aircraft reference system can be obtained as:

$$\mathbf{v}'(t) = \dot{\mathbf{r}}'(t) = \mathbf{v}_t - \left(\mathbf{v}_O(t) + \mathbf{v}_a(t)\right), \tag{1.17}$$

where $\mathbf{v}_t$ is the constant velocity vector of the target. The differential system associated to eq. (1.16) and eq. (1.17) is:

$$\begin{cases} \dot{\mathbf{r}}'(t) = \mathbf{v}'(t) \\ \dot{\mathbf{v}}'(t) = -\mathbf{a}_O(t) - \mathbf{a}_a(t) \end{cases}, \tag{1.18}$$

where the acceleration vector $\mathbf{a}_a(t) = (a_u(t) \quad a_v(t) \quad a_o(t))^T$ is the Dryden acceleration vector and, as discussed before, it is defined as the first derivative of the Dryden velocity vector $\mathbf{v}_a(t)$, i.e. $\mathbf{a}_a(t) = \dot{\mathbf{v}}_a(t)$. The system in (1.18) can be rewritten in matrix form as follows:

$$\dot{\mathbf{x}}'(t) = \mathbf{F}\mathbf{x}'(t) - \mathbf{P}\mathbf{a}_O(t) - \mathbf{P}\mathbf{a}_a(t), \tag{1.19}$$

where:

- $\mathbf{x}'(t) = (\mathbf{r}'(t)^T \quad \mathbf{v}'(t)^T)^T$ is the state vector composed by both relative position and velocity vector of the target;
- $\mathbf{F} = \begin{pmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{pmatrix}$ is the state update matrix;
- $\mathbf{P}$ is a block matrix defined as $\mathbf{P} = \begin{pmatrix} \mathbf{0}_{3\times3} \\ \mathbf{I}_{3\times3} \end{pmatrix}$.

Now we can introduce in the target state model the so-called process noise vector [12], [13]. Process noise stands for any unforeseen disturbance in the target motion model, e.g. small accelerations that could cause deviations of the target from the straight line trajectory. Usually, such noise is modelled by a zero-mean, white, Gaussian random vector $\mathbf{w}(t) = (w_x(t) \quad w_y(t) \quad w_z(t))^T$ with PSD for each component equal to $N_{0,x}$, $N_{0,y}$, and $N_{0,z}$, respectively. Finally, the complete continuous-time signal model can be expressed in matrix form as:

$$\dot{\mathbf{x}}'(t) = \mathbf{F}\mathbf{x}'(t) - \mathbf{P}\mathbf{a}_O(t) - \mathbf{P}\mathbf{a}_a(t) + \mathbf{P}\mathbf{w}(t). \tag{1.20}$$

1.4.2 Discrete-Time Target State Model

To make the model suitable for digital signal processing, we discretize eq. (1.20) with a "sampling interval", i.e. the radar scan time, equal to T. Following [12], [14], the general solution of the differential system in eq. (1.20) can be expressed as:

$$\mathbf{x}'(t+T) = e^{\mathbf{F}T}\mathbf{x}'(t) + \int_t^{t+T} e^{\mathbf{F}(t+T-\tau)}\mathbf{P}\left(\mathbf{w}(\tau) - \mathbf{a}_O(\tau) - \mathbf{a}_a(\tau)\right)d\tau, \tag{1.21}$$

and then, substituting kT into t, eq. (1.21) becomes

$$\mathbf{x}'[(k+1)T] = e^{\mathbf{F}T}\mathbf{x}'[kT] + \int_{kT}^{(k+1)T} e^{\mathbf{F}((k+1)T-\tau)}\mathbf{P}\left(\mathbf{w}(\tau) - \mathbf{a}_O(\tau) - \mathbf{a}_a(\tau)\right)d\tau. \tag{1.22}$$

It is easy to verify that the two exponential matrices can be evaluated as [14]:

$$\mathbf{T} = e^{\mathbf{F}T} = \begin{pmatrix} \mathbf{I}_{3\times3} & T\mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{pmatrix}, \tag{1.23}$$

$$e^{\mathbf{F}((k+1)T-\tau)} = \begin{pmatrix} \mathbf{I}_{3\times3} & \big((k+1)T-\tau\big)\mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{pmatrix}. \tag{1.24}$$

Finally, the discrete time signal model can be written as:

$$\mathbf{x}'\big[k+1\big] = \mathbf{T}\mathbf{x}'\big[k\big] - \mathbf{g}_d\big[k\big] - \mathbf{g}_a\big[k\big] + \mathbf{g}_w\big[k\big], \tag{1.25}$$

where:

$$\mathbf{g}_d\big[k\big] = \int_{kT}^{(k+1)T} e^{\mathbf{F}\left((k+1)T-\tau\right)}\mathbf{P}\mathbf{a}_O\left(\tau\right)d\tau, \tag{1.26}$$

$$\mathbf{g}_a\big[k\big] = \int_{kT}^{(k+1)T} e^{\mathbf{F}\left((k+1)T-\tau\right)}\mathbf{P}\mathbf{a}_a\left(\tau\right)d\tau, \tag{1.27}$$

$$\mathbf{g}_w\big[k\big] = \int_{kT}^{(k+1)T} e^{\mathbf{F}\left((k+1)T-\tau\right)}\mathbf{P}\mathbf{w}\left(\tau\right)d\tau. \tag{1.28}$$

The term $\mathbf{g}_d[k]$ is a deterministic vector and can be calculated either analytically or through numerical integration. As discussed before, this term is due to the particular racetrack course, in fact it depends on the deterministic acceleration vector $\mathbf{a}_O(t)$, that is the acceleration vector of the platform that carries the radar. Such acceleration vector is a priori known, since it can be directly derived from the nominal aircraft racetrack course. The term $\mathbf{g}_w[k]$ is a discrete random vector derived from the process noise $\mathbf{w}(t)$. Since the functional in (1.28) is linear, $\mathbf{g}_w[k]$ is still Gaussian-distributed with the following mean value and temporal autocorrelation, respectively:

$$E\left\{\mathbf{g}_w\big[k\big]\right\} = \mathbf{0}_{6\times1}, \tag{1.29}$$

$$E\left\{\mathbf{g}_w\big[k\big]\mathbf{g}_w^T\big[m\big]\right\} = \mathbf{Q}_w\delta_{k,m}, \tag{1.30}$$

where $\delta_{k,m}$ is the Kronecker delta function. Through some mathematical manipulation, it can be shown ([12], [13]) that the covariance matrix $\mathbf{Q}_w$ is the following block matrix:

$$\mathbf{Q}_w = \begin{pmatrix} \mathbf{Q}_w^{11} & \mathbf{Q}_w^{12} \\ \mathbf{Q}_w^{21} & \mathbf{Q}_w^{22} \end{pmatrix}, \tag{1.31}$$

$$\mathbf{Q}_w^{11} = \frac{T^3}{3}\tilde{\mathbf{N}}, \quad \mathbf{Q}_w^{12} = \frac{T^2}{2}\tilde{\mathbf{N}}, \quad \mathbf{Q}_w^{22} = T\tilde{\mathbf{N}}, \tag{1.32}$$

where $\tilde{\mathbf{N}} = \mathrm{diag}(N_{0,x}, N_{0,y}, N_{0,z})$.

Now consider the vector $\mathbf{g}_a[k]$, i.e. the term due to the atmospheric turbulences that causes the random deviations of the aircraft from its nominal racetrack course. Also this discrete random vector is Gaussian distributed since both the derivative operator (needed to obtain the continuous-time Dryden acceleration processes in eq. (1.18)) and the functional in eq. (1.27) (needed to obtain $\mathbf{g}_a[k]$) are linear. Therefore, it is fully characterized by its first and second order statistics. By taking the expectation of the functional in eq. (1.27), it is easy to show that the mean value of $\mathbf{g}_a[k]$ is a zero vector, i.e. $E\{\mathbf{g}_a[k]\}=\mathbf{0}$. The autocorrelation matrix can be evaluated as:

$$\begin{aligned}\mathbf{Q}_a[k,m] &= E\left\{\mathbf{g}_a[k]\mathbf{g}_a[m]^T\right\} = \\ &= \int_{kT}^{(k+1)T}\int_{mT}^{(m+1)T} e^{\mathbf{F}((k+1)T-\tau)}\mathbf{P}E\left\{\mathbf{a}_a(\tau)\mathbf{a}_a^T(\upsilon)\right\}\mathbf{P}^T\left(e^{\mathbf{F}((m+1)T-\upsilon)}\right)^T d\tau d\upsilon,\end{aligned} \tag{1.33}$$

where

$$E\left\{\mathbf{a}_a(\tau)\mathbf{a}_a^T(\upsilon)\right\} = \begin{pmatrix} r_{a_u}(\tau-\upsilon) & 0 & 0 \\ 0 & r_{a_v}(\tau-\upsilon) & 0 \\ 0 & 0 & r_{a_o}(\tau-\upsilon) \end{pmatrix}, \tag{1.34}$$

and $r_{a_u}(\cdot)$, $r_{a_v}(\cdot)$, $r_{a_o}(\cdot)$ are the three autocorrelation functions (ACFs) of the continuous-time Dryden acceleration processes given in eqs. (1.14) and (1.15). Unfortunately, the closed-form evaluation of the integral in eq. (1.33) is very hard to obtain due to its analytical complexity[1], but it is easy to show that the vector random process $\mathbf{g}_a[k]$ is not a white process since $\mathbf{Q}_a[k,m]\neq\mathbf{Q}_a[k]\delta_{k,m}$. It is important to note that the non-linear kinematic equation of the air platform does not lead to any non-linear transformation of random processes. Moreover, such "non-linearity" is entirely handled by the deterministic vector $\mathbf{g}_d[k]$ that is added to the discrete-time state vector as shown in eq. (1.25). For clarity, in the rest of the paper, the non-linearity of the kinematic equation of the air platform is indicated as manoeuvrability.

1.5 Filter model

[1] In particular, it can be noted the eq. (4.3.1-8) in [12, p.188] cannot be applied to evaluate the integral in eq. (1.33) since the autocorrelation functions of the Dryden acceleration processes are not of the form $r_a(t-\tau)=V(t)\delta(t-\tau)$ (see (4.2.3-1) of [12, pp. 183]). For the same reason, the algorithm proposed in [16] to evaluate integrals involving the matrix exponential cannot be applied directly here.

The goal of a tracking filter is to estimate the target state $\mathbf{x}'[k]$ from the radar measurements. In the following, when it is possible, we write the discrete time index *k* as a subscript to simplify the notation (i.e. $\mathbf{x}'[k] \triangleq \mathbf{x}'_k$). The radar measurements are modelled as:

$$\mathbf{z}_k = \mathbf{h}\left(\mathbf{x}'_k\right) + \mathbf{n}_k, \tag{1.35}$$

where $\mathbf{h}(\cdot)$ is the Cartesian-to-spherical coordinates transformation and $\mathbf{n}_k$ is the zero-mean, white, Gaussian-distributed measurement noise (independent from the process noise terms) with covariance matrix $\mathbf{R}_n$.

As discussed in the previous Section, the dynamic state model, given in eq. (1.25), is not a Markov sequence since the discrete random vector due to turbulences, i.e. $\mathbf{g}_{a,k}$, is a vector of correlated random variables. To apply the standard Kalman Filter [15], one has to reformulate the problem into one with a state that is a Markov sequence. As discussed in [12, Ch. 8, pp. 320-324], this can be accomplished by the state augmentation procedure. First, the prewhitening system has to be obtained for the correlated process noise term, and then the additional state variables have to be added to the augmented target state in order to obtain a Markov sequence. However, the application of the augmentation procedure to our study case (target state estimation in presence of atmospheric turbulences) falls beyond the scope of this paper. The main feature of this Chapter is, in fact, a quantification of the performance loss suffered if a standard KF is used in a turbulent scenario. For this reason, the tracking filter we consider here is based on the turbulence-ignorant target state model, while the radar measurements are generated taking into accounts also the atmospheric turbulences. This mismatch between the turbulence-ignorant filter model and the simulation model allows us to evaluate the losses in terms of tracking performance due to the fact that the turbulences are not considered into the filter model. Finally, to summarize the previous considerations on the filter model, we assume:

- A turbulent-ignorant target state equation derived from eq. (1.25) by cutting off the term $\mathbf{g}_{a,k}$:

$$\mathbf{x}'_{k+1} = \mathbf{T}\mathbf{x}'_k - \mathbf{g}_{d,k} + \mathbf{g}_{w,k}, \tag{1.36}$$

- A measurement equation given by eq. (1.35).

1.5.1 Performance bound for the ideal case

In this Section we provide a performance bound for the ideal case, i.e. for a scenario without turbulences. By comparing the error standard deviation of the turbulence-ignorant tracking filter in a turbulent scenario with the performance bound evaluated in the ideal case, it is possible to quantify the losses in tracking accuracy due to the atmospheric turbulences. Such bound is the well-known Posterior Cramér-Rao Lower Bound [17], [18, Ch. 4]. Consider the filtering problem defined by eqs. (1.35) and (1.36), repeated here for convenience:

$$\mathbf{x}'_{k+1} = \mathbf{T}\mathbf{x}'_k - \mathbf{g}_{d,k} + \mathbf{g}_{w,k},$$

$$\mathbf{z}_k = \mathbf{h}\left(\mathbf{x}'_k\right) + \mathbf{n}_k.$$

Let $\hat{\mathbf{x}}'_{k|k}$ be an unbiased estimator of the state vector $\mathbf{x}'_k$, based on the measurement sequence $Z_k = \{\mathbf{z}_1, \ldots, \mathbf{z}_k\}$. The error covariance matrix of $\hat{\mathbf{x}}'_{k|k}$, denoted as $\mathbf{P}_{k|k}$, has a lower bound (referred to as the PCRLB) expressed as follows [18, Ch. 4]:

$$\mathbf{P}_{k|k} \triangleq E\left\{\left(\hat{\mathbf{x}}'_{k|k} - \mathbf{x}'_k\right)\left(\hat{\mathbf{x}}'_{k|k} - \mathbf{x}'_k\right)^T\right\} \geq \mathbf{I}_k^{-1}, \tag{1.37}$$

where $\mathbf{I}_k$ is the Fisher Information Matrix (FIM). There exists a wide literature on the PCRLB (see, for example, the references in [18, Ch. 4]), and then here we report only the principal equations that allow us to evaluate the PCRLB for the filtering problem given in eqs. (1.35) and (1.36). As shown in [17], the FIM $\mathbf{I}_k$ can be recursively computed as:

$$\mathbf{I}_{k+1} = \mathbf{D}_k^{22} - \mathbf{D}_k^{21}\left(\mathbf{I}_k + \mathbf{D}_k^{11}\right)^{-1}\mathbf{D}_k^{12}, \quad k > 0 \tag{1.38}$$

where, for the filtering problem in eqs. (1.35) and (1.36), the matrices $\mathbf{D}_k^{11}$, $\mathbf{D}_k^{12}$, $\mathbf{D}_k^{21}$ and $\mathbf{D}_k^{22}$ can be evaluated as:

$$\mathbf{D}_k^{11} = \mathbf{T}^T\mathbf{Q}_w^{-1}\mathbf{T}, \tag{1.39}$$

$$\mathbf{D}_k^{12} = \left(\mathbf{D}_k^{21}\right)^T = -\mathbf{T}^T\mathbf{Q}_w^{-1}, \tag{1.40}$$

$$\mathbf{D}_k^{22} = \mathbf{Q}_w^{-1} + E\left\{\mathbf{H}_{k+1}^T \mathbf{R}_n^{-1} \mathbf{H}_{k+1}\right\}, \tag{1.41}$$

where $\mathbf{H}_{k+1}$ is the Jacobian of $\mathbf{h}(\cdot)$ evaluated at the true value of $\mathbf{x}'_{k+1}$, i.e. $\mathbf{H}_{k+1} = \mathbf{J}(\mathbf{h}(\mathbf{x}'_{k+1}))$. The recursive equation (1.38) can be initialized using the same initial error covariance matrix needed to initialize the Kalman filter:

$$\mathbf{I}_0 = E\left\{\left(\hat{\mathbf{x}}_0 - \mathbf{x}_0\right)\left(\hat{\mathbf{x}}_0 - \mathbf{x}_0\right)^T\right\} = \mathbf{P}_0 . \tag{1.42}$$

1.6 Simulation model and data generation

As discussed before, in order to assess the losses in tracking accuracy, the effect of the atmospheric turbulences has to be taken into account in the simulation data model. More precisely, we have to generate the random vector $\mathbf{g}_a[k] \triangleq \mathbf{g}_{a,k}$ defined by (1.27) [19]. The generation of this term can be accomplished in two steps. First, it must be noted that the integral in (1.27) could be evaluated numerically as function of Dryden velocity processes as:

$$\begin{aligned}\mathbf{g}_a[k] &= \int_{kT}^{(k+1)T} e^{\mathbf{F}((k+1)T-\tau)} \mathbf{P}\mathbf{a}_a(\tau)\, d\tau = \int_{kT}^{(k+1)T} \begin{pmatrix} \left[(k+1)T - \tau\right]\mathbf{a}_a(\tau) \\ \mathbf{a}_a(\tau) \end{pmatrix} d\tau \\ &= \begin{pmatrix} (k+1)T\,\mathbf{v}_a(\tau)\Big|_{kt}^{(k+1)T} - \int_{kT}^{(k+1)T} \tau \mathbf{a}_a(\tau)\, d\tau \\ \mathbf{v}_a(\tau)\Big|_{kt}^{(k+1)T} \end{pmatrix} \simeq \begin{pmatrix} T\left(\mathbf{v}_a[k+1] - \mathbf{v}_a[k]\right)/2 \\ \mathbf{v}_a[k+1] - \mathbf{v}_a[k] \end{pmatrix},\end{aligned} \tag{1.43}$$

where the term $\int_{kT}^{(k+1)T} \tau \mathbf{a}_a(\tau)\, d\tau$ is evaluated, using the integration by parts rule as:

$$\begin{aligned}\int_{kT}^{(k+1)T} \tau \mathbf{a}_a(\tau)\, d\tau &= \tau\, \mathbf{v}_a(\tau)\Big|_{kt}^{(k+1)T} - \int_{kT}^{(k+1)T} \mathbf{v}_a(\tau)\, d\tau \\ &\simeq (k+1)T\mathbf{v}_a[k+1] - kT\mathbf{v}_a[k] - T\left(\mathbf{v}_a[k+1] + \mathbf{v}_a[k]\right)/2,\end{aligned} \tag{1.44}$$

where, in the last equality, we have used the trapezoidal integration rule. At this point, we have to generate the velocity vector $\mathbf{v}_a[k]$. To do this, we follow the procedure proposed in [5]. The basic idea behind this method is that the turbulence velocities can be readily generated by passing a white Gaussian signals through appropriate linear filters with

Laplacian transfer functions obtained from the Dryden PSDs given in eqs. (1.5)-(1.7). The derivation of the relative discrete-time filters is addressed in [5].

1.7 Simulation results

In order to deal with the non-linear coordinate transformation in the measurements equation (1.35), we apply the *Converted Measurements Kalman Filter* (CMKF) ([20], [21]). Our simulations have been performed using a turbulence-ignorant filter model with a target dynamic model given in eq. (1.36) and two different simulation models: (*i*) tracking in the presence of atmospheric turbulences, and (*ii*) tracking in an ideal environment of absence of turbulence. In the first case, the turbulence has been generated following the procedure discussed in Section 1.6.

Two different study cases have been investigated in the simulations: (*i*) the medium range target scenario, where the initial radar-target distance is of about 90 km and (*ii*) the short range target scenario where the initial radar-target distance is of about 12 km. The radar parameters used in the simulations are: σ_ρ = 2.4 m, $\sigma_\theta = \sigma_\varepsilon$ = 0.25°, T = 1 s, the flight altitude h is of about 1000 m and the platform speed is v_p = 90 m/s. The detection and false alarm probabilities are assumed to be 1 and 0, respectively. The sea surface target is modelled as a high speed dinghy [1] with a velocity vector defined as $\mathbf{v}_t$ = (7.63 6.76 0) m/s and a power spectral density for each component of the continuous-time acceleration vector $\mathbf{w}(t)$ equal to $N_{0,x} = N_{0,y} = N_{0,z}$ = 0.01 m^2/s^3. The geometrical parameters of the aircraft racetrack course are: d = 60 km and R = 10 km. The time intervals needed to cover the straight and circular stretch of the racetrack are T_l = 667 s and T_c = 350 s , respectively; the racetrack course period is of about 2000 s. The first measurement was collected when the aircraft begins to cover the first straight segment (at the point "A" in fig. 1.1). To define the Dryden PSDs, we used a set of parameters that characterize a standard turbulence level for a flight altitude of 1000 m. According to [5] and [7], we have: V_0 = 96 m/s and $L_u = L_v = L_o$ = 533.54 m. We also simulated two levels of turbulence power [6], [8]: the maximum value for a "standard" turbulence that correspond to $\sigma_u = \sigma_v = \sigma_o$ = 2.1 m/s and a value that characterizes the storm turbulences that correspond to $\sigma_u = \sigma_v = \sigma_o$ = 6.4 m/s.

For both the medium range and the short range target scenarios, a comparison between position and velocity estimates, in the ideal case (absence of turbulences) and in the presence of turbulences is performed in terms of mean value and standard deviation (s. d.) of the estimation error for each component of the state vector. By defining the estimation error for the *i*th component of the state vector as $e_{i,k} = \hat{x}_{i,k} - x_{i,k}$, the error mean value and the error s. d. are $\mu_{e_{i,k}} = E\{e_{i,k}\}$ and $\sigma_{e_{i,k}} = E\{(e_{i,k} - \mu_{e_{i,k}})^2\}^{1/2}$, respectively. Moreover, the error standard deviation (std) is compared with the PCRLB for the ideal case.

The curves of the error mean value relative to the simulations with atmospheric turbulences present a larger variability around the zero level with respect to those relative to the ideal case. This is an expected behaviour that does not carry much information on the loss of tracking accuracy. The most useful index to quantify such losses is the error standard deviation, and for this reason, the curves of the error mean value are not reported here. All performance curves have been obtained by averaging over 500 independent Monte Carlo runs. For sake of brevity, we show only the results relative to the *x* and *y* components of the position and velocity vectors. In Figs. 1.3-1.10, we plot four different curves: 1) the ideal case, 2) the case with a standard level of turbulences ($\sigma_u = \sigma_v = \sigma_o$ = 2.1 m/s), and 3) the case with a stormy level of turbulences ($\sigma_u = \sigma_v = \sigma_o$ = 6.4 m/s) and 4) the PCRLB for the ideal case. In figs. 1.3-1.6, the simulation results relative to the medium range target scenario are shown. As we can see from figs. 1.3 and 1.4, the loss in accuracy in the position estimate is almost negligible (of the order of meters) for both the standard turbulence and storm turbulence. This result holds for all the three components, *x, y,* and *z*. Figs. 1.5 and 1.6 show the error standard deviation for the estimate of the components of the velocity vector. It can be observed that the loss in accuracy with respect to the ideal case is of about 0.4 m/s for the case of standard turbulence, while for the case of storm turbulence is of about 2 m/s for all the three components.

Finally, some consideration on the effects of the manoeuvrability of the aircraft on the tracking performance has to be done. The quasi-ellipsoidal racetrack course causes the periodic-like behaviour of the error std that presents a dip (for the estimate of the *x* component) or a peak (for the estimate of the *y* and *z* components) around the time interval in which the aircraft covers the circular segment of its racetrack course. This means that the air platform manoeuvrability improves the estimate of the *x* component of position and

velocity vector. On the other hand, for the estimate of the *y* and *z* components, a better estimate is reached when the aircraft moves along the straight segment of the racetrack course.

In Figs. 1.7-1.10, the simulation results relative to the short range target scenario are shown. Decreasing the distance between radar and target, the tracking errors decrease too, but the effects of the manoeuvrability of the air platform becomes more evident. Also the PCRLB presents a periodic-like behaviour in the short range scenario. The impact of the turbulence in the position estimate is negligible for the standard turbulence while it is slightly higher for the storm turbulences. The loss in accuracy for the velocity estimate is quite similar to the ones relative to the middle range target scenarios, i.e. of the order of 0.5 m/s for the standard turbulences and 2 m/s for the storm turbulence.

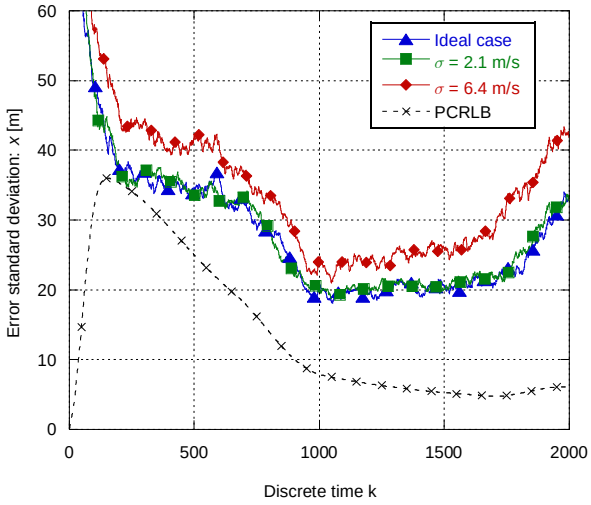

Figure 1.3 – Error std and PCRLB square root for the *x* component of position vector for the medium range target scenario.

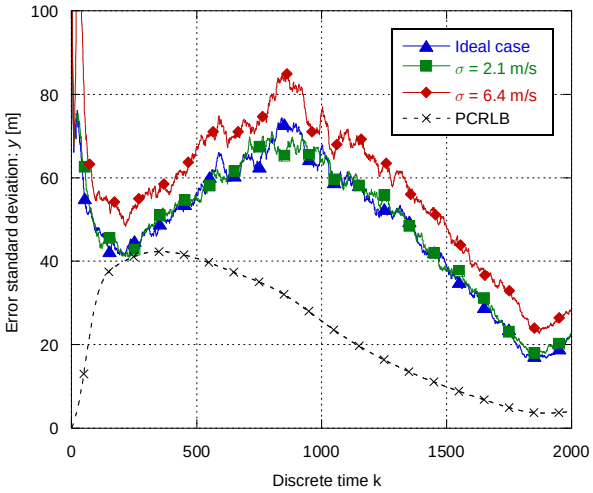

Figure 1.4 – Error std and PCRLB square root for the *y* component of position vector for the medium range target scenario.

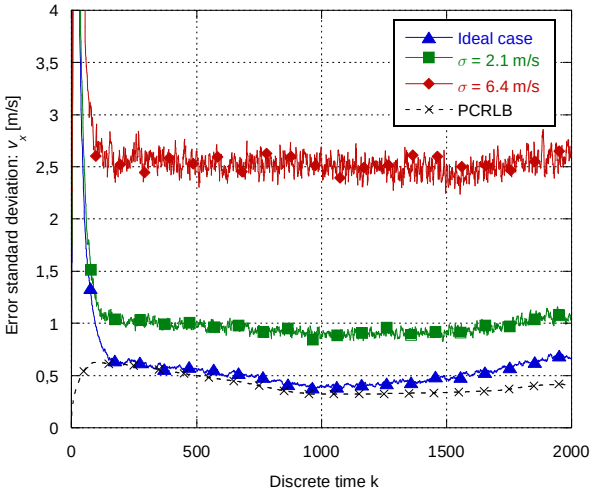

Figure 1.5 – Error std and PCRLB square root for the *x* component of velocity vector for the medium range target scenario.

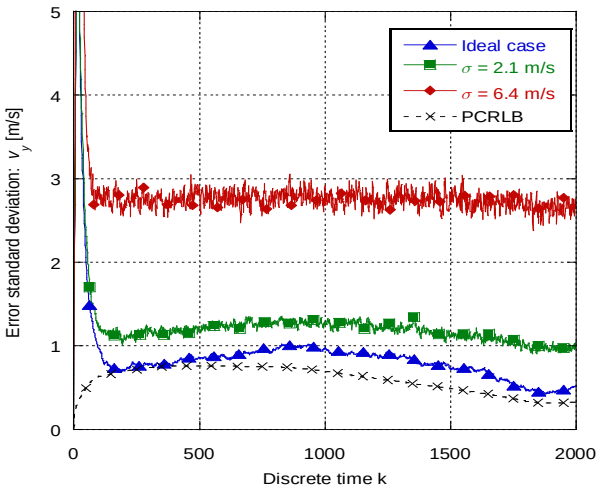

Figure 1.6 – Error std and PCRLB square root for the *y* component of velocity vector for the medium range target scenario.

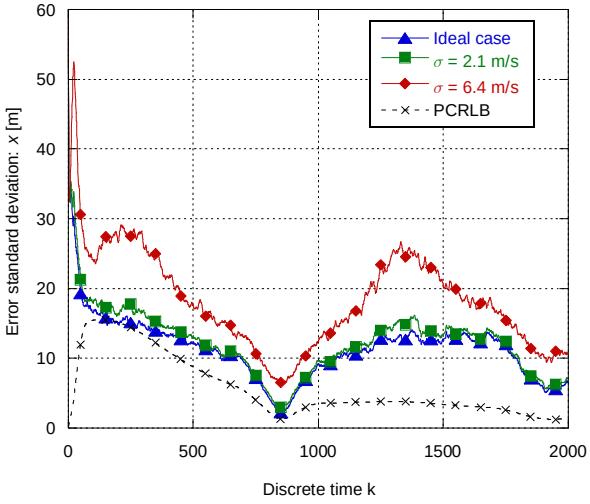

Figure 1.7 – Error std and PCRLB square root for the *x* component of position vector for the short range target scenario.

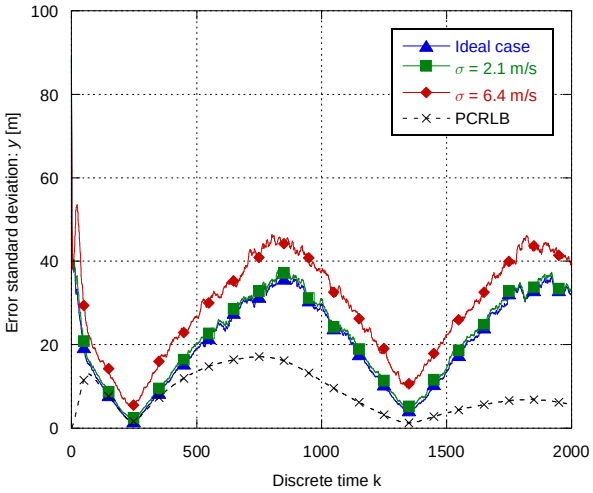

Figure 1.8 – Error std and PCRLB square root for the *y* component of position vector for the short range target scenario.

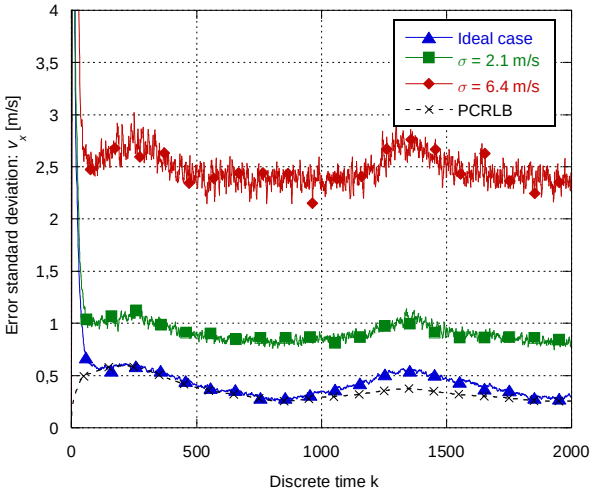

Figure 1.9 – Error std and PCRLB square root for the *x* component of velocity vector for the short range target scenario.

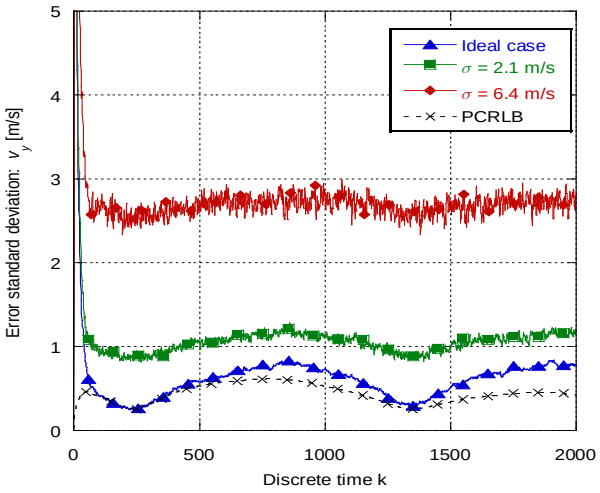

Figure 1.10 – Error std and PCRLB square root for the *y* component of velocity vector for the short range target scenario.

1.8 Summary

In this Chapter, we investigated the impact of the atmospheric turbulences on a turbulence-ignorant tracking algorithm for airborne radar. The flight disturbances have been modelled according to the Dryden model. The discrete-time target state model has been obtained in presence of such turbulence and the relative noise term has been discussed and statistically characterized. Due to the high computational complexity, the turbulence model has not been employed in the filter model, but a simplified, turbulence-ignorant filter has been assumed. The performance of this filter has been assessed in a simulated turbulent scenario and also compared with the PCRLB evaluated for the ideal case of absence of turbulence in the data model. Moreover, a typical racetrack course for the surveillance mission has been chosen for the air platform that carries the radar. The simulations show that the atmospheric turbulences cause a degradation of the tracking performance for the estimate of the target velocity vector, while the losses in the estimate of the target position vector are almost negligible. Moreover, the simulation results show that the air platform manoeuvrability improves the estimate of the *x* component of position and velocity vector,

while, for the estimate of the *y* and *z* components, a better estimate is reached when the aircraft moves along the straight segment of the racetrack course.

References

[1] S. Giompapa, A. Farina, F. Gini, A. Graziano, R. Di Stefano, "Computer Simulation of an Integrated Multi-Sensor System for Maritime Border Control," *IEEE Radar Conference, 2007*, pp.308-313, 17-20 April 2007.

[2] S. Giompapa, F. Gini, A. Farina, A. Graziano, R. Di Stefano, "Maritime border control computer simulation," *IEEE Aerospace and Electronic Systems Magazine*, vol. 23, no. 11, pp. 39-45, November 2008.

[3] S. Giompapa, F. Gini, A. Farina, A. Graziano, R. Croci, R. Di Stefano, "Maritime border control multisensor system," *IEEE Aerospace and Electronic Systems Magazine*, vol. 24, no. 8, pp. 9-15, August 2009.

[4] S. Giompapa, A. Farina, F. Gini, A. Graziano, R. Croci, and R. Di Stefano, "Naval Target Classification by Fusion of Multiple Imaging Sensors Based on the Confusion Matrix," *International Journal of Navigation and Observation*, 15 pages, vol. 2009.

[5] T. R. Beal, "Digital Simulation of Atmospheric Turbulence for Dryden and von Karman Models," *Journal of Guidance, Control, and dynamics*, vol. 16, no. 1, pp. 132-138, Jan.-Feb. 1993.

[6] B. Etkin, "The Turbulent Wind and Its Effect on Flight," *AIAA Wright Brothers Lecture*, 1980.

[7] E. F. Hogge, "B-737 Linear Autoland Simulink Model", Technical Report NASA/CR-2004-213021, May 2004.

Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.65.541.

[8] "Flying Qualities of Piloted Airplanes," MIL-F-8785C, Nov. 5, 1980.

[9] S. Fortunati, M. Greco, F. Gini, A. Farina, A. Graziano, “Impact of Flight Disturbances on Tracking Airborne Radar,” *IRS 2009*, Hamburg, Germany, 09-11 September 2009, pp. 407-411.

[10] W. C. Morchin, *Airborne Early Warning Radar*, Artech House Publishers, Boston, 1990.

[11] A. H. Zemanian, *Distribution Theory and Transform analysis: an Introduction to Generalized Function with Applications*, New York: Dover, 1987.

[12] Y. Bar-Shalom, X. R. Li, T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms, and Software*, New York: Wiley, 2001.

[13] X. R. Li, V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *Aerospace and Electronic Systems, IEEE Transactions on* , vol.39, no.4, pp. 1333- 1364, Oct. 2003.

[14] R. A. Singer, “Estimating Optimal Tracking Filter Performance for Manned Manoeuvring Targets,” *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 6, no.4, pp.473-483, July 1970.

[15] R. E. Kalman, “A new approach to linear filtering and prediction problems”, *Transactions of the ASME – Journal of Basic Engineering*, no. 82 (Series D), pp. 35-45, 1960.

[16] C. F. Van Loan, “Computing integrals involving the matrix exponential,” *Automatic Control, IEEE Transactions on*, vol. 23, no.3, pp.395-404, June 1978.

[17] P. Tichavsky, C. H. Muravchik, A. Nehorai, “Posterior Cramer-Rao bounds for discrete-time nonlinear filtering,” *Signal Processing, IEEE Transactions on* , vol.46, no.5, pp.1386-1396, May 1998.

[18] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House Publishers, London, 2004.

[19] C. W. Therrien, *Discrete Random Signals and Statistical Signal Processing*, Englewood Cliffs, NJ: Prentice-Hall, 1992.

[20] D. Lerro, and Y. Bar-Shalom, “Tracking With Debiased Consistent Converted Measurements Versus EKF”, *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 29, no. 3, pp 1015-1022, July 1993.

[21] M. Longbin, S. Xiaoquan, Z. Yiyu, S. Z. Kang, and Y. Bar-Shalom, “Unbiased converted measurements for tracking”, *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 34, no. 3, pp. 1023-1027, July 1998.

Chapter 2: Correction of the troposheric refraction effects

2.1 Introduction

In this Chapter, we focus our attention on the atmospheric propagation effect on the radar EM signals and their correction for airborne radar tracking. Such atmospheric effects are often neglected when the tracking performance of the airborne radar is evaluated. However, for modern long-range and high-resolution radar systems such errors start to be relevant and they should not be neglected [1]. The atmospheric layer that mostly influences the EM propagation is the first layer, called troposphere, extending from the Earth's surface to about 8 km. The troposphere is characterized by a refractive index that varies with the altitude and with the local weather condition, i.e. local pressure, temperature and humidity. This variability of the refractive index causes an error in the radar measurements (range, azimuth, and elevation). In an ideal environment, i.e. without refractive effects, the radar measurement of the target range is determined as the speed of the light in the vacuum multiplied by the time needed by the EM signal to cover the one-way distance between radar and target, while the radar measurement of the target azimuth and elevation are determined by the direction of arrival (DOA) of the scattered EM signal. In a material medium with a varying refractive index, as the troposphere is, two effects cause an error in the range and elevation measurements [2, Ch. 3]. First, the radar EM signal travels at a speed lower than the speed of light in the vacuum and this induces an overestimate of the target range. Second, due to the variation of the refractive index as function of the height, a ray bending (associated with the refractive index gradient) occurs with a result that the direction of arrival of the scattered EM signal is different from the straight ray path producing an error (or bias) in the elevation measurement. Moreover, the geometrical

distortion of the ray path causes also an error (or bias) in the measurement of the target range. The magnitude of these errors depend on both the gradient and the absolute value of the refractive index along the signal path [2, Ch. 3], [3]. There is also an azimuth measurement error due to the fluctuations of the refractive index value [4, App. D]. The azimuth error is smaller than the elevation error and, under the simplified hypotheses made in this paper on the refractive index model (see Section III), it can be neglected. For these reasons, in the rest of the paper, we consider only the range and the elevation bias errors, while the azimuth bias is assumed negligible.

In addition to the above mentioned effects, an EM signal passing through the troposphere can experience a wide range of anomalous propagation effects [2, Ch. 3], [3] [4, App. D]. The most important are the "ducting" effect and the small-scale fluctuations of the refractive index. Tropospheric ducting occurs when the refractive index decreases with the height at a rate more negative than the standard rate. More precisely, the standard decrease of refractive index with height is at a rate of about -39 *N* units per kilometer (*N*/km)[2]. Two different anomalous situations can occur: the so called subrefraction and superrefraction. Subrefraction implies that the gradient of the refractive index is less negative than the standard value of -39 *N*/km while superrefraction occurs when the gradient is more negative than the standard value of -39 N/km, but less negative than -157 *N*/km. Tropospheric ducting, instead, occurs when the refractive index decreases with the height at a rate that is more negative than -157 *N*/km [2, Ch. 3], [3]. These effects are generally caused by a temperature inversion, and frequently occur in coastal regions. The main consequence of the phenomenon of ducting for an air-to-air or surface-to-air ray path is the creation of a gap or *hole* in the radar coverage. On the other hand, the radar located within a duct will realize increased detection range for target that is also located in the duct. The small-scale fluctuations of the refractive index are mostly caused by atmospheric turbulence, clouds and small scale meteorological phenomena [4, App. D], [5, Part II], [6], [7]. Effects of this type are highly unpredictable and must be regarded as random processes and analysed by statistical methods.

In this Chapter, we propose a tracking algorithm that is able to mitigate the tropospheric range and elevation errors from the estimated position and velocity vectors [8].

[2] The unit of measurement *N* is referred to a scaled version of the refractive index (sometimes called radio refractive index or scaled EM refractivity index) that will be explicitly defined in Section IV.B.

First, a mathematical model to describe and calculate the EM radar signal ray path, under simplified assumptions on the refractive index, is discussed. Using this mathematical model, the range and elevation errors are evaluated and the radar corrupted measurements are then numerically generated. Second, an estimate of the tropospheric errors is obtained from the (simulated) corrupted radar measurements. To obtain these estimates, we use two algorithms, one for the elevation error estimate [1], [9] and one for the range error estimate [10]. The theoretical values of the errors and the estimated values are compared and the effects of certain levels of inaccuracy on the evaluation of the model parameters are analysed. Finally, we propose a tracking algorithm based on the Kalman filter (KF) that, using the estimated values of the range and elevation errors, is able to mitigate the tropospheric errors during the tracking procedure. The performance of the proposed algorithm is evaluated for a medium range scenario, where the radar-target distance varies in a range of 70-100 km.

The rest of the Chapter is organized as follows. In Section 2.2, the geometry for the scenario is described. The choice of this scenario, as discussed previously in Section 1.2, has been performed to provide an example of a typical border surveillance mission in which airborne radars are widely used and it does not represent a limitation to the applicability of the proposed algorithm. In Section 2.3, a mathematical model to describe the EM tropospheric propagation is derived and used to generate the synthetic corrupted radar measurements. Two algorithms for the estimate of the tropospheric range and elevation errors are discussed in Section 2.4. In Section 2.5, the proposed KF-based algorithm is described. Simulation results concerning the performance of the proposed algorithm are presented in Section 2.6. Our conclusions are collected in Section 2.7.

2.2 GEOMETRY OF THE SCENARIO

The considered scenario is exactly the same described in Section 1.2; we recall it here for clarity. An inertial Cartesian reference system (x, y, z) is used to describe the motion of both aircraft and target. Since the airborne radar acquires the radar measurements in its local reference system, to describe the EM ray path in a suitable way to derive the a mathematical model for the tropospheric propagation, we define a relative, aircraft-centred

Cartesian coordinate system, named (x', y', z'). The inertial and the relative aircraft-centred coordinate systems are parallel during the aircraft flight.

In our study we refer to a 3D Airborne Early Warning Radar (AEWR) with a range, azimuth and elevation accuracy given by σ_ρ, σ_θ and σ_ε respectively, a scan time of T seconds and a flight height h. The numerical values of all these parameters are given in Section 2.6. Without loss of generality, the chosen aircraft race is a quasi-ellipsoidal racetrack [11]. This racetrack is adopted in a border security and surveillance mission where the airborne radar must be able to cover a wide area. Its kinematical equations are non-linear and must be accounted for in the design of the tracking algorithm performance. In the following, with reference to Fig. 1.1, the main geometrical parameters of the scenario are:

- $\mathbf{O}_a = \left(O_{a,x}, O_{a,y}, O_{a,z}\right)$: centre of the aircraft race, then $O_{a,z} = h$;
- d: length of the straight stretch of the racetrack;
- R: radius of the circular stretch of the racetrack;
- $\mathbf{r}(t)$: target position vector defined in the inertial reference system;
- $\mathbf{r}'(t)$: target position vector defined in the relative aircraft-centred reference system;
- $\mathbf{r}_O(t)$: aircraft position vector defined in the inertial reference system.

For the ease of mathematical formulation of the problem, the target motion is supposed rectilinear and uniform.

2.3 MATHEMATICAL MODEL FOR TROPOSPHERIC PROPAGATION

As already pointed out in the Introduction, the variation of the refractive index, as function of the altitude, gives rise to a geometrical distortion of the EM radar signal that modifies range and elevation measurements (see Fig. 2.1). Moreover, the knowledge of the refractive index is needed to quantify the slowing of the EM signal that is another cause of

the range error. To derive a model for the EM propagation, we need a mathematical model that characterizes the refractive index $n(x,y,z)$ for all the points of the considered tropospheric region. As discussed before, the wide range of tropospheric phenomena make the formulation of a rigorous and exhaustive mathematical model for the refractive index very difficult to obtain. Here, we adopt two simplified assumptions: (i) the tropospheric temperature and humidity vary only with the altitude; (ii) a spherical model, with radius r_0 for the Earth's geoid is assumed. Assumption (i) implies that, for a given height $\bar{h}$, the tropospheric parameters (temperature, pressure, humidity) remain constant in all the considered region. This is not a realistic assumption because of the presence of the spatial inhomogeneities and of the turbulent processes in the troposphere [4, App. D], [5, Part II]. On the other hand, without assuming these simplified hypotheses, the derivation in close form of the refracted ray path becomes extremely hard.

As shown in Fig. 2.1, the corrupted radar measurement vector $\mathbf{z}'_k$ can be defined in spherical coordinates as:

$$\mathbf{z}'_k = \mathbf{z}_k + \boldsymbol{\mu}_k, \tag{2.1}$$

where k is the discrete time index and $\boldsymbol{\mu}_k = \begin{pmatrix} \Delta\rho_k & 0 & \Delta\varepsilon_k \end{pmatrix}^T$ is the vector of the range and elevation tropospheric errors, $\Delta\rho_k$ and $\Delta\varepsilon_k$ respectively. The vector $\mathbf{z}_k$ represents the noisy radar measurements, defined as:

$$\mathbf{z}_k = \begin{pmatrix} \rho_k & \theta_k & \varepsilon_k \end{pmatrix}^T + \mathbf{n}_k, \tag{2.2}$$

where ρ_k, θ_k and ε_k represent the true range, azimuth and elevation of the target defined in the aircraft reference system and $\mathbf{n}_k$ is the measurement noise vector, modelled as a discrete, zero-mean, Gaussian distributed random vector with diagonal covariance matrix given by $\mathbf{C}_n = \mathrm{diag}\left(\sigma_\rho^2, \sigma_\theta^2, \sigma_\varepsilon^2\right)$. In the rest of the paper, we define as *corrupted* the position vector affected by the only tropospheric errors (not considering the measurement noise) and as *noisy* measurement the radar measures affected by both tropospheric errors and measurement noise, as shown in eq. (2.1).

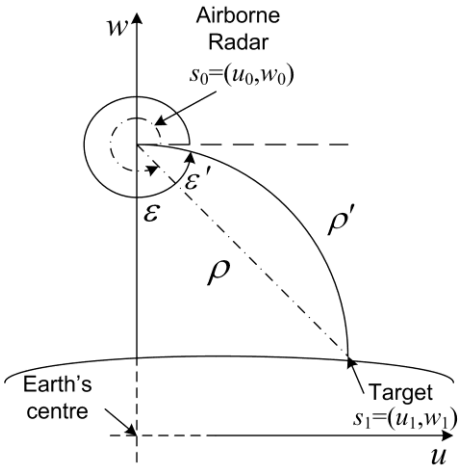

Figure 2.1 - Curvature of EM radar signal.

2.3.1 Generation of the refracted ray path

In geometrical optics, the well known Fermat's principle states that the path taken between two points in a three dimensional space by an EM signal is the path that can be traversed in the least time. The EM signal speed in the troposphere is given by:

$$v(s) = \frac{c}{n(s)}, \tag{2.3}$$

where $n(s)$ is the tropospheric refractive index and c is the speed of light in the vacuum. We can define a two dimensional vector space S such that the aircraft, the target and the Earth's centre belong to S. Such vector space is unique and it is well defined for all radar-target configurations. We can define an orthogonal coordinate system (u,v) on S centred in the Earth's centre (Fig. 2.1). Since both ideal path and refracted path belong to S, the following equations are defined on S, and then the final result will be converted in the usual spherical coordinate system.

The time necessary to the EM signal to cover the path between s_0 and s_1 is given by the following path integral:

$$T = \int_{s_0}^{s_1} \frac{ds}{v(s)} = \frac{1}{c}\int_{s_0}^{s_1} n(s)\,ds = \frac{1}{c}\int_{u_0}^{u_1} \sqrt{1+\left(\dot{w}(u)\right)^2}\, n(u,w)\,du \text{ [3]}. \tag{2.4}$$

For ease of notation, we define $F(u,w,\dot{w}) \triangleq \sqrt{1+\left(\dot{w}(u)\right)^2}\, n(u,w)$. Our aim is to calculate the path $s(u) = \left(u, w(u)\right)$ that is the solution of the following functional minimum problem:

$$\begin{aligned} &\min_{w\in C^1([u_0,u_1])} J(w), \\ &J(w) = \frac{1}{c}\int_{u_0}^{u_1} F\left(u,w,\dot{w}\right)du. \end{aligned} \tag{2.5}$$

From a fundamental theorem of the calculus of variation [12], the function $w(u)$ that minimizes the problem in eq. (2.5) is the solution of the following differential equation, called Euler-Lagrange formula[4]:

$$\frac{d}{du}\left(\frac{\partial}{\partial \dot{w}} F\left(u,w,\dot{w}\right)\right) = \frac{\partial}{\partial w} F\left(u,w,\dot{w}\right). \tag{2.6}$$

To solve the equation in (2.6) we need to specify a refractive index profile defined on S, i.e. $n(u,v)$. The most widely used model is the exponential model [2, Ch. 3], [3] that can be expressed as:

$$n(u,w) = 1+(n_0-1)e^{-b(\sqrt{u^2+w^2}-r_0)} = 1+10^{-6}N_0 e^{-b(\sqrt{u^2+w^2}-r_0)}, \tag{2.7}$$

where n_0 is the sea-level refractive index, $N_0 \triangleq 10^6(n_0-1)$ is a constant introduced only to deal with the small value of the refractive index and it will be better defined in Section 2.4.2, r_0 is the Earth's radius and b is a physical constant parameter named scale height. Both n_0 and b depend on the physical local tropospheric characteristics, e. g. temperature,

---

[3] The change of variable in the path integral in (2.4) can be easily explained as follows: $ds = \sqrt{du^2+dw^2} = \sqrt{1+(dw/du)^2}\,du = \sqrt{1+\left(\dot{w}(u)\right)^2}\,du.$

[4] The Euler-Lagrange differential equation is a fundamental equation of the calculus of variations [15]. It states that if J is a functional, defined by an integral of the form $J(x) = \int f\left(t,x(t),\dot{x}(t)\right)dt$, then $\bar{x}$ is a stationary value for J if it is solution of the following (Euler-Lagrange) differential equation: $\frac{d}{dt}\left(\frac{\partial}{\partial \dot{x}} f\left(t,x(t),\dot{x}(t)\right)\right) = \frac{\partial}{\partial x} f\left(t,x(t),\dot{x}(t)\right).$

pressure and humidity, and vary with the seasons and with the particular geographical area, then they must be chosen accordingly with on-site measurements. Moreover, the value of the sea-level refractive index over many Earth locations and for every season could be obtained by using the radiosonde data (available to general public) collected in the radiosonde archive IGRA (Integrated Global Radiosonde Archive) [13].

The solution of (2.6), with the exponential model of (2.7) yields the following Dirichlet's problem for differential equations (for the proof, see Appendix A):

$$\begin{cases} \ddot{w} = \dfrac{(1+\dot{w}^2)}{n(u,w)} \cdot \dfrac{(w-\dot{w}u)}{\sqrt{u^2+w^2}} \cdot (-b)\left(n(u,w)-1\right) \\ w_0(u_0) = w_0 \\ w_1(u_1) = w_1 \end{cases} . \tag{2.8}$$

In Fig. 2.2, a comparison between the ideal (straight) ray path and the refracted ray path obtained by solving numerically the Dirichlet's problem in eq. (2.8) is shown for a radar-target distance of 90 km.

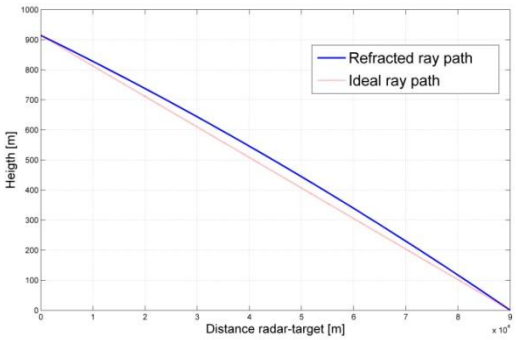

Figure 2.2 - Comparison between the refracted ray path and the ideal ray path for a radar-target distance of 90 km.

2.3.2 Generation of range and elevation errors

To generate the troposheric range error, we have to solve the path integral in eq. (2.4). From the solution of the Dirichlet's problem in eq. (2.8), we know the EM ray path coordinate *w* and its first derivative $\dot{w}$, then the integral in eq. (2.4) can be evaluated via

numerical integration. From Fig. 2.1, we have that the corrupted range and the true range can be expressed as:

$$\rho' = \int_{x_0}^{x_1} \sqrt{1+\left(\dot{w}(u)\right)^2}\, n(u,w)\, du\ , \tag{2.9}$$

$$\rho = \sqrt{\left(u_1 - u_0\right)^2 + \left(w_1 - w_0\right)^2}. \tag{2.10}$$

Then, the range error is:

$$\Delta\rho = \rho' - \rho. \tag{2.11}$$

Now, we have to generate the tropospheric elevation error. The corrupted elevation angle (see Fig. 2) is given by:

$$\varepsilon' = 2\pi + \operatorname{atan}\left(\dot{w}(u_0)\right), \tag{2.12}$$

while the true elevation angle can be obtained as:

$$\varepsilon = 2\pi + \operatorname{atan}\left(\frac{w_1 - w_0}{u_1 - u_0}\right). \tag{2.13}$$

Then, the tropospheric elevation error is given by:

$$\Delta\varepsilon = \varepsilon' - \varepsilon\ . \tag{2.14}$$

2.4 EVALUATION OF THE TROPOSPHERIC ERRORS

2.4.1 Evaluation of tropospheric elevation error

The correction of the elevation error is a very important task in airborne tracking radar. In order to evaluate such error, we use the procedure shown in [5] and [13], where an estimation of $\Delta\varepsilon_k$ is obtained by assuming the exponential tropospheric model shown in eq. (2.7). The ray curvature is a function of the derivative of refractive index $n(h)$, where h is the height from which the EM ray path starts (in our case, h is the radar platform height), and it can be expressed as [1], [9]:

$$\frac{1}{r_C} \triangleq -\frac{\cos(\varepsilon')}{n(h)}\frac{dn(h)}{dh}, \tag{2.15}$$

where r_C is the radius of curvature and $1/r_C$ is the curvature. By evaluating the derivative term, eq. (2.15) becomes:

$$\frac{1}{r_C} = \frac{N_0 b e^{-bh}\cdot 10^{-6}}{1+N_0 e^{-bh}\cdot 10^{-6}}\cos(\varepsilon'_k), \tag{2.16}$$

where ε'_k is the corrupted elevation measure at time *k*. From geometrical considerations (the proof can be found in [1] and [9]), the elevation error is given by:

$$\Delta\hat{\varepsilon}_k = \frac{1}{2}\rho'_k \frac{N_0 b e^{-bh}\cdot 10^{-6}}{1+N_0 e^{-bh}\cdot 10^{-6}}\cos(\varepsilon'_k), \tag{2.17}$$

where ρ'_k is the corrupted range measure at time *k*. It is important to note that such measures, ρ'_k and ε'_k, are considered free from the measurement noise, then, before using eq. (2.17) to evaluate the tropospheric elevation error, we have to filter (e.g. with a Kalman filter) the radar measurements.

2.4.2 Evaluation of tropospheric range error

We now derive an estimator of the tropospheric range error from the corrupted radar measures (without measurement noise). As pointed out in [10], the tropospheric range error can be evaluated using the following path integral:

$$\Delta\hat{\rho} = \int_{s_0}^{s_1}\left(n(s)-1\right)ds = \int_{s_0}^{s_1} n(s)\,ds - \rho, \tag{2.18}$$

where $n(s)$ is the refractive index and ds is the infinitesimal element of the straight ray path between radar and target. Actually, the integral in eq. (2.18) is not equivalent to the one in eq. (2.9) because the integral in eq. (2.18) is a path integral along the straight line between radar and target, while it should be performed along the curved path as in eq. (2.9). In other words, evaluating the range error as shown in eq. (2.18) implies that the contribution to the range error due to the ray bending is neglected, and only the contribution due to the slowing down of the EM signal is taken into account. However, in Section VI, using a simulated

scenario, we show that such approximation yields a very small error . As pointed out before, due to the small value of *n*, a scaled EM refractivity index can be defined as [10]:

$$N(s) \triangleq \left(n(s)-1\right)10^6. \tag{2.19}$$

Through a change of variable (see Fig. 2.3 and Appendix B), the integral in (2.18) can be expressed as:

$$\Delta\hat{\rho}_k = 10^{-6}\int_{r_1}^{r_{2,k}} N(r)\frac{rdr}{\sqrt{r^2 - r_1^2\cos^2\varepsilon_k}}, \tag{2.20}$$

where $r_1=r_0+h$ is the distance from the Earth's centre to the radar (r_0 is the Earth's radius and h is the radar altitude, as pointed out before), ε_k is the true target elevation angle at time k, $r_{2,k}$ is the distance between the Earth's centre and the target at time k, and $N(r)$ is the EM refractivity index as function of r. The term r_1 is a priori known, since both Earth's radius and radar altitude are known. To evaluate the term $r_{2,k}$, we use the Carnot's theorem:

$$r_{2,k} = \sqrt{r_1^2 + \rho_k'^2 + 2r_1\rho'\sin\varepsilon_k}, \tag{2.21}$$

where ρ_k' is the corrupted range measure while ε_k is the true target elevation value. This means that, before evaluating the terms in (2.20) and (2.21), we must perform the elevation error correction using eq. (2.17).

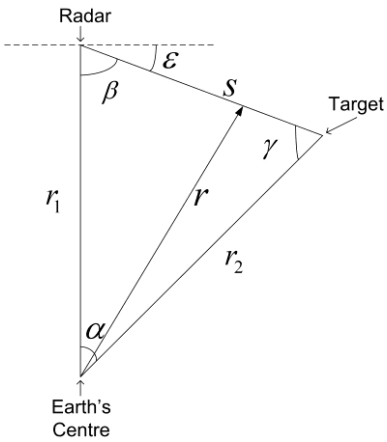

Figure 2.3 - Geometry for the change of variable.

To describe how the refractive index, or equivalently the scaled EM refractivity index defined in (2.19), varies with the altitude, we do not use the exponential model as previously, but the Hopfield model [14]. This model is a polynomial model and, for this reason, it is possible to perform a closed form solution of the integral in eq. (2.20) [10], whereas it would not be possible using the exponential model. Following [14], $N(r)$is given by:

$$N(r) = \sum_{i=1}^{2} N_i \left(1 - \frac{r - r_0}{H_i}\right)^4 \left[u(r - r_0) - u(r - (r_0 + H_i))\right], \tag{2.22}$$

where N_1 and N_2 are the surface dry component and the surface wet component of the refractive index respectively, and they are linked to the sea-level refractive index, involved in the exponential model, through the following linear relation [15]:

$$N_0 = N_1 + N_2. \tag{2.23}$$

Moreover, N_1 and N_2 can be evaluated, as functions of the surface pressure p_0 (in mbar), of the surface temperature T_0 (in Kelvin) and of the partial pressure of water vapour e_0 (in mbar), through the classical Smith and Weintraub formulas [2, Ch. 3], [15]:

$$N_1 = 77.6 \frac{p_0}{T_0}, \tag{2.24}$$

$$N_2 = 3.73 \cdot 10^5 \frac{e_0}{T_0^2}, \tag{2.25}$$

where the numerical coefficients in eqs. (2.24) and (2.25) are empirically determined. The parameters H_1 and H_2 are physical parameters, named dry and wet characteristic heights of the troposphere, and depend on the seasons and on the particular geographical area and they must be chosen accordingly with on-site measurements. An example of how to determine H_1 and H_2 from the on-site meteorological measurements can be found in [16] where it is proposed to use a linear regression method. The function $u(r)$ is the classical unit step function. By substituting eq. (2.22) in the integral in eq. (2.20), we obtain an estimate of the range error $\Delta\hat{\rho}_k$. It is important to remark that we have used two different models for the refractive index for the refraction error estimate: the exponential model for the estimate of the elevation error and the Hopfield model for the estimate of the range error. The exponential model is a 2-term model, i. e. it is fully defined by the parameters N_0 and b,

while the Hopfield model is a 4-term model, it is characterized by four parameters: N_1, N_2, H_1 and H_2. As discussed previously, there exists a linear relation, given in eq. (2.23), between the surface refractivity parameters of the two models. Unfortunately, in the open literature, it is not possible to find a mathematical relation that links the scale height b of the exponential model with the characteristic heights H_1 and H_2 of the Hopfield model. However, although the exponential and the Hopfield models are not fully inter-changeable due to the lack of a relation among b, H_1 and H_2, this discrepancy yields negligible difference between the exponential and Hopfield refractive index profile. As empirical proof of this assumption, a comparison between the exponential and the Hopfield refractivity index curves is given in Section 2.4.

In [10], a closed form solution for the integral in eq. (2.20) is derived, but the geometry is quite different. We derive a general solution, independent of the particular geometry that can be used for all radar-target configurations. In Appendix B all the mathematical details are given, here we report only the main facts:

1. If the target altitude is greater than the radar platform altitude (this is the geometry considered in [14]), then the term by term integration of eq. (2.20) leads to:

$$\Delta\hat{\rho}_k = 10^{-6}\sum_{i=1}^{2} N_i\left[I_i\left(r_{2,k}\right) - I_i\left(r_1\right)\right], \tag{2.26}$$

   where the function $I(\cdot)$ is defined in [10] (see Appendix B).

2. If the target altitude is smaller than the radar platform altitude, we must consider two cases. By defining $J_k \triangleq \dfrac{{r_{2,k}}^2 + \rho_k'^2 - r_1^2}{2r_{2,k}\rho_k'}$, we have:

   - If $J_k<0$, then the term by term integration of eq. (2.20) leads to:

$$\Delta\hat{\rho}_k = 10^{-6}\sum_{i=1}^{2} N_i\left[I_i\left(r_1\right) - I_i\left(r_{2,k}\right)\right], \tag{2.27}$$

   - Otherwise, if $J_k \geq 0$, then the term by term integration of eq. (2.20) leads to:

$$\Delta\hat{\rho}_k = 10^{-6}\sum_{i=1}^{2} N_i\left[I_i\left(r_1\right) + I_i\left(r_{2,k}\right) - 2I_i\left(r_1\cos\varepsilon_k\right)\right]. \tag{2.28}$$

It is important to note that eqs. (2.26), (2.27) and (2.28) depend on the corrupted range ρ_k' but on the real elevation angle ε_k as well. This means that, before estimating the range error, we have to correct the elevation measure by subtracting the estimated elevation error in eq. (2.17) to the corrupted elevation value ε_k'.

2.5. MODIFIED KF FOR TROPOSPHERIC ERROR CORRECTION

In the previous section we have discussed a way to estimate, from the corrupted radar measurements, the tropospheric range and elevation errors, $\Delta\hat{\rho}_k$ and $\Delta\hat{\varepsilon}_k$, respectively. Now, we propose a tracking algorithm that mitigates the tropospheric errors from the estimated state vector, i.e. position and velocity vector of the target. The proposed algorithm is a modified version of the extended Kalman filter (EKF) and it is composed of two blocks (see Fig. 2.4): in the first block it estimates the corrupted position vector, "free" from the measurement noise (as the classical KF), while in the second block it corrects the tropospheric range and elevation errors.

In the first block we apply the KF with converted measurement (CMKF) [17], [18]. This approach is shown to be more accurate in terms of accuracy of the position and velocity vector estimation than the EKF. At the output of this first block, we obtain an estimate of the corrupted position vector, then we are able to evaluate the range and elevation errors from this vector using (2.17) and (2.20). Finally, a new state update equation for the state vector estimation is proposed in order to correct the tropospheric errors.

2.5.1 The discrete-time model for target motion and the measurement model

We define here the signal and the measurement models for the tracking filter. As discussed also in Section 1.2, the target state vector $\mathbf{x}_k$ evolves according to the following discrete-time stochastic model [19], [20], [21]:

$$\mathbf{x}_k = \mathbf{T}\mathbf{x}_{k-1} - \mathbf{g}_{k-1} + \mathbf{w}_k, \tag{2.29}$$

where:

- $\mathbf{x}_k$ is the target state vector defined as $\mathbf{x}_k = \left(\mathbf{s}_k^T \quad \mathbf{v}_k^T\right)^T$ where $\mathbf{s}_k$ and $\mathbf{v}_k$ are the target position and velocity vectors in Cartesian coordinates,
- $\mathbf{T} = \begin{pmatrix} \mathbf{I}_{3\times3} & \mathbf{I}_{3\times3}T \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{pmatrix}$ is the state update matrix,
- $\mathbf{g}_k$ is a deterministic known vector due to the non-linearity of the platform motion. The mathematical derivation of this term is discussed in Section 1.2 [21]. Here, for brevity, we report only the analytical expression:

$$\mathbf{g}_k = \int_{kT}^{(k+1)T} \begin{pmatrix} \left((k+1)T - \tau\right)\mathbf{a}_a(\tau) \\ \mathbf{a}_a(\tau) \end{pmatrix} d\tau, \tag{2.30}$$

  where $\mathbf{a}_a(t)$ is the known, continuous-time, aircraft acceleration vector,
- $\mathbf{w}_k$ is a discrete zero-mean Gaussian random process that stands for any unforeseen disturbances in the target motion model, commonly named process noise. The covariance matrix of this process is [19], [20]:

$$\mathbf{Q}_w = \begin{pmatrix} \mathbf{Q}_w^{11} & \mathbf{Q}_w^{12} \\ \mathbf{Q}_w^{21} & \mathbf{Q}_w^{22} \end{pmatrix}, \tag{2.31}$$

$$\mathbf{Q}_w^{11} = \frac{T^3}{3}\tilde{\mathbf{N}}, \quad \mathbf{Q}_w^{12} = \frac{T^2}{2}\tilde{\mathbf{N}}, \quad \mathbf{Q}_w^{22} = T\tilde{\mathbf{N}}, \tag{2.32}$$

where $\tilde{\mathbf{N}} = \mathrm{diag}(N_{0,x}, N_{0,y}, N_{0,z})$, and $N_{0,x}$, $N_{0,y}$ and $N_{0,z}$ are the power spectral densities of the continuous-time process noise components.

The radar measurements, without tropospheric errors, defined in (2.2), are related to the target state via the measurement equation:

$$\mathbf{z}_k = \mathbf{h}^{-1}\left(\mathbf{H}\mathbf{x}_k\right) + \mathbf{n}_k, \tag{2.33}$$

where $\mathbf{h}(\cdot)$ is the spherical-to-Cartesian coordinates transformation and $\mathbf{H}$ is a matrix defined as $\mathbf{H} = \left(\mathbf{I}_{3\times 3} \quad \mathbf{0}_{3\times 3}\right)$. The measurement noise process $\mathbf{n}_k$ is assumed to be independent of the process noise $\mathbf{w}_k$. To introduce the tropospheric errors in the measurement model, they must be added to the radar measures in eq. (2.33) as shown explicitly in eq. (2.1).

2.5.2 Modified Kalman Filter for tropospheric error correction

In the following, the proposed tracking algorithm is described step-by-step. Before that, some mathematical details about the coordinate transformations of the corrupted position vector have to be explained. For ease of notation, in the following equations the dependence by *k* will be omitted. Let $\mathbf{s}$ and $\mathbf{s}'$ be the true and the corrupted target position vectors defined in Cartesian coordinates as:

$$\mathbf{s} = \begin{pmatrix} \rho\cos\theta\cos\varepsilon \\ \rho\sin\theta\cos\varepsilon \\ \rho\sin\varepsilon \end{pmatrix}, \tag{2.34}$$

and

$$\mathbf{s}' = \begin{pmatrix} \left(\rho+\Delta\rho\right)\cos\theta\cos\left(\varepsilon+\Delta\varepsilon\right) \\ \left(\rho+\Delta\rho\right)\sin\theta\cos\left(\varepsilon+\Delta\varepsilon\right) \\ \left(\rho+\Delta\rho\right)\sin\left(\varepsilon+\Delta\varepsilon\right) \end{pmatrix}, \tag{2.35}$$

Through some algebra, it can be shown that the corrupted position vector $\mathbf{s}'$ can be expressed as function of the true target position $\mathbf{s}$ as:

$$\mathbf{s}' = \cos\left(\Delta\varepsilon\right)\mathbf{s} + \mathbf{b}, \tag{2.36}$$

where $\mathbf{b}$ is the vector

$$\mathbf{b}=\begin{pmatrix}\Delta\rho\cos\theta\cos\varepsilon\cos\Delta\varepsilon-(\rho+\Delta\rho)\cos\theta\sin\varepsilon\sin\Delta\varepsilon\\ \Delta\rho\sin\theta\cos\varepsilon\cos\Delta\varepsilon-(\rho+\Delta\rho)\sin\theta\sin\varepsilon\sin\Delta\varepsilon\\ \Delta\varepsilon\sin\varepsilon\cos\Delta\varepsilon+(\rho+\Delta\rho)\cos\varepsilon\sin\Delta\varepsilon\end{pmatrix}. \tag{2.37}$$

The input of the CMKF at time *k* is the vector of the *converted measurements* defined as [21]:

$$\mathbf{s}'^{u}_{m,k}=\mathbf{h}^{u}\left(\mathbf{z}'_{k}\right)=\begin{pmatrix}\lambda_{\theta}^{-1}\lambda_{\varepsilon}^{-1}z'_{1,k}\cos z'_{2,k}\cos z'_{3,k}\\ \lambda_{\theta}^{-1}\lambda_{\varepsilon}^{-1}z'_{1,k}\sin z'_{2,k}\cos z'_{3,k}\\ \lambda_{\varepsilon}^{-1}z'_{1,k}\sin z'_{3,k}\end{pmatrix}, \tag{2.38}$$

where $\lambda_{\theta}^{-1}=e^{-\sigma_{\theta}^{2}/2}$ and $\lambda_{\varepsilon}^{-1}=e^{-\sigma_{\varepsilon}^{2}/2}$ are two coefficients that correct the bias errors produced by the non-linear transformation of the measurement noise.

Now, we describe the proposed filtering algorithm.

1. Initialization: both corrupted and the error free target state vectors and the error covariance matrix are initialized following the approach in [22] as:

$$\hat{\mathbf{x}}'_{1|1}=\hat{\mathbf{x}}_{1|1}=\left(\mathbf{s}'^{u\ T}_{m,1}\quad\mathbf{0}^{T}_{3\times1}\right)^{T}, \tag{2.39}$$

and

$$\mathbf{P}_{1|1}=\begin{pmatrix}\mathbf{C}_{\mathbf{s}'}&\mathbf{0}_{3\times3}\\ \mathbf{0}_{3\times3}&\mathbf{V}\end{pmatrix}, \tag{2.40}$$

where the matrix $\mathbf{C}_{\mathbf{s}'}$ represents the covariance matrix of the unbiased measurement vector defined in [18], and the matrix $\mathbf{V}=\mathrm{diag}\left(v_{x\max}^{2}/3,v_{y\max}^{2}/3,v_{z\max}^{2}/3\right)$ is a diagonal matrix where $v_{x\max}$, $v_{y\max}$ and $v_{z\max}$ are the maximum possible speed of a target, thus represent a sort of a priori information ($v_{\max}^{2}/3$ is the values of the variance of a uniformly distributed random variable defined in $\left[-v_{\max},v_{\max}\right]$).

2. For each *k*, the corrupted target state vector and the Kalman gain matrix $\mathbf{K}_k$ are estimated through the CMKF:

$$\left(\hat{\mathbf{x}}'_{k|k},\mathbf{K}_{k}\right)=\mathrm{CMKF}\left[\mathbf{s}'^{u}_{m,k},\hat{\mathbf{x}}'_{k|k-1}\right]. \tag{2.41}$$

The purpose of the next steps is to remove the atmospheric errors from $\hat{\mathbf{x}}'_{k|k}$:

3. Coordinate transformation from Cartesian-to-spherical:

$$\left(\hat{\rho}'_{k|k} \quad \hat{\theta}'_{k|k} \quad \hat{\varepsilon}'_{k|k}\right)^T = \mathbf{h}^{-1}\left(\mathbf{H}\hat{\mathbf{x}}'_{k|k}\right). \tag{2.42}$$

4. The elevation error is evaluated through eq. (2.17) and corrected:

$$\hat{\varepsilon}_{k|k} = \hat{\varepsilon}'_{k|k} - \Delta\hat{\varepsilon}_{k|k}. \tag{2.43}$$

5. The range error $\Delta\hat{\rho}_{k|k}$ is evaluated through eq. (2.20).

At this point we have both range and elevation errors at time *k*. In order to obtain the unbiased target state vector, we have to apply a suitable measure update equation.

Then, the remaining two steps are:

6. Prediction of the unbiased target state vector according with the dynamic model in eq. (2.29):

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{T}\hat{\mathbf{x}}_{k-1|k-1} - \mathbf{g}_{k-1}. \tag{2.44}$$

7. Taking into account eq. (2.36), a suitable measure update equation can be formulated as follows :

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\left(\mathbf{s}'^{u}_{m,k} - \left(\cos\left(\Delta\hat{\varepsilon}_{k|k}\right)\mathbf{H}\hat{\mathbf{x}}_{k|k-1} + \hat{\mathbf{b}}_{k|k}\right)\right), \tag{2.45}$$

where $\mathbf{K}_k$ is the Kalman gain matrix evaluated at step 2.

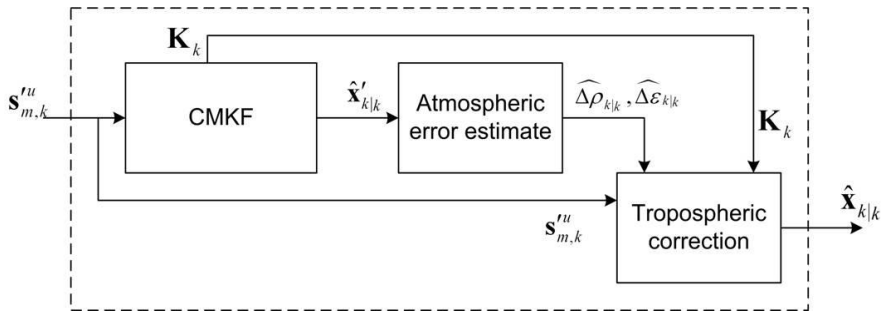

Figure 2.4 - Algorithm for the tropospheric correction: the input is the converted measurement vector, while the output is the estimated state vector.

2.6 SIMULATION RESULTS

The physical tropospheric parameters, surface pressure, temperature and partial pressure of water vapour, used in the simulations have chosen according to the International Standard Atmosphere (ISA) specifications [23], as follows: p_0=1013.25 mbar, T_0=288.15 K and e_0=15 mbar. From eqs. (2.24) and (2.25), it is possible to obtain the values of the surface dry and wet refractivity component as N_1=273 and N_2=67.17, then the surface refractivity component can be obtained from eq. (2.23) as N_0=340.17. For the scale height b of the exponential model, we have chosen the standard value of b=1/7000=1.4286·10^{-4} m^{-1}. Finally, the characteristic heights H_1 and H_2 for the Hopfield model have to be determined from the on-site measurements or from the data stored in the IGRA [16], as discussed in Section IV.B. In our simulations, we have assumed two reasonable values for H_1 and H_2 [10]: H_1=42819 m and H_2=12000 m. As concerning the parameters that characterize the radar system and the target, all the simulations have been performed using the following radar parameters: σ_ρ = 2.4 m, $\sigma_\theta = \sigma_\varepsilon$ = 0.25°, T=1 s, the flight height h is 1000 m and the platform speed is v_p = 90 m/s. The detection and false alarm probabilities are assumed to be 1 and 0, respectively. The target is modeled as a high speed dinghy [11] with a velocity vector defined as $\mathbf{v}_t$ = (7.63 6.76 0) m/s and a standard deviation for each component of the continuous-time process noise vector equal to $N_{0,x} = N_{0,y} = N_{0,z}$ = 0.01 m^2/s^3. The geometrical parameters of the aircraft racetrack are (see Fig. 1.1): d = 60 km and R = 10 km. The time intervals needed to cover the straight and circular segment of the racetrack are T_l=667 s and T_c=350 s respectively, then the racetrack period is of about 2000 s. First, we want to validate the two assumptions made in Section 2.4.B: the inter-changeability between the exponential and the Hopfield model and the "straight line" integration in eq. (2.18). In Fig. 2.5, a comparison between the exponential and the Hopfield refractivity index profile, using the tropospheric parameters introduced before, is shown. Although the two models are not fully inter-changeable for the reasons discussed in Section 2.4.2, it can be noted that the differences are negligible at the considered quote (about 1000 m). The second assumption is the "straight line" approximation used to derive the estimate of the range error. Using the tropospheric parameters introduced before, a comparison between the theoretical value of the range error, given by eq. (2.9) through integration along the curved ray path, and the estimated value, given by eq. (2.18) through straight line integration, is shown in Fig. 2.6 as function of the radar-target distance. As we can see, the

differences between the theoretical and the estimated values become relevant beyond 300 km. Since our simulations are performed with a radar-target distance of about 70-100 km, then the straight line approximations is valid.

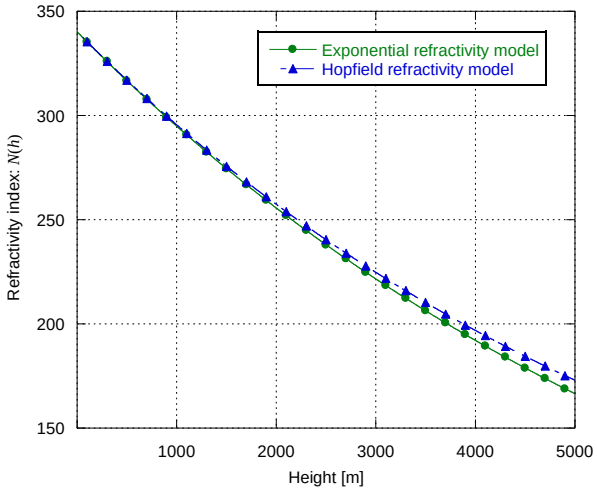

Figure 2.5 Comparison between the exponential and the Hopfield refractivity profiles for the tropospheric parameters given in Section 2.6.

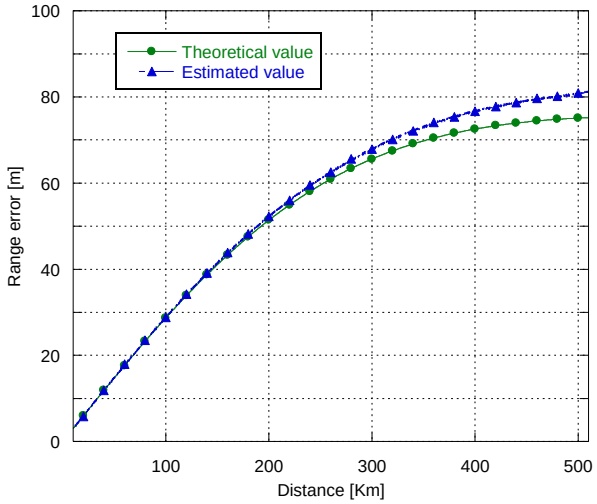

Figure 2.6 - Comparison between the theoretical (from eq. (2.11)) and the estimated value (from eq. (2.18)) of the range as function of the radar-target distance in km.

Now, we can start to describe the procedure that we have used to assess the performance of the proposed tracking filter in a refractive scenario. The geometry of the scenario is described in Section 2.2 and a visual representation is given in Fig. 1.1. All the simulations are referred to a medium range scenario, where the radar-target distance varies in a range of 70-100 km. It is important to note that a small radar-target distance involves low incident elevation angle of the ray path and this leads to the ducting effect discussed in the Introduction. A relation to evaluate the critical elevation angle, as function of the radar height and of the refractivity index, is given in [6, Ch. 3]. The choice of the medium range scenario allows us to avoid the ducting effect that is not addressed by the mathematical propagation model assumed in this paper. On the other hand, at the considered radar height of about 1 km, a radar-target distance greater than 110 km would involve over-the-horizon path propagation and neither this phenomenon is tackled in our propagation model. As discussed in the previous Sections, all the fundamental physical tropospheric parameters, i.e. temperature, humidity, partial pressure of water vapour and so on, have to be evaluated by means of on-site measurements or of stored data in IGRA. Both these procedures will introduce an uncertainty on the evaluation of the dry and wet refractivity index, of the scale height b and of the characteristic heights H_1 and H_2. In order to take into account this model uncertainty, we define $\tilde{N}_0$ and $\tilde{b}$ as two Gaussian random variables with mean value and standard deviation such that $\tilde{N}_0 \sim \mathcal{N}(N_0, \sigma_N^2)$ and $\tilde{b} \sim \mathcal{N}(b, \sigma_b^2)$, where $N_0=N_1+N_2$. During the data generation, we have considered the exponential refractivity index in eq. (2.7) as a parametric random process with random parameters given by $\tilde{N}_0$ and $\tilde{b}$. In the filtering phase, to implement the two estimators for the range and elevation errors we have used the nominal values of the refractivity index N_0 and of the scale height b, i. e. the mean values of the random variables $\tilde{N}_0$ and $\tilde{b}$. This introduces a model mismatch that allows us to evaluate the effect of the uncertainty introduced in the evaluation of the physical tropospheric parameters. For the standard deviation of $\tilde{N}_0$ and $\tilde{b}$, i.e. σ_N and σ_b respectively, we have chosen three different values of 2%, 5% and 10% of the respective nominal values. In Fig. 8 and Fig. 9, we show the standard deviation of the estimation error for range and elevation errors, $\sigma_{e_{\Delta\rho,k}}$ and $\sigma_{e_{\Delta\varepsilon,k}}$ respectively, defined as: $e_{\Delta\rho,k} = \Delta\rho_k - \Delta\hat{\rho}_k$ and $e_{\Delta\varepsilon,k} = \Delta\varepsilon_k - \Delta\hat{\varepsilon}_k$, where $\Delta\rho_k$ and $\Delta\varepsilon_k$ are the theoretical values defined in eqs. (2.11),

(2.14), obtained by solving the Dirichlet's problem in eq. (2.18) using a refractive index with random parameters; while $\Delta\hat{\rho}_k$ and $\Delta\hat{\varepsilon}_k$ are the estimated values obtained by the estimator in eqs. (2.17) and (2.20) where we have used the nominal value of the parameters to define the relative refractive index. The standard deviation (std) of the estimation errors is evaluated using 100 independent Monte Carlo runs. The periodic-like behavior of both elevation and range errors is due to the quasi-elliptical aircraft racetrack (see Fig. 1.1). As we can see, the std of the estimation error for the range error, i.e. $\sigma_{e_{\Delta\rho,k}}$, is about 4 m (Fig. 2.7) while for the elevation error, i.e. $\sigma_{e_{\Delta\varepsilon,k}}$, is about 0.025° (Fig. 2.8) in the worst case, i.e. when σ_N=0.1N_0 and σ_b=0.1b. Obviously, by decreasing the model mismatch, i.e. by decreasing the variances σ_N^2 and σ_b^2, the std of the estimation error decreases, too. For σ_N=0.05N_0 and σ_b=0.05b, we have that $\sigma_{e_{\Delta\rho,k}}$ is lower than 2 m and $\sigma_{e_{\Delta\varepsilon,k}}$ is lower than 0.015°, while for σ_N=0.02N_0 and σ_b=0.02b, $\sigma_{e_{\Delta\rho,k}}$ is lower than 1 m and $\sigma_{e_{\Delta\varepsilon,k}}$ is lower than 0.01°. Further simulations, not included here for lack of space, show that the 80% -90% of the error std of the range error is due to a mismatch in the value of the refractivity index N_0, while the mismatch in the scale height causes a negligible effect. On the other hand, for the elevation error, the contribution to the error std of the mismatch in the refractivity value and in scale height are almost equivalent.

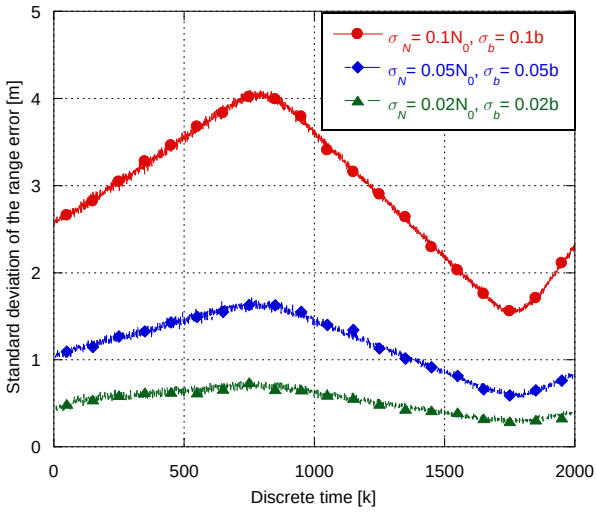

Figure 2.7 - Standard deviation of the estimated range error.

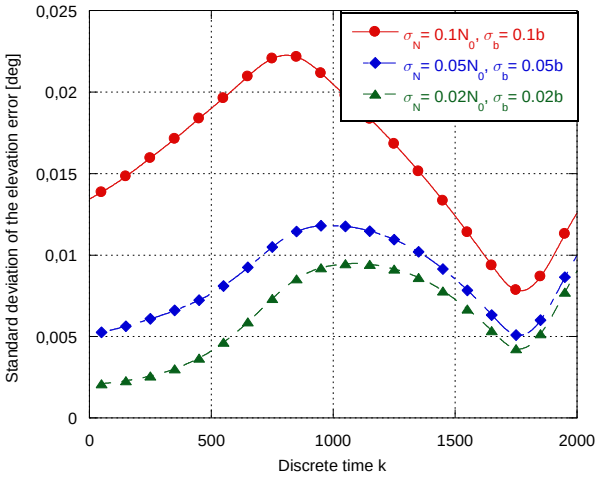

Figure 2.8 - Standard deviation of the estimated elevation error.

Finally, in Figs 2.9-2.16, the performance of the proposed modified KF-based algorithm are analyzed in terms of mean value and std of the estimation error for each component of the state vector. By defining the estimation error for the *i*th component of the state vector as $e_{i,k} = \hat{x}_{i,k} - x_{i,k}$, the error mean value and the error std are defined as follow: $\mu_{e_{i,k}} = E\{e_{i,k}\}$ and $\sigma_{e_{i,k}} = E\{(e_{i,k} - \mu_{e_{i,k}})^2\}^{1/2}$, respectively. All performance curves have been obtained by averaging over 100 independent Monte Carlo runs. For brevity, here we only show the results relative to the *x* and *z* components of both position and velocity vectors, since the results for the *y* component are similar to the *x* ones. Moreover, the most important coordinate to check the proposed algorithm is the *z* component, because it is the one more heavily affected by tropospheric errors. In fact, it is affected by both range and elevation errors, whereas the *x* and *y* components are affected only by the range error. In the following figures, we show three different curves: (1) the ideal case, i.e. the radar measurements have no tropospheric errors; (2) the case in which the measurements are affected by tropospheric errors and the correction of such errors is performed using the proposed KF-based algorithm; (3) the case in which the measurements are corrupted but no correction is made. For the position estimate, the bias error due to the tropospheric error, in the worst case is, of about -70 m (Fig. 2.9) and 310 m (Fig. 2.11) for the *x* and *z* components of position vector respectively. It can be noted that, the proposed algorithm is

able to mitigate the impact of the troposheric errors in the bias of the estimates, in fact, the curves relative to the KF-based correction algorithm are pretty close to the ones relative to the ideal case. In Figs. 2.10 and 2.12, the error std relative to the *x* and *z* components of the position vector are shown. As we can see, for the *x* component, the three curves (ideal case, without correction and with correction) are all close to each other. This means that the random model mismatch, introduced in the data generation, does not affect significantly the filter performance in the estimate of the *x* component of the position vector. In Fig. 2.12, the error std relative to the *z* component of position is shown. In this case, in a time interval between 500 and 1000 sec (that correspond to the turn of the platform that carries the radar), the two curves relative to the cases with correction and without correction differ from the ideal case of about 20 m. This performance degradation is due to the model mismatch and the proposed algorithm is not able to correct such effect, in fact the curve relative to the case "with correction" is equal to the case "without correction". For the velocity vector estimate, the bias error for the *x* and *y* components is negligible, while for the *z* component is of about $\pm$ 0.5 m/s in the worst case (Figs. 2.13 and 2.15). Also in this case, the proposed algorithm seems to be able to reject the bias of the estimate produced by the refraction errors. The curves of the error std in the three cases (ideal case, without correction and with correction) are close to each other (see Figs. 2.14 and 2.16). As before, this means that the model mismatch does not affect the performance of the components of the velocity vector. Finally, it can be noted that, as pointed out before, the difference in the magnitude of the bias error of both position and velocity vector estimate between the *x* and *y* components and the *z* component is due to the fact that the *z* component is affected by both elevation and range errors, while the bias error in *x* and *y* components is only due to the range error. A comment has to be made about the particular progress of the error mean value and std curves: this periodic-like behavior is caused by the non-linear relative motion between the radar and the target that is composed of the periodic racetrack of the platform that carries the radar and of the straight line motion of the target. From this simulation results, we can assert that the proposed KF-based algorithm is able to mitigate the effects of the tropospheric range and elevation errors on the bias of the estimate of both position and velocity vector. Moreover, it seems to be a robust algorithm with respect to the model mismatch introduced in the simulated data. Of course, a deeper investigation on the behavior of the proposed algorithm with more realistic models for the tropospheric refractive index is needed.

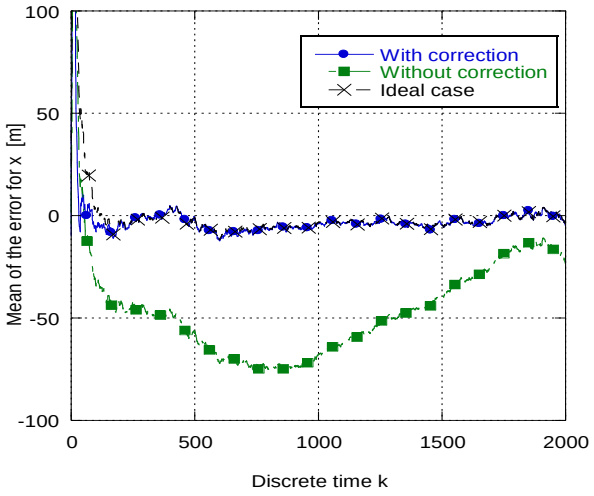

Figure 2.9 - Error mean value for the *x* component of the position vector.

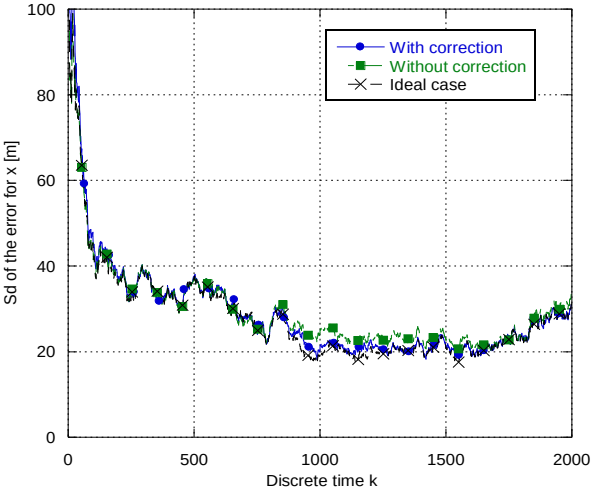

Figure 2.10 - Error std for the *x* component of the position vector.

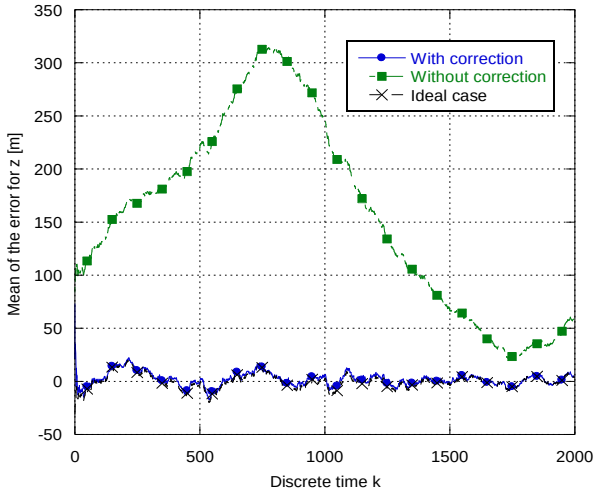

Figure 2.11 - Error mean value for the *z* component of the position vector.

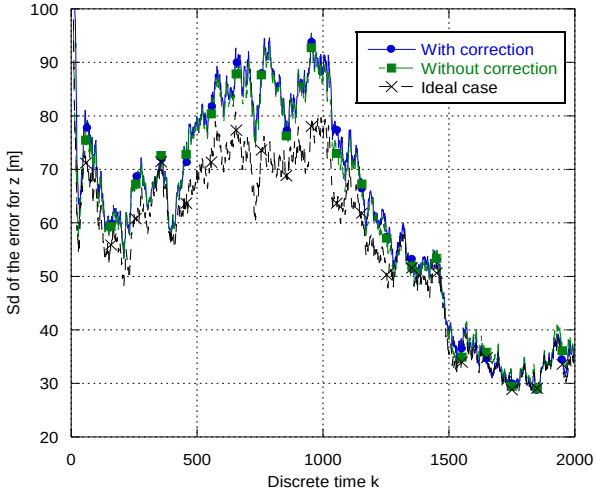

Figure 2.12 - Error mean value for the *z* component of the position vector.

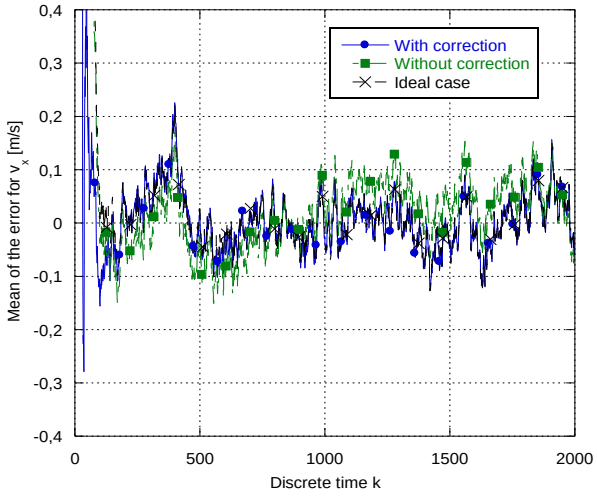

Figure 2.13 - Error mean value for the *x* component of the velocity vector.

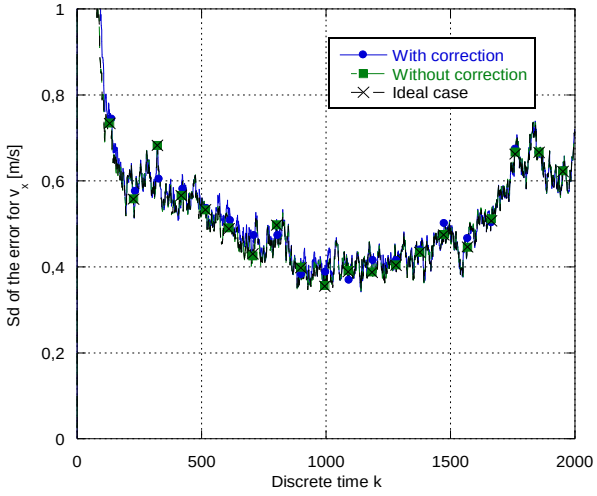

Figure 2.14 - Error std for the *x* component of the velocity vector.

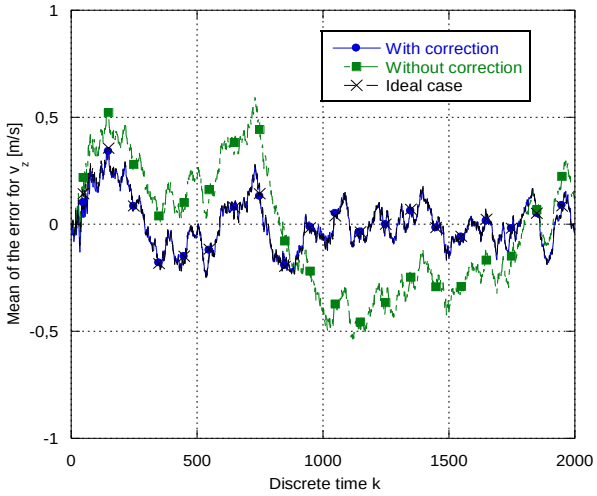

Figure 2.15 - Error mean value for the *z* component of the velocity vector.

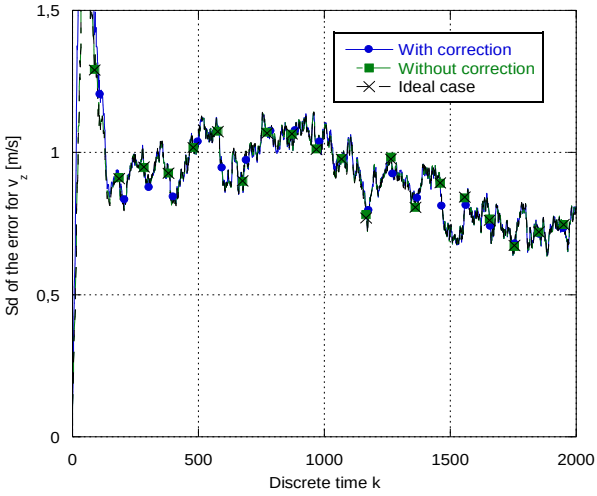

Figure 2.16 - Error std for the *z* component of the velocity vector.

2.7 Summary

In this Chapter, we propose a tracking algorithm, based on the KF, with the aim to mitigate the effects of the tropospheric errors. A simplified mathematical model of the

tropospheric propagation of the EM radar signal has been introduced. The assumed model does not take into account anomalous propagation effects (e.g. ducting effect or small fluctuation of the refractive index) and the error in the azimuth measure is assumed to be negligible. According to this simplified model, two algorithms for the estimate of such errors from the corrupted radar measurements were derived and their performance investigated. Such estimators assume two different models for the tropospheric refractive index: the exponential model for the estimate of the elevation error and the Hopfield model for the estimate of the range error. Although these two models are not fully inter-changeable, we have shown that such discrepancy leads to a negligible error. Finally, a modification of the classical KF has also been proposed to remove the tropospheric errors from the state vector estimate. The performance of the proposed algorithm has been investigated for a medium range scenario in the presence of a random mismatch between the model used for the data generation and the one used in the tropospheric error estimations. The choice of the medium range scenario is motivated by observing that for small radar-target distance, it implies a low elevation angle, the propagation of the radar signal could experience the ducting affect that, as said before, it is not considered in our ray tracing model. On the other hand, it must be noted that the proposed algorithm could present a performance degradation for a very long range scenarios (such as e.g. satellite geometries), due to the straight line assumption used to derive the estimator for the range error.

Simulation results show the effectiveness of the proposed algorithm and its ability to mitigate the effects of the tropospheric errors on the estimated target state vector. Moreover, it seems to be robust with respect to model mismatch. However, it must be remarked that, in our simulations, we have used a simplified model to describe the refractive index that is not able to fully characterize its variability. For this reasons a deeper investigation of the effects of the anomalous propagation effects and of the tropospheric random fluctuations is needed and it will be the subject of future research.

References

[1] M. Denny, "Refracted propagation effects for airborne radar", *IEEE International Radar Conference*, Washington, DC, 7-12 May 2000, pp. 554-559.

[2] W. Morchin**,** *Airborne Early Warning Radar*, Norwood, MA: Artech House, 1990.

[3] A. K. Shukla, D. J. Fraser, A. Lockton, A. H. Y. Chan, J. E. N. Field, and S. L. Shemar, "The Impact of Tropospheric Propagation on Data Fusion from Multiple Radars", *7th International Conf. on Information Fusion*, Philadelphia, PA, USA, 25-28 July 2005, vol. 1, pp. 290-296.

[4] D. K. Barton and H. R. Ward, *Handbook of Radar Measurement*, Norwood, MA: Artech House, 1984.

[5] V. I. Tatarski**,** *Wave Propagation in a Turbulent Medium*, New York: McGraw-Hill, 1961.

[6] R. J. Fitzgerald, "Random tropospheric effects in command guidance by multilateration and interferometry," *IEEE Transactions on Aerospace and Electronic Systems*, vol.30, no.1, pp. 249-257, Jan 1994.

[7] A. V. Prokopov, "Elongation of electromagnetic wave trajectories on paths with localized refractive index inhomogeneities of air," *Radiophysics and Quantum Electronics*, vol. 24, no. 11, pp. 889-892, 1982.

[8] S. Fortunati, F. Gini, M. Greco, A. Farina, A. Graziano, S. Giompapa, F. R. Castella, "Correction of refracted propagation effects for airborne radar tracking," *IEEE International Radar Conference*, 2010, Washington, DC, 10-14 May 2010, pp.578-583.

[9] C. A. H. Wright, S. F. Haase, and M. Y. Shen, "Techniques for improving target altitude estimations in an air target tracking system", *International Radar Symposium 1998*, Munich, Germany, 15-17 September 1998, vol. 2, pp. 685-694.

[10] F. R. Castella, "Intratropospheric range delay for two refractivity models", *IEE Proc.-Radar, Sonar, Navig.*, vol 145, no. 2, pp. 119-122, April 1998.

[11] S. Giompapa, A. Farina, F. Gini, A. Graziano, R. Di Stefano, “Computer Simulation of an Integrated Multi-Sensor System for Maritime Border Control,” *IEEE Radar Conference, 2007*, Boston, USA, 17-20 April 2007, pp.308-313.

[12] A. R. Forsyth, *Calculus of Variations*, New York: Dover, 1960, pp. 17-20 and 29.

[13] IGRA database: http://www.ncdc.noaa.gov/oa/climate/igra/index.php.

[14] H. S. Hopfield, “Two-quartic tropospheric refractivity profile for correcting satellite data”, *J. Geophys. Res.*, vol. 74, no. 18, pp. 4487-4499, 1969.

[15] E. K. Smith, N. Weintraub, “The constants in the equation for atmospheric refractive index at radio frequencies,” *J. Res. NBS*, vol. 50, no. 1, pp. 39–41, 1953.

[16] K. Saha, K. Parameswaran and C. Suresh Raju, “Tropospheric delay in microwave propagation for tropical atmosphere based on data from the Indian subcontinent”, *Journal of Atmospheric and Solar-Terrestrial Physics*, vol. 69, pp. 875–905, 2007.

[17] D. Lerro, and Y. Bar-Shalom, “Tracking with debiased consistent converted measurements versus EKF”, *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 29, no. 3, pp 1015-1022, July 1993.

[18] M. Longbin, S. Xiaoquan, Z. Yiyu, S. Z. Kang, and Y. Bar-Shalom, “Unbiased converted measurements for tracking”, *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 34, no. 3, pp 1023-1027, July 1998.

[19] Y. Bar-Shalom, X. R. Li, T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms, and Software*, New York: Wiley, 2001.

[20] X. R. Li, V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *Aerospace and Electronic Systems, IEEE Transactions on* , vol.39, no.4, pp. 1333- 1364, Oct. 2003.

[21] S. Fortunati, M. Greco, F. Gini, A. Farina, A. Graziano, “Impact of flight disturbances on tracking airborne radar”, *International Radar Symposium 2009*, Hamburg, Germany, 09-11 September 2009, pp. 407-411.

[22] M. Mallick, and B. La Scala, “Comparison of single-point and two-point difference track initiation algorithm using position measurements”, *Acta Automatica Sinica*, vol. 34, no. 3, pp. 258-265, March 2008.

[23] International Organization for Standardization, *Standard Atmosphere*, ISO 2533:1975, 1975.

Appendix A

Here, the mathematical manipulations that allow us to solve the Euler-Lagrange equation in eq. (2.6) and obtain the Dirichlet's problem in eq. (2.8) are discussed. Rewrite the Euler-Lagrange equation:

$$\frac{d}{du}\left(\frac{\partial}{\partial \dot{w}}F\left(u,w,\dot{w}\right)\right)=\frac{\partial}{\partial w}F\left(u,w,\dot{w}\right) \tag{A1}$$

where

$$F\left(u,w,\dot{w}\right)\triangleq\sqrt{1+\left(\dot{w}(u)\right)^{2}}\,n\left(u,w\right) \tag{A2}$$

and, according with the notation used before, $\dot{w}(u)=dw(u)/du$.

In the following, for ease of notation, we write the partial first or second order derivatives of the function F as: $\frac{\partial}{\partial w}F(u,w,\dot{w})\triangleq F_{w}(u,w,\dot{w})$ or, for example, $\frac{\partial^{2}}{\partial w\partial \dot{w}}F(u,w,\dot{w})\triangleq F_{w\dot{w}}(u,w,\dot{w})$. Moreover, we can define a vector function $\mathbf{g}(u)=(u \quad w(u) \quad \dot{w}(u))^{T}$, then the function $F(u,w,\dot{w})$ can be rewritten as $F(\mathbf{g}(u))$. Using this notation, the Euler-Lagrange equation can be rewritten as:

$$\frac{d}{du}\left(F_{\dot{w}}\left(\mathbf{g}(u)\right)\right)=F_{w}\left(\mathbf{g}(u)\right) \tag{A3}$$

Now, we can apply the chain rule of derivation to the first term of eq. (A3):

$$\frac{d}{du}\left(F_{\dot{w}}\left(\mathbf{g}(u)\right)\right)=\nabla F_{\dot{w}}\left(\mathbf{g}(u)\right)\cdot\dot{\mathbf{g}}(u), \tag{A4}$$

where $\nabla F_{\dot{w}}(\mathbf{g}(u))=(F_{\dot{w}u}(\mathbf{g}(u)) \quad F_{\dot{w}w}(\mathbf{g}(u)) \quad F_{\dot{w}\dot{w}}(\mathbf{g}(u)))$ is the gradient of $F_{\dot{w}}(\mathbf{g}(u))$ with respect to $\mathbf{g}(u)$ and $\dot{\mathbf{g}}(u)=(1 \quad \dot{w}(u) \quad \ddot{w}(u))^{T}$. So, by performing the classical matrix product, we get:

$$\frac{d}{du}\left(F_{\dot{w}}\left(\mathbf{g}(u)\right)\right)=F_{\dot{w}u}\left(\mathbf{g}(u)\right)+\dot{w}(u)F_{\dot{w}w}\left(\mathbf{g}(u)\right)+\ddot{w}(u)F_{\dot{w}\dot{w}}\left(\mathbf{g}(u)\right) \tag{A5}$$

and, by substituting eq. (A5) in eq. (A3), we obtain the differential equation of the path $w(u)$. Finally, by using the exponential atmospheric model in eq. (2.7) and by imposing the boundary value conditions, we obtain the Dirichlet's problem in eq. (2.8).

Appendix B

In this Appendix, we give all the mathematical details to evaluate the integral in eq. (2.20). The geometry of the problem is shown in Fig. 3. In the following, we give the proof of the change of variable to pass from the integral in eq. (2.18) to the one in eq. (2.20). First, from the geometry shown in Fig. 3, it can be noted that:

$$\cos^2 \varepsilon = \sin^2 \beta \tag{B1}$$

Since the mathematical formulation becomes more easy if we consider the angle β instead of the angle ε, we can formulate all the equations as a function of the angle β and finally, using the equality in eq. (B1), they can be rewritten as function of ε. From the Carnot's theorem (or law of cosines), we have:

$$r^2 = r_1^2 + s^2 - 2r_1 s \cos\beta \tag{B2}$$

and, by differentiation of both terms in eq. (B2), we get:

$$2rdr = 2sds - 2r_1 \cos\beta ds \tag{B3}$$

that implies:

$$ds = \frac{r}{s - r_1 \cos\beta} dr \tag{B4}$$

From eq. (B2), through some algebraic manipulations and by using the trigonometry equality $\sin^2 \beta = 1 - \cos^2 \beta$, we get:

$$r^2 - r_1^2 \sin^2 \beta = \left(s - r_1 \cos\beta\right)^2 \tag{B5}$$

that implies:

$$s - r_1 \cos\beta = \pm\sqrt{r^2 - r_1^2 \sin^2 \beta} \tag{B6}$$

By substituting eq. (B6) in eq. (B4), we get the desired change of variable:

$$ds = \pm \frac{r}{\sqrt{r^2 - r_1^2 \sin^2 \beta}} dr \tag{B7}$$

Finally, using the equality in (B1), we can rewrite this change of variable as a function of the elevation angle ε:

$$ds = \pm \frac{r}{\sqrt{r^2 - r_1^2 \cos^2 \varepsilon}} dr \tag{B8}$$

To perform the integral in eq. (2.20), we must distinguish three different cases to assure the monotonicity of the change of variable on the integration domain. Here we list all the possible cases:

1. If $\beta \geq \frac{\pi}{2}$, the integral in eq. (2.20) is:

$$\Delta\hat{\rho} = 10^{-6} \int_{r_1}^{r_2} N(r) \frac{rdr}{\sqrt{r^2 - r_1^2 \cos^2 \varepsilon}} \tag{B9}$$

   in fact the variable r increases monotonically from r_1 to r_2.

2. If $\beta < \frac{\pi}{2}$, we have two different cases:

   - If $\gamma \geq \frac{\pi}{2}$, that implies, from the Carnot's theorem, that $\frac{r_2^2 + s^2 - r_1^2}{2sr_2} < 0$,

     the integral in eq. (2.20) becomes:

$$\Delta\hat{\rho} = -10^{-6} \int_{r_1}^{r_2} N(r) \frac{rdr}{\sqrt{r^2 - r_1^2 \cos^2 \varepsilon}} \tag{B10}$$

     in fact the variable r decreases monotonically from r_1 to r_2.

   - If $\gamma < \frac{\pi}{2}$, that implies, from the Carnot's theorem, that $\frac{r_2^2 + s^2 - r_1^2}{2sr_2} \geq 0$,

     the integral in eq. (2.20) becomes:

$$\Delta\hat{\rho} = -10^{-6}\int_{r_1}^{r_1\sin\beta} N(r)\frac{rdr}{\sqrt{r^2 - r_1^2\cos^2\varepsilon}} + 10^{-6}\int_{r_1\sin\beta}^{r_2} N(r)\frac{rdr}{\sqrt{r^2 - r_1^2\cos^2\varepsilon}} \qquad \text{(B11)}$$

in fact the variable r decreases monotonically from r_1 to $r_1\sin\beta$, then increases monotonically from $r_1\sin\beta$ to r_2. It can be noted that $r_1\sin\beta = r_1\cos\varepsilon$.

Now, following [14], we give a closed form expression for the primitive function for the integral in eq. (B9) (and then for all other integrals in eqs. (B10) and (B11)). We define the primitive function $I(r)$ as:

$$\int N_i(r)\frac{rdr}{\sqrt{r^2 - r_1^2\cos^2\varepsilon}} = I_i(r) + C \qquad \text{(B12)}$$

where C is a real constant. Substituting one term of the sum in eq. (2.22) into eq. (B12), expanding the quadratic term and performing term by term integration, we obtain the expression of the primitive [14]:

$$\begin{aligned} I_i(r) = \sqrt{r^2 - A^2}\Bigg[& a_i + \frac{b_i r}{2} + \frac{C_i}{3}\left(2A^2 + r^2\right) + \\ & + d_i\left(\frac{r}{4}\left(r^2 + \frac{3A^2}{2}\right)\right) + e_i\left(\frac{1}{5}\left(r^4 + \frac{4A^2r^2}{3} + \frac{8A^4}{3}\right)\right)\Bigg] + \\ & + \left[\frac{b_i A^2}{2} + \frac{3d_i A^4}{8}\right]\ln\left(r + \sqrt{r^2 - A^2}\right), \end{aligned} \qquad \text{(B13)}$$

where:

$A = r_1\cos\varepsilon$, $a_i = 1 + 4x_i + 6x_i^2 + 4x_i^3 + x_i^4$, $b_i = -\frac{4}{H_i}\left(1 + 3x_i + 3x_i^2 + x_i^3\right)$,

$c_i = \frac{6}{H_i^2}\left(1 + 2x_i + x_i^2\right)$, $d_i = -\frac{3}{H_i^3}\left(1 + x_i\right)$, $e_i = \frac{1}{H_i^4}$, and $x_i = \frac{r_0}{H_i}$.

Part II: The grid-locking problem

Chapter 3: The relative grid-locking problem

3.1 Introduction

Interest in integrating a set of stand-alone sensors into an integrated multisensor system has been increasing in the last few years. Rather than to develop new sensors to achieve more accurate tracking and surveillance systems, it is more useful to integrate existing stand-alone sensors into a single system in order to obtain performance improvements [1], [2]. However, an important prerequisite for successful multisensor integration is the sensor registration process. The problem of sensor registration arises when a set of data coming from two or more sensors must be combined. This problem involves the coordinate transformation and the reciprocal alignment among the various sensors: streams of data from different sensors must be converted into a common coordinate system (or frame) and aligned before they could be used in a tracking or surveillance system. If not corrected, the registration errors can seriously degrade the global system performance by increasing tracking errors and even introducing ghost tracks. In naval system, the process of automatic registration is often referred as "grid-locking" process. For brevity, in the following we use such term to define a general registration process regardless of the particular application.

A first basic distinction is usually made between *relative* grid-locking and *absolute* grid-locking. The relative grid-locking process aligns remote data to local data under the assumption that the local data are bias free and that all biases reside with the remote sensor. The problem is that, actually, also the local sensor is affected by biases that cannot be corrected by means of this approach. The absolute grid-locking process assumes that all the sensors in the scenario are affected by errors that must be corrected. One source of registration errors is represented by the sensor calibration (or offset) errors, also called measurement errors. Although the sensors are usually initially calibrated, the calibration may deteriorate over time. There are three measurement errors, one for each component of

the measurement vector, i.e. range, azimuth, and elevation. Another kind of registration errors are the attitude or orientation errors. Attitude errors can be caused by biases in the gyros of the inertial measurement unit (IMU) of the sensor. There are three possible attitude errors, one for each body-fixed rotation axis. The last source of registration errors is represented by the location (or position) errors caused by bias errors in the navigation system associated with the sensors.

Various algorithms for sensor bias estimation have been proposed in the literature, both for relative and absolute grid-locking process. These algorithms fall into two main classes. A first class formulates the grid-locking problem as a track association problem. Two examples of this class of algorithms can be found in [3] and [4]. In [3] a registration algorithm for two radars is proposed, whereas in [4] a similar procedure is applied to a scenario composed by an active sensor and a passive sensor. The scenario with two active sensors is not investigated in [4]. A second class of algorithms does not use a track-level data, but simply a plot-level data. To estimate the registration errors, such algorithms need only a set of common detections (i.e. each radars in the system must detect the same target at the same time). Since the algorithm derived in this work falls into this second class, in the following a detailed summary of the existing algorithms for this class is provided.

In [5] a 2-D scenario with two radars is investigated. Both sensors are assumed to be biased, but only the measurement errors in range and azimuth are taken into account, whereas the elevation error is neglected. The measurement model is linearized and the measurement range and azimuth errors are estimated using a maximum likelihood estimator (MLE). The Cramér-Rao lower bound (CRLB) is then evaluated for the linearized data model. A similar scenario (two 2D radars both affected by the range and azimuth measurement errors) is considered in [6] and [7, vol. 2, Ch. 5]. A linearized form of the congruence or *alignment equation*[5] is derived where the unknowns are the range and azimuth measurements errors of the two radars. The CRLB is not provided there. In [8] a relative grid-locking problem is addressed for a two radar system in a 2-D scenario. Only an attitude bias error (the north angle error) and two location bias errors are taken into account. A least squares (LS) algorithm with covariance weighting is derived, but the CRLB is not provided. In [2], the relative grid-locking problem is addressed for two 3-D

[5] The *alignment equation* is obtained by expressing the condition for the alignment between the measurements taken by the two considered radars looking at the same target in the same time instant, as detailed in Sect. II.

radars. Both measurement and attitude errors are considered, and the relative location of the two radars is assumed perfectly known, i.e. location errors are not present. The estimation algorithm is a Kalman Filter (KF) built from a linearized alignment model. A strong limitation to the applicability of this algorithm is that the maximum separation distance between the two radars must be 100m. The CRLB is not provided. The possibility to use a KF-based algorithm to estimate the registration errors is investigated also in [9]. The absolute grid-locking problem is addressed in [10] and [11]. All radars in the scenario are considered biased and both the flat model and the ellipsoidal model for the Earth are investigated. However, only the measurement bias errors are taken into account. This limits quite a lot the usefulness of the proposed algorithm. The estimation strategy for the registration errors is based on MLE approach. In [11], the CRLB is evaluated under a linearized measurement model for three different target trajectories. This analysis shows that the estimation performance is strongly dependent on the target trajectory chosen in the simulations. Finally, in a recent works, the use of the Expectation-Maximization (EM) [12] algorithm is proposed to solve the sensor registration problem [13], [14], [15] and [16]. In [13], the authors apply the EM algorithm to estimate the target state vector when the measurement model is not completely known. Moreover, they apply the same algorithm to an absolute grid-locking problem considering only a subset of all the possible registration error, more precisely only the measurement bias errors. In [14], the authors deal with a data fusion problem; they track a cooperative vehicle using the measurements coming from dissimilar sensors (Differential GPS, Inertial Navigation System, radar and camera). Then, in order to have an unbiased parameter estimate, the EM algorithm is incorporated within the Kalman filter to give simultaneous state and registration parameters estimate. Finally, in [15] and in [16], the same approach was applied to a radar network system but, also in this case, only the measurement bias errors were considered.

This Chapter concerns with the relative grid-locking problem. Unlike previous works, all the registration errors (i.e. measurement, attitude, and location errors) are taken into account in the data model. We propose two different estimation algorithm: a linear Least Squares (LS) algorithm and an Expectation-Maximization-based algorithm. To derive the linear LS algorithm [17], an alignment equation that involves all the errors is established to transform and align the data coming from the biased radar (radar #2) to the reference frame of the unbiased radar (radar #1). Afterwards, in order to obtain a linearized data model, a

first order Taylor expansion of the alignment equation is derived. Numerical results have shown that the LS estimator is not an efficient estimator for most of the registration biases. Such non-efficiency could be caused by the (strong) linearization implied by the linear LS algorithm. Then, in order to obtain a more efficient estimation algorithm, an Expectation-Maximization (EM) algorithm [12] is derived. There are various motivations for the application of the EM algorithm to solve this problem. First of all, the EM algorithm is able to deal with estimation problems that involve an incomplete data set (as in our case). Second, the iteration of such algorithm converges to the maximum likelihood (ML) estimate under some regularity assumptions [18]. This means that the EM algorithm is, at least asymptotically, an efficient estimator. Third, by using the EM algorithm we can obtain at the same time the estimate of the registration bias errors and the estimate of the target state vector. Fourth, the EM algorithm can be extended to solve the absolute grid-locking problem.

The two estimation algorithms are derived under the following assumptions: (1) one of the two radars is assumed unbiased, i.e. free of registration errors (relative grid-locking process); (2) the registration biases are time-invariant during the observation interval; (3) K synchronous pairs of measures coming from a common target are available; (4) the Earth model is the flat model.

The performance of the proposed algorithms is compared with the so-called hybrid Cramér-Rao Lower bound (HCRLB) [19], [20], [21]. The need to use this performance bound instead of the conventional CRLB arises from the prohibitive analytical complexity in the derivation of the conventional one. Unlike [5] and [11], in the calculation of the HCRLB, no hypothesis of linearity of the model is made and all possible bias errors are taken into account.

The last part of this Chapter deals with the identifiability problem [22], [23], [24] and its particular application to the estimate of the grid-locking errors. The identifiability problem concerns with the possibility of drawing inferences from observed data to an underlying theoretical structure. This is equivalent to saying that different structures may generate different probability distributions of the observable data in order to make the structures "observable". First, the general identifiability problem is discussed and rigorously defined starting from fundamental works on this topic ([22],[23],[23]). Our attention is focused on the parametric models, i.e. such models in which every structure can

be represented by a vector in $\mathbb{R}^m$. The parametric models are a wide class of models and, for sure, it is the most useful class in practical applications (see e.g. [25] and [26, Ch. 6]). However, in many practical applications, the data model is affected by additional random parameters whose estimation is not strictly required, the so-called *nuisance parameters* [27], [28]. In these cases, the classical definition of identifiability, which requires calculation of the Fisher Information Matrix (FIM) and of its rank, is often difficult or impossible to do. Instead, the Modified Fisher Information Matrix (MFIM) [20], [29], [30] can be computed.

Here, we generalize the main results on the identifiability problem to take the presence of random nuisance parameters into account. We provide an alternative definition of identifiability, that can be always applied but that is weaker than the classical definition, and we investigate the relationships between the identifiability condition and the MFIM. Finally, we apply the obtained results on the identifiability in presence of nuisance parameters to the relative grid-locking problem.

The rest of the Chapter is organized as follows. Section 3.2 provides a geometrical description of the scenario and introduces all the parameters involved in the relative grid-locking process. The measurement models of both unbiased (radar #1) and biased (radar #2) sensors and the target state model are described in Sections 3.3 and 3.4, respectively. The LS estimator of the sensor registration errors is derived in Section 3.5 while the EM-based algorithm is derived in Section 3.6. In Section 3.7, the HCRLB is evaluated. Numerical analysis of the accuracy of the two proposed estimators and the comparison with the HCRLB is reported in Section 3.8. In Section 3.9, the EM algorithm is generalized to the case of a multi-target scenario and the performance improvements with respect to the single target scenario are discussed. In Section 3.10, the general identifiability problem is analyzed. Moreover, this general framework is used to validate some intuitive result on the indentifiability of the grid-locking errors.

3.2 The relative grid-locking problem

The geometry of the scenario for the relative grid-locking problem is shown in Fig. 1 [17]. The main parameters are:

- $(x_{S_1}, y_{S_1}, z_{S_1})$: radar #1 reference system. Radar #1 is assumed to be ideal, then its reference system coincides with the absolute system.
- $(x_{S_2}, y_{S_2}, z_{S_2})$: radar #2 reference system.
- $\mathbf{r}_k$: true target position vector in the absolute reference system.
- $\mathbf{t}_t$: true position vector of radar #2 in the absolute reference system.
- $\mathbf{q}_k$: true target position vector in radar #2 reference system.

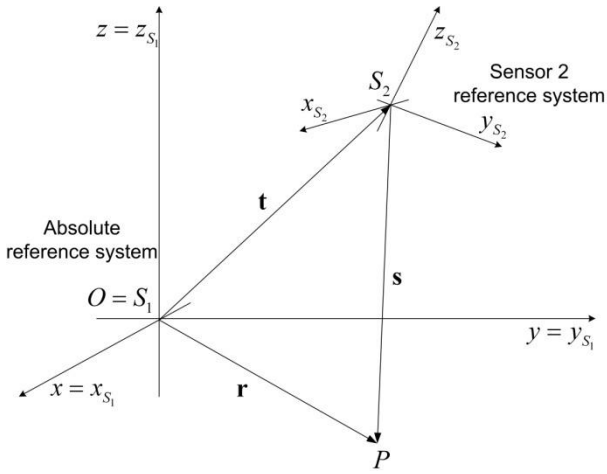

Figure 3.1 - Geometry of the relative grid-locking problem.

From the geometry of the problem (Fig. 3.1), the following relation holds:

$$\mathbf{r}_k = \mathbf{R}(\chi, \psi, \xi)\mathbf{q}_k + \mathbf{t}_t, \tag{3.1}$$

where **R** is 3x3 the rotation matrix of angles χ, ψ and ξ that aligns the radar #2 reference frame to the radar #1 reference frame. Matrix **R** is given by:

$$\mathbf{R}(\chi,\psi,\xi)=\begin{pmatrix}\cos\xi\cos\psi & \cos\xi\sin\psi\sin\chi+\sin\xi\cos\chi & -\cos\xi\sin\psi\cos\chi+\sin\xi\sin\chi\\ -\sin\xi\cos\psi & -\sin\xi\sin\psi\sin\chi+\cos\xi\cos\chi & \sin\xi\sin\psi\cos\chi+\cos\xi\sin\chi\\ \sin\psi & -\cos\psi\sin\chi & \cos\psi\cos\chi\end{pmatrix}. \tag{3.2}$$

The angles χ, ψ and ξ are named *roll*, *pitch* and *yaw*, and represent the rotation angles around *x*, *y* and *z* axes, respectively.

As pointed out before, there are three different types of biases: attitude, measurement, and location biases. In the rest of the paper, we use the following notation:

1. *Attitude biases*: $\mathbf{\Theta}_t=[\chi_t \quad \psi_t \quad \xi_t]^T$, $\mathbf{\Theta}_m=[\chi_m \quad \psi_m \quad \xi_m]^T$, and $d\mathbf{\Theta}=[d\chi \quad d\psi \quad d\xi]^T$ denote the true attitude angles vector, the measured attitude angles vector and the attitude bias errors vector, respectively.

2. *Measurement biases*: $\mathbf{v}_t^k=[\rho_t^k \quad \theta_t^k \quad \varepsilon_t^k]^T$, $\mathbf{v}_m^k=[\rho_m^k \quad \theta_m^k \quad \varepsilon_m^k]^T$, and $d\mathbf{v}=[d\rho \quad d\theta \quad d\varepsilon]^T$ denote the true target position vector, the measured target position, and the measurement bias errors vector, respectively. These vectors are defined in a spherical coordinate reference system.

3. *Location biases*: $\mathbf{t}_t=[t_{x,t} \quad t_{y,t} \quad t_{z,t}]^T$, $\mathbf{t}_m=[t_{x,m} \quad t_{y,m} \quad t_{z,m}]^T$, and $d\mathbf{t}=[dt_x \quad dt_y \quad dt_z]^T$ denote the true relative position vector, the measured relative position vector and the location bias errors vector. These vectors are defined in Cartesian coordinate reference system.

The convention adopted here is that the biases must be added to the measured value to obtain the true value of the specific parameter. According to this, we get the following equations for attitude angles (3.3), relative location vector (3.4), and measurement model (3.5):

$$\mathbf{\Theta}_t=\mathbf{\Theta}_m+d\mathbf{\Theta}, \tag{3.3}$$

$$\mathbf{t}_t=\mathbf{t}_m+d\mathbf{t}, \tag{3.4}$$

$$\mathbf{v}_m^k=\mathbf{v}_t^k-d\mathbf{v}+\mathbf{n}^k, \tag{3.5}$$

where $\mathbf{n}^k$ is the measurement noise modelled by Gaussian random vector with zero-mean and diagonal covariance matrix $\mathbf{C}_n$.

It must be noted that if the rotation around z axis is applied first, the azimuth measurement bias $d\theta$ and the yaw attitude bias $d\xi$ cannot be distinguished and have to be merged into a single bias. This point will be extensively discussed in Section 3.10 in the general framework of the identifiability problem. Because of this geometrical coupling, we can define a single bias error as $d\zeta=d\theta+d\xi$. To take this into account, we define the new vectors $d\tilde{\boldsymbol{\Theta}} = [d\chi \quad d\psi \quad d\zeta]^T$ and $d\tilde{\mathbf{v}} = [d\rho \quad 0 \quad d\varepsilon]^T$. The 8x1 unknown parameter vector is therefore given by:

$$\boldsymbol{\Phi} = \left[d\rho \quad d\varepsilon \quad d\chi \quad d\psi \quad d\zeta \quad dt_x \quad dt_y \quad dt_z \right]^T. \tag{3.6}$$

It is worth stressing, that in this work vector **r** refers to the position vector in Cartesian coordinates, while **v** refers to the position in spherical coordinates. The geometrical convention for the Cartesian coordinates system is that y axis is aligned with the North, x axis with the East, and z points upwards. For the spherical coordinates we have that the azimuth θ is positive clockwise from the North (y axis) and the elevation ε is positive counter-clockwise from the x-y plane. The spherical-to-Cartesian transformation is denoted by $\mathbf{h}(\cdot)$ and its inverse transformation, i.e. Cartesian-to-spherical, by $\mathbf{h}^{-1}(\cdot)$. Thus, we can rewrite eq. (3.1) as follows:

$$\mathbf{h}\left(\mathbf{v}_{1,t}^k\right) = \mathbf{R}\left(\boldsymbol{\Theta}_t\right) \cdot \mathbf{h}\left(\mathbf{v}_{2,t}^k\right) + \mathbf{t}_t, \tag{3.7}$$

where $\mathbf{v}_{1,t}^k$ and $\mathbf{v}_{2,t}^k$ are the true target position vector, in spherical coordinates, defined in radar #1 and radar #2 reference systems, respectively. Equation (3.7) is the *fundamental alignment equation*, which allows us to align the data coming from radar #2 to radar #1, that is assumed to be unbiased. By making use of (3.3), (3.4), and (3.5), equation (3.7) can be rewritten as:

$$\mathbf{h}\left(\mathbf{v}_{1,m}^k - \mathbf{n}_1^k\right) = \mathbf{R}\left(\boldsymbol{\Theta}_m + d\boldsymbol{\Theta}\right) \cdot \mathbf{h}\left(\mathbf{v}_{2,m}^k + d\mathbf{v} - \mathbf{n}_2^k\right) + \left(\mathbf{t}_m + d\mathbf{t}\right), \tag{3.8}$$

where the registration errors appears explicitly.

Equations (3.7) and (3.8) refer to a single observation. The goal here is to estimate vector parameter $\boldsymbol{\Phi}$, defined by (3.6), based on K observations of the two sensor

measurements, $\left\{\mathbf{v}_{1,m}^{k},\mathbf{v}_{2,m}^{k}\right\}_{k=1}^{K}$. The number K observations is the number of target detections we use for sensor registration.

3.3 The measurements model

The true target position in the absolute reference frame and the true target position in radar #2 reference system associated to the kth observation are denoted by $\mathbf{r}_k$ and $\mathbf{q}_k$, respectively. Since radar #1 is assumed to be unbiased, its reference system can be assumed as the absolute reference frame. Under this assumption, which characterizes the relative grid-locking problem, the radar #1 measurement model is:

$$\mathbf{v}_{1,m}^{k}=\mathbf{h}^{-1}\left(\mathbf{r}_k\right)+\mathbf{n}_1^{k}, \tag{3.9}$$

where the measurement noise vector $\mathbf{n}_1{}^k$ is a zero-mean Gaussian distributed vector with covariance matrix $\mathbf{C}_1=\mathrm{diag}(\sigma_{\rho,1}^2,\sigma_{\theta,1}^2,\sigma_{\varepsilon,1}^2)$. To derive the radar #2 measurement model, recast equation (3.1) as a function of the bias errors as follows:

$$\mathbf{r}_k=\mathbf{R}\left(\mathbf{\Theta}_m+d\mathbf{\Theta}\right)\mathbf{q}_k+\left(\mathbf{t}_m+d\mathbf{t}\right). \tag{3.10}$$

Solving (3.10) for $\mathbf{q}_k$ yields:

$$\mathbf{q}_k=\mathbf{R}^T\left(\mathbf{\Theta}_m+d\mathbf{\Theta}\right)\left[\mathbf{r}_k-\left(\mathbf{t}_m+d\mathbf{t}\right)\right], \tag{3.11}$$

where we used the fact that $\mathbf{R}$ is a rotation matrix, so $\mathbf{R}^{-1}=\mathbf{R}^T$. Then, by applying the inverse coordinate transformation and by adding the measurement bias errors and the measurement noise, we get [17]:

$$\begin{aligned}\mathbf{v}_{2,m}^{k}&=\mathbf{h}^{-1}\left(\mathbf{R}^T\left(\mathbf{\Theta}_m+d\mathbf{\Theta}\right)\left[\mathbf{r}_k-\left(\mathbf{t}_m+d\mathbf{t}\right)\right]\right)-d\mathbf{v}+\mathbf{n}_2^{k}\\&\triangleq\boldsymbol{\mu}\left(\mathbf{r}_k;\mathbf{\Phi}\right)+\mathbf{n}_2^{k},\end{aligned} \tag{3.12}$$

where we used eq. (5), $\mathbf{v}_{2,m}^{k}=\mathbf{v}_{2,t}^{k}-d\mathbf{v}+\mathbf{n}_2^{k}$, and the fact that $\mathbf{v}_{2,t}^{k}=\mathbf{h}^{-1}(\mathbf{q}_k)$. The measurement noise $\mathbf{n}_2{}^k$ is a zero-mean Gaussian distributed vector with covariance matrix $\mathbf{C}_2=\mathrm{diag}(\sigma_{\rho,2}^2,\sigma_{\theta,2}^2,\sigma_{\varepsilon,2}^2)$. The definition of $\boldsymbol{\mu}(\mathbf{r}_k;\mathbf{\Phi})$ immediately derives from (3.12).

Summarizing, the pdf of the measurements coming from the two radars, conditioned to the unobservable target position, are given by:

$$\mathbf{v}_{1,m}^{k}\left|\mathbf{r}_{k}\right. \sim \mathcal{N}\left(\mathbf{h}^{-1}\left(\mathbf{r}_{k}\right),\mathbf{C}_{1}\right), \tag{3.13}$$

$$\mathbf{v}_{2,m}^{k}\left|\mathbf{r}_{k}\right. \sim \mathcal{N}\left(\boldsymbol{\mu}\left(\mathbf{r}_{k},\boldsymbol{\Phi}\right),\mathbf{C}_{2}\right). \tag{3.14}$$

3.4 Target kinematic model

In literature, different models can be found to describe the target kinematic model via a state vector: in [15], for example, a multiple model, composed of the constant velocity model, the constant acceleration model and the coordinate turn model, is assumed. A different scenario was considered in [17], where it was assumed that independent detections coming from multiple targets were available for solving the grid-locking problem. Hence, the target position vector was assumed to be uniformly distributed in a given surveillance volume. Here, we choose for the target trajectory a simple constant velocity model [31]. Under this assumption, the discrete target state vector is defined as $\mathbf{s}_k = [\mathbf{r}_k^T \quad \mathbf{v}_k^T]^T$, where $\mathbf{r}_k = [x_k \quad y_k \quad z_k]^T$ and $\mathbf{v}_k = [v_{x,k} \quad v_{y,k} \quad v_{z,k}]^T$ are the target position vector and the target velocity vector defined in a 3-dimensional Cartesian coordinate system. The stochastic discrete state model is:

$$\mathbf{s}_{k+1} = \mathbf{F}\mathbf{s}_k + \mathbf{w}_k, \tag{3.15}$$

where **F** is a block matrix defined as:

$$\mathbf{F} = \begin{pmatrix} \mathbf{I} & T\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}, \tag{3.16}$$

where **I** is the 3x3 identity matrix and T is the sensor sampling interval. The process noise term $\mathbf{w}_k$ represents the random acceleration vector, which is assumed to be a white Gaussian noise process with zero mean and covariance matrix given by:

$$\mathbf{Q}_w = q\begin{pmatrix} \boldsymbol{\Gamma}_1 & \boldsymbol{\Gamma}_2 \\ \boldsymbol{\Gamma}_2 & \boldsymbol{\Gamma}_3 \end{pmatrix}, \tag{3.17}$$

where q is a noise parameter and $\mathbf{\Gamma}_1$, $\mathbf{\Gamma}_2$ and $\mathbf{\Gamma}_3$ are three matrices defined as $\mathbf{\Gamma}_1 = T^3/3\mathbf{I}$, $\mathbf{\Gamma}_2 = T^2/2\mathbf{I}$, $\mathbf{\Gamma}_3 = T\mathbf{I}$ respectively.

3.5 The linear least squares (LS) algorithm

In order to handle the nonlinear transformation of the measurement noise **n**, we adopt here the *unbiased conversion function* from spherical-to-Cartesian coordinates derived in [32]:

$$\mathbf{h}\left(\rho_m, \theta_m, \varepsilon_m\right) = \begin{pmatrix} \lambda_\theta^{-1}\lambda_\varepsilon^{-1}\rho_m \cos\varepsilon_m \sin\theta_m \\ \lambda_\theta^{-1}\lambda_\varepsilon^{-1}\rho_m \cos\varepsilon_m \cos\theta_m \\ \lambda_\varepsilon^{-1}\rho_m \sin\varepsilon_m \end{pmatrix}, \tag{3.18}$$

where $\lambda_\theta^{-1} = e^{-\sigma_\theta^2/2}$ and $\lambda_\varepsilon^{-1} = e^{-\sigma_\varepsilon^2/2}$ and σ_θ^2, σ_ε^2 are the variances of the measurement noise components for azimuth and elevation measurements. From (3.8), and taking into account the unbiased property of coordinate transformation (3.18), we have that:

$$E\left\{\mathbf{h}\left(\mathbf{v}_{1,m}^k\right) - \mathbf{R}\left(\mathbf{\Theta}_m + d\mathbf{\Theta}\right)\cdot\mathbf{h}\left(\mathbf{v}_{2,m}^k + d\mathbf{v}\right) - \left(\mathbf{t}_m + d\mathbf{t}\right)\right\} = 0. \tag{3.19}$$

Therefore, (3.8) can be expressed as:

$$\mathbf{h}\left(\mathbf{v}_{1,m}^k\right) = \mathbf{R}\left(\mathbf{\Theta}_m + d\mathbf{\Theta}\right)\cdot\mathbf{h}\left(\mathbf{v}_{2,m}^k + d\mathbf{v}\right) + \left(\mathbf{t}_m + d\mathbf{t}\right) + \boldsymbol{\varepsilon}, \tag{3.20}$$

where $\boldsymbol{\varepsilon}$ represents the un-modelled zero-mean error. The nonlinear least squares (NLLS) estimate of the unknown parameter vector $\mathbf{\Phi}$ can be obtained by minimizing the objective function:

$$J(\mathbf{\Phi}) = \left\|\mathbf{h}\left(\mathbf{v}_{1,m}^k\right) - \mathbf{R}\left(\mathbf{\Theta}_m + d\mathbf{\Theta}\right)\cdot\mathbf{h}\left(\mathbf{v}_{2,m}^k + d\mathbf{v}\right) - \left(\mathbf{t}_m + d\mathbf{t}\right)\right\|_2^2, \tag{3.21}$$

with respect to $\mathbf{\Phi}$, i.e. w.r.t. the $d\mathbf{\Theta}$, $d\mathbf{v}$ and $d\mathbf{t}$. Due to the heavy computational complexity of the nonlinear minimization search, which makes the estimator hard to implement in real time in a radar system, we resort here to a linear least squares (LS) estimator. To this purpose, we need to linearize the alignment equation (3.20).

A first-order approximation is obtained by performing the first-order Taylor expansion around the unknown parameters vectors. It must be noted that, because of the geometrical coupling between the azimuth measurement bias and the yaw attitude bias, the Taylor expansion has to be taken around the vectors $d\tilde{\boldsymbol{\Theta}}$, $d\tilde{\mathbf{v}}$, and $d\mathbf{t}$. Each entry of the rotation matrix $\mathbf{R}(\boldsymbol{\Theta}_m + d\tilde{\boldsymbol{\Theta}})$ can be linearized as [17]:

$$\left[\mathbf{R}\left(\boldsymbol{\Theta}_m + d\tilde{\boldsymbol{\Theta}}\right)\right]_{ij} \cong \left[\mathbf{R}\left(\boldsymbol{\Theta}_m\right)\right]_{ij} + \nabla\left[\mathbf{R}\left(\boldsymbol{\Theta}_m\right)\right]_{ij} d\tilde{\boldsymbol{\Theta}}, \tag{3.22}$$

where the term $\nabla[\mathbf{R}(\boldsymbol{\Theta}_m)]_{ij}$ is the gradient of the entry with indices i, j of the matrix $\mathbf{R}$, evaluated at the measured attitude angles vector $\boldsymbol{\Theta}_m$. Through some matrix manipulation, the second term of the sum in (3.22), can be rewritten in a matrix form as:

$$d\mathbf{R} = \nabla\left[\mathbf{R}\left(\boldsymbol{\Theta}_m\right)\right]_{i,\,j=1,2,3} d\tilde{\boldsymbol{\Theta}} = \left(\frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \chi} d\chi + \frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \psi} d\psi + \frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \xi} d\zeta\right), \tag{3.23}$$

$d\mathbf{R}$ is a 3×3 matrix where $\partial\mathbf{R}/\partial\chi$, $\partial\mathbf{R}/\partial\psi$ and $\partial\mathbf{R}/\partial\xi$ are the 3×3 matrices of the partial derivatives of $\mathbf{R}$ with respect to the roll, pitch and yaw angles evaluated at $\boldsymbol{\Theta}_m$.

The second nonlinear term in (3.20) is the spherical-to-Cartesian coordinate transformation $\mathbf{h}(\mathbf{v}_{2,m}^k + d\mathbf{v})$. As pointed out before, the second component of the measurement bias vectors $d\mathbf{v}$, i.e. $d\theta$, is merged with the yaw bias error $d\xi$ because of their geometrical coupling, then the linearization has to be made around $d\rho$ and $d\varepsilon$, or, in vector notation, around $d\tilde{\mathbf{v}}$. The first-order Taylor expansion for the coordinate transformation can be expressed as [17]:

$$\left[\mathbf{h}\left(\mathbf{v}_{2,m}^k + d\mathbf{v}\right)\right]_i \cong \left[\mathbf{h}\left(\mathbf{v}_{2,m}^k\right)\right]_i + \nabla\left[\mathbf{h}\left(\mathbf{v}_{2,m}^k\right)\right]_i d\tilde{\mathbf{v}}, \tag{3.24}$$

where $\nabla[\mathbf{h}(\mathbf{v}_{2,m}^k)]_i$ is the gradient of the ith component of the spherical-to-Cartesian unbiased coordinate transformation. The second term in the sum in (3.24) can be rewritten as:

$$d\mathbf{r}_2^k = \nabla\left[\mathbf{h}\left(\mathbf{v}_{2,m}^k\right)\right]_i d\tilde{\mathbf{v}} = \left(\frac{\partial \mathbf{h}\left(\mathbf{v}_{2,m}^k\right)}{\partial \rho} d\rho + \frac{\partial \mathbf{h}\left(\mathbf{v}_{2,m}^k\right)}{\partial \varepsilon} d\varepsilon\right)$$
$$= \left(\frac{\partial \mathbf{h}\left(\mathbf{v}_{2,m}^k\right)}{\partial \rho} \quad \mathbf{0}_{3\times 1} \quad \frac{\partial \mathbf{h}\left(\mathbf{v}_{2,m}^k\right)}{\partial \varepsilon}\right) d\tilde{\mathbf{v}}, \tag{3.25}$$

where $i = 1,2,3$, $\partial \mathbf{h}(\mathbf{v}_{2,m}^k)/\partial \rho$ and $\partial \mathbf{h}(\mathbf{v}_{2,m}^k)/\partial \varepsilon$ are 3×1 column vectors of partial derivatives of $\mathbf{h}(\mathbf{v}_{2,m}^k)$ with respect to the range ρ and the elevation ε, evaluated at the measured values $\rho_{2,m}^k$ and $\varepsilon_{2,m}^k$. The term $d\mathbf{r}_2^k$ is then a 3×1 column vector. By inserting (3.22) and (3.24) in (3.20), we obtain the *linearized alignment equation*:

$$\mathbf{h}\left(\mathbf{v}_{1,m}^k\right) = \left(\mathbf{R}\left(\boldsymbol{\Theta}_m\right) + d\mathbf{R}\right) \cdot \left(\mathbf{h}\left(\mathbf{v}_{2,m}^k\right) + d\mathbf{r}_2^k\right) + \left(\mathbf{t}_m + d\mathbf{t}\right) + \boldsymbol{\varepsilon}. \tag{3.26}$$

Then, through some algebra and neglecting the second-order term[6], we get:

$$\mathbf{h}\left(\mathbf{v}_{1,m}^k\right) = \mathbf{R}\left(\boldsymbol{\Theta}_m\right)\mathbf{h}\left(\mathbf{v}_{2,m}^k\right) + \mathbf{t}_m + \mathbf{R}\left(\boldsymbol{\Theta}_m\right) d\mathbf{r}_2^k + d\mathbf{R}\mathbf{h}\left(\mathbf{v}_{2,m}^k\right) + d\mathbf{t} + \boldsymbol{\varepsilon}. \tag{3.27}$$

Eq. (3.27) can now be recast in a compact form, useful for the direct application of the linear LS algorithm. The first three terms of (3.27) can be combined to define the "new" measurement vector $\mathbf{z}_k$ as:

$$\mathbf{z}_k \triangleq \mathbf{h}\left(\mathbf{v}_{1,m}^k\right) - \mathbf{R}\left(\boldsymbol{\Theta}_m\right)\mathbf{h}\left(\mathbf{v}_{2,m}^k\right) - \mathbf{t}_m. \tag{3.28}$$

Note that $\mathbf{z}_k$ is a function of all the available measurements. Eq. (3.28) basically represents the alignment equation between the two radars that we would obtained by neglecting the bias errors correction. If we now apply the distributive property of matrix multiplication, the term $d\mathbf{R}\mathbf{h}(\mathbf{v}_{2,m}^k)$ can be rewritten as:

[6] It can be noted here that the direct linearization of the product between the rotation matrix and spherical-to-Cartesian coordinate transformation, $\mathbf{R}(\cdot)\mathbf{h}(\cdot)$, with respect to the corresponding registration biases as a whole yields the same result obtained with the linearization procedure shown in eqs. (3.22), (3.24), and (3.27).

$$
\begin{aligned}
d\mathbf{R}\mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right) &= \left(\frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \chi} d\chi + \frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \psi} d\psi + \frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \xi} d\zeta \right) \mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right) \\
&= \left(\frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \chi} \mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right) d\chi + \frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \psi} \mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right) d\psi + \frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \xi} \mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right) d\zeta \right) \\
&= \left(\frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \chi} \mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right) \quad \frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \psi} \mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right) \quad \frac{\partial \mathbf{R}\left(\boldsymbol{\Theta}_m\right)}{\partial \xi} \mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right) \right) d\tilde{\boldsymbol{\Theta}} \\
&= \left(\mathbf{l}_{\chi}\left(\boldsymbol{\Theta}_m, \mathbf{v}_{2,m}^{k}\right) \quad \mathbf{l}_{\psi}\left(\boldsymbol{\Theta}_m, \mathbf{v}_{2,m}^{k}\right) \quad \mathbf{l}_{\xi}\left(\boldsymbol{\Theta}_m, \mathbf{v}_{2,m}^{k}\right) \right) d\tilde{\boldsymbol{\Theta}},
\end{aligned}
\tag{3.29}
$$

where $\mathbf{l}_{\chi,\psi,\xi}(\boldsymbol{\Theta}_m, \mathbf{v}_{2,m}^{k})$ are three 3×1 column vectors. Moreover, using eqs. (3.25) and (3.29), the last three terms of (3.27) can be rewritten in a compact matrix form as:

$$
\mathbf{R}\left(\boldsymbol{\Theta}_m\right) d\mathbf{r}_2^k + d\mathbf{R}\mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right) + d\mathbf{t} \triangleq \mathbf{H}_k \boldsymbol{\Phi}, \tag{3.30}
$$

where $\boldsymbol{\Phi}$ is the parameter vector to be estimated, defined by (6), and matrix $\mathbf{H}_k$ is defined as:

$$
\mathbf{H}_k = \left(\mathbf{R}\left(\boldsymbol{\Theta}_m\right) \frac{\partial \mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right)}{\partial \rho} \quad \mathbf{R}\left(\boldsymbol{\Theta}_m\right) \frac{\partial \mathbf{h}\left(\mathbf{v}_{2,m}^{k}\right)}{\partial \varepsilon} \quad \mathbf{l}_{\chi}\left(\boldsymbol{\Theta}_m, \mathbf{v}_{2,m}^{k}\right) \quad \mathbf{l}_{\psi}\left(\boldsymbol{\Theta}_m, \mathbf{v}_{2,m}^{k}\right) \quad \mathbf{l}_{\xi}\left(\boldsymbol{\Theta}_m, \mathbf{v}_{2,m}^{k}\right) \quad \mathbf{I}_3 \right), \tag{3.31}
$$

where $\mathbf{I}_3$ is the 3x3 identity matrix. Finally, using (3.28) and (3.30), the *linearized alignment equation* of (3.27) can be expressed in a compact matrix form as:

$$
\mathbf{z}_k = \mathbf{H}_k \boldsymbol{\Phi} + \boldsymbol{\varepsilon}, \quad \text{for } k = 1, \ldots, K. \tag{3.32}
$$

If we assume that K measurements are available, the LS problem can be recast in the following well-known form [17]:

$$
J_L(\boldsymbol{\Phi}) = \sum_{k=1}^{K} \left\| \mathbf{z}_k - \mathbf{H}_k \boldsymbol{\Phi} \right\|_2^2, \tag{3.33}
$$

where $J_L(\boldsymbol{\Phi})$ is the linearized objective function. The LS estimate $\hat{\boldsymbol{\Phi}}_{LS}$ is obtained by minimizing $J_L(\boldsymbol{\Phi})$, where $\boldsymbol{\Phi} \in \mathbb{R}^{d_{\boldsymbol{\Phi}}}$ and $d_{\boldsymbol{\Phi}} = \dim(\boldsymbol{\Phi}) = 8$. By defining the vector $\tilde{\mathbf{z}}$ and the matrix $\tilde{\mathbf{H}}$ as:

$$
\tilde{\mathbf{z}} = \left[\mathbf{z}_1^T \quad \ldots \quad \mathbf{z}_k^T \quad \ldots \quad \mathbf{z}_K^T \right]^T, \tag{3.34}
$$

$$\tilde{\mathbf{H}} = \begin{bmatrix} \mathbf{H}_1^T & \dots & \mathbf{H}_k^T & \dots & \mathbf{H}_K^T \end{bmatrix}^T, \tag{3.35}$$

we can express the LS estimate of $\mathbf{\Phi}_{LS}$ as:

$$\hat{\mathbf{\Phi}}_{LS} = (\tilde{\mathbf{H}}^T \tilde{\mathbf{H}})^{-1} \tilde{\mathbf{H}}^T \tilde{\mathbf{z}} = \tilde{\mathbf{H}}^{\#} \tilde{\mathbf{z}}, \tag{3.36}$$

where $\tilde{\mathbf{H}}^{\#}$ is the pseudo-inverse matrix of $\tilde{\mathbf{H}}$. We assume here that $\tilde{\mathbf{H}}$ is a full rank matrix.

3.6 The Expectation-Maximization (EM) algorithm

In this Section, we develop an algorithm based on the EM approach to solve the relative grid-locking problem. The algorithm jointly estimates all the registration errors (measurement, attitude and position errors) and is more efficient than the previously derived linear LS estimate. As discussed in the Introduction of this Chapter, to the authors' knowledge, no other works on this topic provide a non-linear estimator (as the EM algorithm) for the solution of the general grid-locking problem.

3.6.1 The Expectation-Maximization algorithm: a brief outline

The EM algorithm is an iterative procedure to compute the ML estimate in presence of "incomplete data" [12], [18]. Let $\mathcal{X}$ and X be a sample space and a set of observed data sampled from $\mathcal{X}$. It is possible to define a family of probability density functions (pdf) on the observed data set X parameterized by a parameter vector $\mathbf{\Phi} \in \Omega$ as $\{p_X(X;\mathbf{\Phi}) : \mathbf{\Phi} \in \Omega\}$. The ML estimate of the parameter vector $\mathbf{\Phi}$ is the vector $\hat{\mathbf{\Phi}}_{ML} \in \Omega$ that maximizes the so-called log-likelihood function (LLF) $L_X(\mathbf{\Phi}) = \ln p_X(X;\mathbf{\Phi})$.

In some practical estimation problems, we have not at our disposal the complete data set X for estimating the parameter vector, but we have only an "incomplete" observed data set. The term "incomplete data" in its general form implies the existence of two sample spaces $\mathcal{Y}$ and $\mathcal{X}$ and a surjective map f from $\mathcal{X}$ to $\mathcal{Y}$. The observed data set Y is a realization from $\mathcal{Y}$. The corresponding set X in $\mathcal{X}$ is not observed directly, but only

indirectly through *Y*. More specifically, it is assumed there exists a mapping $f: X \mapsto Y = f(X)$ and that *X* is known only to lie in $\mathcal{X}(Y)$, the subset of $\mathcal{X}$ is determined by the intrinsic equation $Y = f(X)$, where *Y* is the observed data set and *X* is the complete data set. By taking into account the family of pdf on the complete data *X*, we can derive the general expression of the pdf of the incomplete observed data *Y* parameterized by the parameter vector $\boldsymbol{\Phi} \in \Omega$ as

$$p_Y(Y;\boldsymbol{\Phi}) = \int_{\mathcal{X}(Y)} p_X(X;\boldsymbol{\Phi})\, dX \ , \tag{3.37}$$

where $\mathcal{X}(Y) = \{X : f(X) = Y\}$. Finally, an estimate of the parameter vector can be obtained by applying the ML algorithm to the pdf of the incomplete data $p_Y(Y;\boldsymbol{\Phi})$. However, this quantity is often intractable due to the mathematical difficulties involved in the closed-form evaluation of the integral on $p_X(X;\boldsymbol{\Phi})$.

The EM algorithm is useful in such cases when the pdf (parameterized by $\boldsymbol{\Phi}$) of the incomplete data set $p_Y(Y;\boldsymbol{\Phi})$ is not available in closed form while the closed-form expression of the pdf (parameterized by $\boldsymbol{\Phi}$) of the complete data set, i.e. $p_X(X;\boldsymbol{\Phi})$, is known. Then, the basic idea behind the EM algorithm is:

1. Estimate the LLF of the (unobservable) complete data set $L_X(\boldsymbol{\Phi}) = \ln p_X(X;\boldsymbol{\Phi})$, given the observed data set *Y* and the current estimate of the parameter vector $\hat{\boldsymbol{\Phi}}^n$:

$$L_X(\boldsymbol{\Phi}) = E\left\{\ln p_X(X;\boldsymbol{\Phi}) \middle| Y, \hat{\boldsymbol{\Phi}}^n\right\}. \tag{3.38}$$

2. Maximize the estimated LLF of the unobservable complete data set with respect to $\boldsymbol{\Phi}$ in order to obtain a new estimate of the parameter vector:

$$\hat{\boldsymbol{\Phi}}^{n+1} = \arg\max_{\boldsymbol{\Phi}\in\Omega} \left\{L_x(\boldsymbol{\Phi}); \hat{\boldsymbol{\Phi}}^n\right\}. \tag{3.39}$$

The previous two steps can be reformulated in the form of a constrained optimization problem:

1. **E-step**: compute the objective function:

$$Q\left(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}^n\right) = E\left\{\ln p_X(X;\boldsymbol{\Phi}) \middle| Y, \hat{\boldsymbol{\Phi}}^n\right\}. \tag{3.40}$$

2. **M-step**: choose $\hat{\boldsymbol{\Phi}}^{n+1}$ to be any value in Ω that maximizes $Q(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^{n})$, i.e.:

$$\hat{\boldsymbol{\Phi}}^{n+1} = \arg\max_{\boldsymbol{\Phi}\in\Omega} \left\{ Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^{n}\right) \right\}. \tag{3.41}$$

It can be proved (see e.g. [12], [18]) that, under some regularity assumptions, the iterations of the EM algorithm produce an estimate of $\boldsymbol{\Phi}$ that converges to the value that maximizes the LLF of the incomplete data, i.e. $L_Y(\boldsymbol{\Phi}) = \ln p_Y(Y;\boldsymbol{\Phi})$.

3.6.2 Application of the EM algorithm to the relative grid-locking problem

To apply the EM algorithm to the estimate of the parameter vector $\boldsymbol{\Phi}$, first we have to define the set of complete data *X* and the set of incomplete data *Y*. To this purpose, let us define three sets of *K* elements: $V_1 = \{\mathbf{v}_{1,m}^k\}_{k=1}^K$, $V_2 = \{\mathbf{v}_{2,m}^k\}_{k=1}^K$, and $S = \{\mathbf{s}_k\}_{k=1}^K$. The complete data set *X* is composed by all the measurements coming from radars #1 and #2 and all the target state vectors, i.e. $X = \{V_1 \quad V_2 \quad S\}$. The incomplete data set *Y* is composed only by the measurements coming from radars #1 and #2, i.e. $Y = \{V_1 \quad V_2\}$. In fact, the target state vector *S* is not directly observable, but it can be estimated through the radar measurements.

The EM algorithm consists on iterating the E-step and the M-step until convergence. To perform the E-step, we have to evaluate the objective function $Q(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^{n})$ defined in eq. (3.40). This requires first the evaluation of the pdf of the complete data *X*, $p_X(X;\boldsymbol{\Phi})$. Starting from eqs. (3.9), (3.12) and (3.15), we have:

$$\begin{aligned} p_X\left(X;\boldsymbol{\Phi}\right) &= p_{S,Y}\left(S,Y;\boldsymbol{\Phi}\right) \\ &= p_{V_1|S}\left(V_1|S;\boldsymbol{\Phi}\right) p_{V_2|S}\left(V_2|S;\boldsymbol{\Phi}\right) p_S\left(S\right) \\ &= p_{V_2|S}\left(V_2|S;\boldsymbol{\Phi}\right) p_{V_1|S}\left(V_1|S\right) p_S\left(S\right), \end{aligned} \tag{3.42}$$

where the 2nd equality derives from the fact that the measurement vectors are independent conditionally to the state vector *S*. The last equality follows from the fact that the

measurements of the radar #1 do not depend on the parameter vector $\mathbf{\Phi}$ to be estimated, since they have been assumed unbiased.

To evaluate the term $p_S(S)$ we have from eq. (3.15) that the joint pdf of the K target state vectors is given by the product:

$$p_S(S) = p_{\mathbf{s}_1}(\mathbf{s}_1)\prod_{k=2}^{K} p_{\mathbf{s}_k|\mathbf{s}_{k-1}}(\mathbf{s}_k|\mathbf{s}_{k-1}), \tag{3.43}$$

where $\mathbf{s}_1 \sim \mathcal{N}(\boldsymbol{\mu}_0, \mathbf{Q}_w)$ and $\mathbf{s}_k|\mathbf{s}_{k-1} \sim \mathcal{N}(\mathbf{F}\mathbf{s}_{k-1}, \mathbf{Q}_w)$. From eq. (3.13), $p_{V_1|S}(V_1|S)$ can be rewritten as:

$$p_{V_1|S}(V_1|S) = \prod_{k=1}^{K} p_{\mathbf{v}_{1,m}^k|\mathbf{s}_k}(\mathbf{v}_{1,m}^k|\mathbf{s}_k), \tag{3.44}$$

where $\mathbf{v}_{1,m}^k|\mathbf{s}_k \sim \mathcal{N}(\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k), \mathbf{C}_1)$ and $\tilde{\mathbf{h}}^{-1}(\cdot)$ is the Cartesian-to-spherical coordinate transformation of the first three components of the target state vector $\mathbf{s}_k$, i.e. $\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k) = \mathbf{h}^{-1}(\mathbf{r}_k)$ [17]. Finally, from eq. (3.14), the term $p_{V_2|S}(V_2|S;\mathbf{\Phi})$ can be evaluated as:

$$p_{V_2|S}(V_2|S;\mathbf{\Phi}) = \prod_{k=1}^{K} p_{\mathbf{v}_{2,m}^k|\mathbf{s}_k}(\mathbf{v}_{2,m}^k|\mathbf{s}_k;\mathbf{\Phi}), \tag{3.45}$$

where $\mathbf{v}_{2,m}^k|\mathbf{s}_k \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\mathbf{\Phi}), \mathbf{C}_2)$ and $\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\mathbf{\Phi}) = \boldsymbol{\mu}(\mathbf{r}_k,\mathbf{\Phi})$. Through eqs. (3.43), (3.44), and (3.45), the logarithm of the pdf of the complete data X can be evaluated, ignoring constants, as ([14], [12], [16]):

$$\begin{aligned}
\ln p_X(X;\mathbf{\Phi}) &= \ln p_{V_2|S}(V_2|S;\mathbf{\Phi}) + \ln p_{V_1|S}(V_1|S) + \ln p_S(S) \\
&= -\frac{1}{2}\sum_{k=1}^{K}\left[\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\mathbf{\Phi})\right]\mathbf{C}_2^{-1}\left[\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\mathbf{\Phi})\right]^T \\
&\quad -\frac{1}{2}\sum_{k=1}^{K}\left[\mathbf{v}_{1,m}^k - \tilde{\mathbf{h}}^{-1}(\mathbf{s}_k)\right]\mathbf{C}_1^{-1}\left[\mathbf{v}_{1,m}^k - \tilde{\mathbf{h}}^{-1}(\mathbf{s}_k)\right]^T \\
&\quad -\frac{1}{2}\sum_{k=2}^{K}\left[\mathbf{s}_k - \mathbf{F}\mathbf{s}_{k-1}\right]\mathbf{Q}_w^{-1}\left[\mathbf{s}_k - \mathbf{F}\mathbf{s}_{k-1}\right]^T + const.
\end{aligned} \tag{3.46}$$

It can be noted that only the first term in eq. (3.46) depends on $\mathbf{\Phi}$. Hence, for this reason all the other terms can be neglected. Using the property of the *trace* operator and by defining $\boldsymbol{\alpha}_k$ as:

$$\boldsymbol{\alpha}_k = \mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k, \boldsymbol{\Phi}\right), \tag{3.47}$$

eq. (3.46) can be rewritten as:

$$\ln p_X\left(X;\boldsymbol{\Phi}\right) = -\frac{1}{2}\mathrm{tr}\left\{\mathbf{C}_2^{-1}\sum_{k=1}^{K}\boldsymbol{\alpha}_k\boldsymbol{\alpha}_k^T\right\} + const. \tag{3.48}$$

We can now perform the E-step of the EM algorithm by evaluating the objective function $Q(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n)$, i.e. the conditional expectation of the LLF of the complete data, $\ln p_X(X;\boldsymbol{\Phi})$, given the incomplete data set *Y* and the current estimate of the parameter vector $\hat{\boldsymbol{\Phi}}^n$:

$$Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right) = -\frac{1}{2}\mathrm{tr}\left\{\mathbf{C}_2^{-1}\sum_{k=1}^{K}E\left\{\boldsymbol{\alpha}_k\boldsymbol{\alpha}_k^T\middle|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\right\}\right\}. \tag{3.49}$$

First, we define, using the classical notation, the conditional mean (CM) and the conditional error covariance matrix as:

$$\hat{\mathbf{s}}_{k|K}^n = E\left\{\mathbf{s}_k\middle|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\right\}, \tag{3.50}$$

$$\mathbf{P}_{k|K}^n = E\left\{(\mathbf{s}_k - \hat{\mathbf{s}}_{k|K}^n)(\mathbf{s}_k - \hat{\mathbf{s}}_{k|K}^n)^T\middle|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\right\}. \tag{3.51}$$

A recursive procedure to evaluate terms $\hat{\mathbf{s}}_{k|K}^n$ and $\mathbf{P}_{k|K}^n$ is provided in [33]. To evaluate explicitly the term $\boldsymbol{\alpha}_k\boldsymbol{\alpha}_k^T$, we assume that the function $\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi})$ is a differentiable function with respect to $\mathbf{s}_k$. Then, it can be approximated by the first-order Taylor series expansion as follows:

$$\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}\right) \cong \tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}\right) + \mathbf{M}_k^T\left(\boldsymbol{\Phi}\right)\left(\mathbf{s}_k - \hat{\mathbf{s}}_{k|K}^n\right), \tag{3.52}$$

where $\mathbf{M}_k(\boldsymbol{\Phi}) = \partial\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi})/\partial\mathbf{s}_k\Big|_{\mathbf{s}_k=\hat{\mathbf{s}}_{k|K}^n}$. Analytical calculation of matrix $\mathbf{M}_k(\boldsymbol{\Phi})$ is addressed in Appendix C. It can be noted that a similar approach was used in [14], [15], and [16]. Using the linearized expression in eq. (3.52), the term $\boldsymbol{\alpha}_k\boldsymbol{\alpha}_k^T$ can be explicitly evaluated as:

$$\begin{aligned}\boldsymbol{\alpha}_k\boldsymbol{\alpha}_k^T &= \left(\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}\right)\right)\left(\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}\right)\right)^T \\ &\cong \left(\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}\right) - \mathbf{M}_k^T\left(\boldsymbol{\Phi}\right)\left(\mathbf{s}_k - \hat{\mathbf{s}}_{k|K}^n\right)\right)\times \\ &\qquad \times\left(\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}\right) - \mathbf{M}_k^T\left(\boldsymbol{\Phi}\right)\left(\mathbf{s}_k - \hat{\mathbf{s}}_{k|K}^n\right)\right)^T \\ &= \mathbf{W}_k\left(\boldsymbol{\Phi}\right) - \mathbf{V}_k\left(\boldsymbol{\Phi}\right) - \mathbf{V}_k^T\left(\boldsymbol{\Phi}\right) + \mathbf{M}_k^T\left(\boldsymbol{\Phi}\right)\left(\mathbf{s}_k - \hat{\mathbf{s}}_{k|K}^n\right)\left(\mathbf{s}_k - \hat{\mathbf{s}}_{k|K}^n\right)^T \mathbf{M}_k\left(\boldsymbol{\Phi}\right),\end{aligned} \tag{3.53}$$

where:

$$\mathbf{W}_k\left(\boldsymbol{\Phi}\right) \triangleq \left(\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}\right)\right)\left(\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}\right)\right)^T, \tag{3.54}$$

$$\mathbf{V}_k\left(\boldsymbol{\Phi}\right) \triangleq \left(\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}\right)\right)\left(\mathbf{s}^k - \hat{\mathbf{s}}_{k|K}^n\right)^T \mathbf{M}_k\left(\boldsymbol{\Phi}\right). \tag{3.55}$$

The conditional expectation $E\{\boldsymbol{\alpha}_k(\boldsymbol{\alpha}_k)^T \,|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\}$ can be evaluated as:

$$E\left\{\boldsymbol{\alpha}_k(\boldsymbol{\alpha}_k)^T \,|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\right\} = \mathbf{W}_k\left(\boldsymbol{\Phi}\right) + \mathbf{M}_k^T\left(\boldsymbol{\Phi}\right)\mathbf{P}_{k|K}^n\mathbf{M}_k\left(\boldsymbol{\Phi}\right), \tag{3.56}$$

where we used the fact that $E\{\mathbf{V}_k(\boldsymbol{\Phi})|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\} = 0$. Finally, collecting previous results, we get:

$$Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right) = -\frac{1}{2}\mathrm{tr}\left\{\mathbf{C}_2^{-1}\sum_{k=1}^{K}\mathbf{W}_k\left(\boldsymbol{\Phi}\right) + \mathbf{M}_k^T\left(\boldsymbol{\Phi}\right)\mathbf{P}_{k|K}^n\mathbf{M}_k\left(\boldsymbol{\Phi}\right)\right\} + const. \tag{3.57}$$

For sake of clarity, we explicitly rewrite $Q(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n)$ as:

$$\begin{aligned}Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right) = &-\frac{1}{2}\mathrm{tr}\left\{\mathbf{C}_2^{-1}\sum_{k=1}^{K}\left(\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}\right)\right)\left(\mathbf{v}_{2,m}^k - \tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}\right)\right)^T\right\} \\ &-\frac{1}{2}\mathrm{tr}\left\{\mathbf{C}_2^{-1}\sum_{k=1}^{K}\mathbf{M}_k^T\left(\boldsymbol{\Phi}\right)\mathbf{P}_{k|K}^n\mathbf{M}_k\left(\boldsymbol{\Phi}\right)\right\},\end{aligned} \tag{3.58}$$

where $\hat{\mathbf{s}}_{k|K}^n$ and $\mathbf{P}_{k|K}^n$ are the smoothed state vector estimate and the smoothed error covariance matrix of eqs. (3.50) and (3.51), respectively.

The M-step can be now performed as:

$$\hat{\boldsymbol{\Phi}}^{n+1} = \arg\max_{\boldsymbol{\Phi}\in\Omega}\left\{Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right)\right\}. \tag{3.59}$$

A closed-form solution of the optimization problem in (3.59) is infeasible due to the analytical complexity of the objective function $Q(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n)$. The problem is solved numerically by resorting to a *gradient-based* algorithm. More specifically, the *Sequential Quadratic Programming* (SQP) method [34] [35] was used. It requires that the objective and constraint functions are both continuous and have continuous first derivatives. It is easy to shown that $Q(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n)$ satisfies both these conditions. The steps of the estimator of the parameter vector $\boldsymbol{\Phi}$ based on the EM algorithm are outlined in Table 3.1 and in Fig. 3.2.

Estimator of the parameter vector $\boldsymbol{\Phi}$ based on the EM algorithm

1) **Initialization**

The EM algorithm is initialized using the linear Least Squares Estimator (LSE):

$$\hat{\boldsymbol{\Phi}}^0 = \mathrm{LSE}\{V_1, V_2\}.$$

2) **Iterative algorithm**

while $|Q(\hat{\boldsymbol{\Phi}}^{n+1}) - Q(\hat{\boldsymbol{\Phi}}^n)| \geq \varepsilon$, where ε is the tolerance,

E-Step:

for k = 1 to K (where K is the maximum number of available observations)

estimate the k^{th} smoothed target state vector and the relative smoothed error covariance matrix using the K measurements coming from both radars #1 and #2, V_1 and V_2 and the current estimate of the vector parameters $\hat{\boldsymbol{\Phi}}^n$:

$$\hat{\mathbf{s}}_{k|K}^n = E\left\{\mathbf{s}^k \middle| V_1, V_2; \hat{\boldsymbol{\Phi}}^n\right\}, \quad \mathbf{P}_{k|K}^n = E\left\{(\mathbf{s}_k - \hat{\mathbf{s}}_{k|K}^n)(\mathbf{s}_k - \hat{\mathbf{s}}_{k|K}^n)^T \middle| V_1, V_2; \hat{\boldsymbol{\Phi}}^n\right\}.$$

end for

M-Step

Calculate numerically the maximum of $Q(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}^n)$ defined in eq. (3.58):

$$\hat{\boldsymbol{\Phi}}^{n+1} = \arg\max_{\boldsymbol{\Phi} \in \Omega} \left\{Q\left(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}^n\right)\right\}.$$

end while

Table 3.1 – Summary of the proposed EM algorithm.

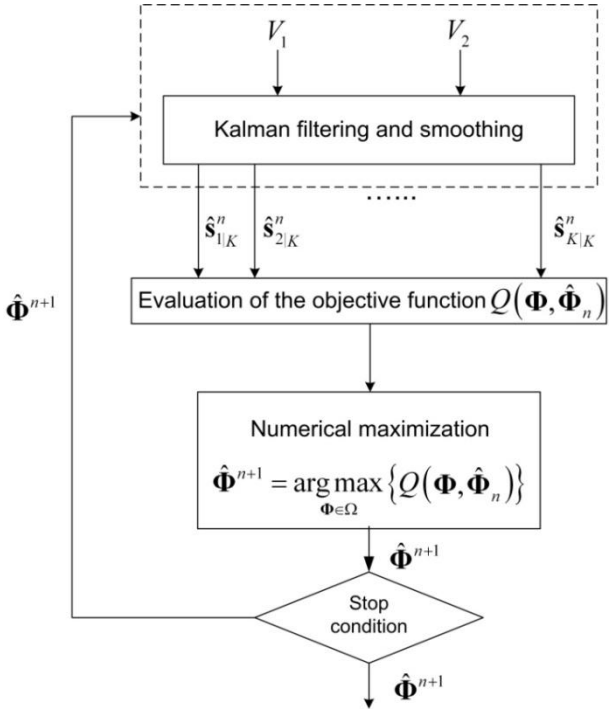

Figure 3.2 - Flow chart of the proposed EM algorithm for grid-locking in a netted radar system.

3.7 Performance bound

It is well-known that the Cramér-Rao lower bound (CRLB) provides a lower bound on the accuracy achievable by any unbiased estimator of the signal parameter vector $\mathbf{\Phi}$ using a set of measurements $\mathbf{z}$ [28]. However, calculation of the CRLB can be infeasible in some cases where the vector $\mathbf{\Psi}$ of unknown parameters is composed of a deterministic vector $\mathbf{\Phi}$ and of a random vector $\mathbf{w}$ of nuisance parameters, because of the difficulty of calculating the marginal pdf $p_{\mathbf{z}}(\mathbf{z};\mathbf{\Phi})$, where dependency on $\mathbf{w}$ has been averaged out. In these cases, the hybrid CRLB (HCRLB) may be useful, even if it is generally less tight than the CRLB [19], [20], [21]. The HCRLB is obtained by using the joint pdf $p_{\mathbf{z},\mathbf{w}}(\mathbf{z},\mathbf{w};\mathbf{\Phi})$. The hybrid FIM (HFIM) is defined as:

$$\left[\mathbf{I}_H(\mathbf{\Phi})\right]_{ij} = -E_{\mathbf{z},\mathbf{w}}\left\{\frac{\partial^2}{\partial\Psi_i\partial\Psi_j}\ln\left(p_{\mathbf{z},\mathbf{w}}(\mathbf{z},\mathbf{w};\mathbf{\Phi})\right)\right\}, \tag{3.60}$$

where $\boldsymbol{\Psi} = [\boldsymbol{\Phi}^T \quad \mathbf{w}^T]^T$ is the hybrid parameter vector. The HCRLB is defined as the inverse of the HFIM, i.e. $\text{HCRLB}(\Psi_i) = [\mathbf{I}_H^{-1}(\boldsymbol{\Phi})]_{ii}$ and inequality $E\left\{(\hat{\boldsymbol{\Psi}} - \boldsymbol{\Psi})^2\right\} \geq \mathbf{I}_H^{-1}(\boldsymbol{\Phi})$ holds true for any wide-sense unbiased estimator $\hat{\boldsymbol{\Psi}}$.[7]

The relative grid-locking problem can be seen as an hybrid estimation problem where the observations are the measurements from radar #1 and radar #2, V_1 and V_2, respectively; while the hybrid parameter vector is $\boldsymbol{\Psi} = [\boldsymbol{\Phi}^T \quad \mathbf{s}_1^T \quad \ldots \quad \mathbf{s}_K^T]^T$, where $\boldsymbol{\Phi}$ is the deterministic parameter vector of the relative grid-locking errors and $\{\mathbf{s}_1^T, \ldots, \mathbf{s}_K^T\}$ are the random target state vectors for the K observations. The HFIM for this estimation problem can be expressed as:

$$\begin{aligned}\left[\mathbf{I}_H(\boldsymbol{\Phi})\right]_{ij} &= -E_{V_1,V_2,S}\left\{\frac{\partial^2 \ln p(V_1, V_2, S; \boldsymbol{\Phi})}{\partial \Psi_i \partial \Psi_j}\right\} \\ &= -E_{V_1,V_2,S}\left\{\frac{\partial^2 \ln\left(p(V_1|S)\, p(V_2|S;\boldsymbol{\Phi})\, p(S)\right)}{\partial \Psi_i \partial \Psi_j}\right\}.\end{aligned} \tag{3.61}$$

From eq. (3.44), the logarithm of the conditional pdf of the measurements of radar #1 can be expressed as:

$$\ln p(V_1|S) = \sum_{k=1}^{K} \ln p\left(\mathbf{v}_{1,m}^k | \mathbf{s}_k\right), \tag{3.62}$$

where $\mathbf{v}_{1,m}^k | \mathbf{s}_k \sim \mathcal{N}(\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k), \mathbf{C}_1)$. Similarly, from eq. (3.45), the logarithm of the conditional pdf of the measurements of radar #2 is:

$$\ln p(V_2|S;\boldsymbol{\Phi}) = \sum_{k=1}^{K} \ln p\left(\mathbf{v}_{2,m}^k | \mathbf{s}_k; \boldsymbol{\Phi}\right), \tag{3.63}$$

where $\mathbf{v}_{2,m}^k | \mathbf{s}_k \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k, \boldsymbol{\Phi}), \mathbf{C}_2)$. Finally, from eq. (3.43), the logarithm of the joint pdf of all the target state vectors is:

---

[7] A wide-sense unbiased estimator $\hat{\boldsymbol{\Psi}}$ of the hybrid parameter vector $\boldsymbol{\Psi} = [\boldsymbol{\Phi}^T \quad \mathbf{w}^T]^T$, where $\mathbf{w}$ is the random vector of nuisance parameters, must satisfy $E_{\mathbf{z},\mathbf{w}}\left\{\hat{\boldsymbol{\Psi}} - \boldsymbol{\Psi}\right\} = \mathbf{0}$. A strict-sense unbiased estimator $\hat{\boldsymbol{\Psi}}$ must satisfy $E_{\mathbf{z}}\left\{(\hat{\boldsymbol{\Psi}} - \boldsymbol{\Psi})|\mathbf{w}\right\} = \mathbf{0}$ for any realization of $\mathbf{w}$.

$$\ln p(S) = \ln p(\mathbf{s}_1) + \sum_{k=2}^{K} \ln p(\mathbf{s}_k | \mathbf{s}_{k-1}), \tag{3.64}$$

where $\mathbf{s}_1 \sim \mathcal{N}(\boldsymbol{\mu}_0, \mathbf{Q}_w)$ and $\mathbf{s}_k | \mathbf{s}_{k-1} \sim \mathcal{N}(\mathbf{F}\mathbf{s}_{k-1}, \mathbf{Q}_w)$.

It can be noted that the logarithm of the joint pdf of the target state vectors can always be calculated with the assumed model of eq. (3), but it is not always possible in general. More precisely, ln$p(S)$ cannot be evaluated for all the joint pdfs that are compact support distributions. For example, in [17], where a uniform distribution in the volume under search was adopted for the target position vectors, ln$p(S)$ could not be obtained and a different approach was adopted to derive a performance bound.

Using the eqs. (3.62), (3.63), and (3.64), the HFIM can be rewritten as:

$$\left[\mathbf{I}_H(\boldsymbol{\Phi})\right]_{ij} = E_S\left\{\mathbf{I}_{H|S}(\boldsymbol{\Psi})\right\}, \tag{3.65}$$

where the conditional HFIM is defined as:

$$\begin{aligned}\left[\mathbf{I}_{H|S}(\boldsymbol{\Psi})\right]_{ij} &= -E_{V_1,V_2|S}\left\{\frac{\partial^2 \ln p(V_1, V_2, S; \boldsymbol{\Phi})}{\partial \Psi_i \partial \Psi_j}\right\} \\ &= \sum_{k=1}^{K} l_{ij}(\mathbf{s}_k) + \sum_{k=1}^{K} g_{ij}(\mathbf{s}_k; \boldsymbol{\Phi}) + u_{ij}(\mathbf{s}_k),\end{aligned} \tag{3.66}$$

where:

$$l_{ij}(\mathbf{s}_k) \triangleq -E_{\mathbf{v}_{1,m}^k|\mathbf{s}_k}\left\{\frac{\partial^2 \ln p(\mathbf{v}_{1,m}^k | \mathbf{s}_k)}{\partial \Psi_i \partial \Psi_j}\right\}, \tag{3.67}$$

$$g_{ij}(\mathbf{s}_k; \boldsymbol{\Phi}) \triangleq -E_{\mathbf{v}_{2,m}^k|\mathbf{s}_k}\left\{\frac{\partial^2 \ln p(\mathbf{v}_{2,m}^k | \mathbf{s}_k; \boldsymbol{\Phi})}{\partial \Psi_i \partial \Psi_j}\right\}, \tag{3.68}$$

$$u_{ij}(\mathbf{s}_k) \triangleq -\frac{\partial^2 \ln p(\mathbf{s}_1)}{\partial \Psi_i \partial \Psi_j} - \sum_{k=2}^{K} \frac{\partial^2 \ln p(\mathbf{s}_k | \mathbf{s}_{k-1})}{\partial \Psi_i \partial \Psi_j}. \tag{3.69}$$

It is possible to show through direct calculation of eq. (3.69) that the function $u_{ij}(\mathbf{s}_k)$ represents the entries of a block matrix $\mathbf{U}$ that can be expressed as:

$$\mathbf{U}=\begin{pmatrix}\mathbf{0}_{d_\Phi\times d_\Phi} & \cdots & \cdots & \cdots & \mathbf{0}_{d_\Phi\times d_\Phi}\\ \vdots & \boldsymbol{\Omega} & \boldsymbol{\Lambda} & \cdots & \mathbf{0}_{6\times 6}\\ \vdots & \boldsymbol{\Lambda}^T & \boldsymbol{\Omega} & \ddots & \vdots\\ \vdots & \vdots & \ddots & \ddots & \boldsymbol{\Lambda}\\ \mathbf{0}_{d_\Phi\times d_\Phi} & \mathbf{0}_{6\times 6} & \cdots & \boldsymbol{\Lambda}^T & \mathbf{Q}_w^{-1}\end{pmatrix}, \tag{3.70}$$

where:

$$\boldsymbol{\Lambda}=-\mathbf{F}^T\mathbf{Q}_w^{-1}, \tag{3.71}$$

$$\boldsymbol{\Omega}=\mathbf{Q}_w^{-1}+\mathbf{F}^T\mathbf{Q}_w^{-1}\mathbf{F}. \tag{3.72}$$

The term $l_{ij}(\mathbf{s}_k)$ and $g_{ij}(\mathbf{s}_k;\boldsymbol{\Phi})$ can be evaluated as [28]:

$$l_{ij}\left(\mathbf{s}_k\right)=\left(\frac{\partial\tilde{\mathbf{h}}^{-1}\left(\mathbf{s}_k\right)}{\partial\Psi_i}\right)^T\mathbf{C}_1^{-1}\left(\frac{\partial\tilde{\mathbf{h}}^{-1}\left(\mathbf{s}_k\right)}{\partial\Psi_j}\right), \tag{3.73}$$

$$g_{ij}\left(\mathbf{s}_k;\boldsymbol{\Phi}\right)=\left(\frac{\partial\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}\right)}{\partial\Psi_i}\right)^T\mathbf{C}_2^{-1}\left(\frac{\partial\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}\right)}{\partial\Psi_j}\right). \tag{3.74}$$

The details of the calculation of these terms are reported in Appendix D. Matrix $[\mathbf{L}(\mathbf{s}_k)]_{ij}=l_{ij}(\mathbf{s}_k)$ can be rewritten in a block matrix form as:

$$\mathbf{L}\left(\mathbf{s}_k\right)=\begin{pmatrix}\mathbf{0}_{d_\Phi\times d_\Phi} & \mathbf{0}_{d_\Phi\times 6} & \cdots & \mathbf{0}_{d_\Phi\times 6}\\ \mathbf{0}_{6\times d_\Phi} & \cdots & \cdots & \cdots\\ \vdots & \cdots & \mathbf{N}_k & \cdots\\ \mathbf{0}_{6\times d_\Phi} & \cdots & \cdots & \mathbf{0}_{6\times 6}\end{pmatrix} \tag{3.75}$$

where the matrix $\mathbf{N}_k$ is given by:

$$\left[\mathbf{N}_k\right]_{ij}=l_{ij}\left(\mathbf{s}_k\right),\quad i,j=d_\Phi+(6k-5),\ldots,d_\Phi+6k. \tag{3.76}$$

Similarly, matrix $[\mathbf{G}(\mathbf{s}_k;\boldsymbol{\Phi})]_{ij}=g_{ij}(\mathbf{s}_k;\boldsymbol{\Phi})$ can be expressed as:

$$\mathbf{G}\left(\mathbf{s}_k;\boldsymbol{\Phi}\right)=\begin{pmatrix}\mathbf{F}_k & \mathbf{0}_{d_\Phi\times 6(k-1)} & \mathbf{B}_k & \mathbf{0}_{d_\Phi\times 6(K-k)}\\ \mathbf{0}_{6(k-1)\times d_\Phi} & \cdots & \cdots & \cdots\\ \mathbf{B}_k^T & \cdots & \mathbf{M}_k & \cdots\\ \mathbf{0}_{6(K-k)\times d_\Phi} & \cdots & \cdots & \mathbf{0}_{6(K-k)\times 6(K-k)}\end{pmatrix}, \tag{3.77}$$

where matrices $\mathbf{F}_k$, $\mathbf{B}_k$ and $\mathbf{M}_k$ are given by:

$$\left[\mathbf{F}_k\right]_{ij} = g_{ij}\left(\mathbf{s}_k;\boldsymbol{\Phi}\right), \quad i,j = 1,\ldots,d_{\boldsymbol{\Phi}}, \tag{3.78}$$

$$\left[\mathbf{B}_k\right]_{ij} = g_{ij}\left(\mathbf{s}_k;\boldsymbol{\Phi}\right), \quad i = 1,\ldots,d_{\boldsymbol{\Phi}};\ j = d_{\boldsymbol{\Phi}} + (6k-5),\ldots,d_{\boldsymbol{\Phi}} + 6k, \tag{3.79}$$

$$\left[\mathbf{M}_k\right]_{ij} = g_{ij}\left(\mathbf{s}_k;\boldsymbol{\Phi}\right), \quad i,j = d_{\boldsymbol{\Phi}} + (6k-5),\ldots,d_{\boldsymbol{\Phi}} + 6k. \tag{3.80}$$

By collecting the previous results, the conditional HFIM of eq. (3.66) can be expressed in block matrix form as:

$$\mathbf{I}_{H|S}\left(\boldsymbol{\Psi}\right) = \begin{pmatrix} \sum_{k=1}^{K}\mathbf{F}_k & \mathbf{B}_1 & \mathbf{B}_2 & \cdots & \mathbf{B}_K \\ \mathbf{B}_1^T & \boldsymbol{\Pi}_1 & \boldsymbol{\Lambda} & \cdots & \mathbf{0}_{6\times 6} \\ \mathbf{B}_2^T & \boldsymbol{\Lambda}^T & \boldsymbol{\Pi}_2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \boldsymbol{\Lambda} \\ \mathbf{B}_K^T & \mathbf{0}_{6\times 6} & \cdots & \boldsymbol{\Lambda}^T & \boldsymbol{\Xi} \end{pmatrix}, \tag{3.81}$$

where:

$$\boldsymbol{\Pi}_k = \mathbf{N}_k + \mathbf{M}_k + \boldsymbol{\Omega}, \quad i = 1,2,\ldots,K-1, \tag{3.82}$$

$$\boldsymbol{\Xi} = \mathbf{N}_K + \mathbf{M}_K + \mathbf{Q}_w^{-1}. \tag{3.83}$$

The mean value with respect to *S* in eq. (3.65) is evaluated through independent Monte Carlo trials. In particular, in our numerical analysis we use 100 Monte Carlo trials.

3.8 Numerical analysis

In this Section, we evaluate the performance of the proposed EM and LS algorithms for a single target scenario. The multi-target scenario is analyzed in the next Section. The comparison is carried out in terms of root mean square error (RMSE) as function of the standard deviation (std) of the measurement noise of radars #1 and #2, and of the number of observations *K*. Moreover, the RMSE is compared to the HCRLB to assess the efficiency of the proposed algorithm. Tracking performance are assessed by comparing the std of the smoothing filter output to the HCRLB. Radar #1 is located in the centre of the absolute reference system, while radar #2 is located in $\mathbf{t}_t=(2,2,2)\cdot 10^3$ m, so the distance between the

two radars is about 6.4641 km (see Fig. 2). The target is supposed to follow the discrete target state model in eq. (3.15) with an initial target state vector given by $\mathbf{s}_0=[\mathbf{r}_0{}^T\ \mathbf{v}_0{}^T]^T$ where the initial position vector is chosen to be $\mathbf{r}_0=[1000\ 1000\ 0]^T$ m while the initial velocity vector is $\mathbf{v}_0=[7.63\ 0\ 0]^T$ m/s. The power spectral density (PSD) for the three components of the noise vector is equal to $N_{0,x}=N_{0,y}=N_{0,z}=q=0.01$ m^2/s^3, where q is a introduced in eq. (3.17). In all the simulations the following parameters have been used:

- The actual bias errors values are: *i*) measurement biases: $d\rho$=-10m, $d\theta=d\varepsilon$=-0.0573°; *ii*) attitude biases: $d\chi=d\psi$=-0.0573° and $d\xi$=-0.1146°; *iii*) position biases: $dt_x=dt_y=dt_z$=-30 m.
- Scan time of both radars: T=1 sec.
- The tolerance is $\delta=10^{-6}$, i.e. absolute value of the difference between the objective function evaluated at the current estimate of the parameter vector and the objective function evaluated at the previous estimate, $|Q(\hat{\mathbf{\Phi}}^{n+1})-Q(\hat{\mathbf{\Phi}}^{n})|\leq\delta$.
- The number of independent Monte Carlo trials is: #MC=50.
- Searching volume Ω for the optimization algorithm, i.e. the region of $\mathbb{R}^{d_{\mathbf{\Phi}}}$ which the solution belongs to. We describe this volume whit the following set of inequalities: $\mathbf{a}\leq\mathbf{\Phi}\leq-\mathbf{a}$,where.**a**=(-100m, -5.7296°, -5.7296°,-5.7296°,-5.7296°, -100m, -100m, -100m).

The scenario under investigation is shown in Fig. 3.3 where the target track is a realization of the discrete state model in eq. (3.15).

In Figs. 3.4-3.11 the RMSE of the proposed EM algorithm and of the LS algorithm is plotted as a function of the radar measurements accuracy. More precisely, by defining with $\sigma_{\rho,i}$, $\sigma_{\theta,i}$ and $\sigma_{\varepsilon,i}$, i=1,2 the accuracies of the radars #1 and #2, we assume that: *i*) $\sigma_{\rho,1}=\sigma_{\rho,2}=\sigma_\rho$ and *ii*) $\sigma_{\theta,i}=\sigma_{\varepsilon,i}=\sigma_\theta$, i=1,2. The values of σ_ρ and σ_θ are supposed to vary between 1 m and 100 m and between 0.05° and 0.55°, respectively. The number of observations is K = 200. The HCRLB is also plotted to evaluate the efficiency of the proposed estimator.

Figs. 3.12-3.27 show the Error mean value and RMSE of the EM algorithm and of the LS algorithm, and the HCRLB as a function of the discrete time k, with $1\leq k\leq K$, K being the number of measurements of radars #1 and #2. In these simulations, radars #1 and #2 are characterized by the following accuracies: i) $\sigma_{\rho,1}=\sigma_{\rho,2}$=50 m; ii) $\sigma_{\theta,\mathrm{i}}=\sigma_{\varepsilon,\mathrm{i}}$=0.3°, i=1,2.

The number of available observations varies from 200 to 400. Such range has been chosen in order to avoid the so-called "catastrophic errors", due to the *threshold effect* (see e.g. [p. 170, 28]). The catastrophic errors occur for all the non-linear estimators and depend on both the number of observations and the level of the noise power (or equivalently on the signal-to-noise power ratio). In our case, if the radar accuracy is too low or if K is too small, the estimates of $\hat{\mathbf{s}}_{k|K}^{n}$ and $\mathbf{P}_{k|K}^{n}$ (needed to calculate the objective function in eq. (3.58)) could not be sufficiently accurate. Such low estimation accuracy could cause a wrong progress of the objective function and consequently a wrong localization of the maximum.

In Figs. 3.28 and 3.29 the accuracy on the estimate of the x component of the target position and velocity vectors is shown for radar accuracies σ_ρ=50 m, σ_θ=σ_ε=0.3° and for K=400. The estimate of the target state is carried out by using the smoothing algorithm described in [33].

From the numerical results we derived, most of which are not reported here for brevity, the following considerations can be drawn:

- The EM algorithm generally outperforms the LS algorithm, mainly for low radar accuracy. For the estimate of some parameters, e.g. the pitch error and the x component of the location error (see Figs. 3.7 and 3.9), the two algorithms have pretty close performance.
- Comparing the RMSE of the EM and LS algorithms as a function of the number K of observations, it can be noted that, generally, the EM algorithm outperforms the LS, even if for some set of parameter values (all the attitude errors and the x and y components of the location error) their performance is pretty close.
- To avoid catastrophic errors the EM algorithm requires an enough large number of observations. Through simulations, it is possible to infer that the minimum number is about K=200 observations (for radar accuracies of σ_ρ=50 m, σ_θ=σ_ε=0.3°) .
- The error mean value of the EM estimates decreases by increasing the number of iterations.
- The non-linear EM estimate of the deterministic parameter vector $\mathbf{\Phi}$ is generally not tight to the HCRLB, whereas, the EM estimate of the random target state

vector is generally tight to the HCRLB. However, it must be noted that, in general, also the HCRLB for the deterministic parameter vector estimate is not tight [19].

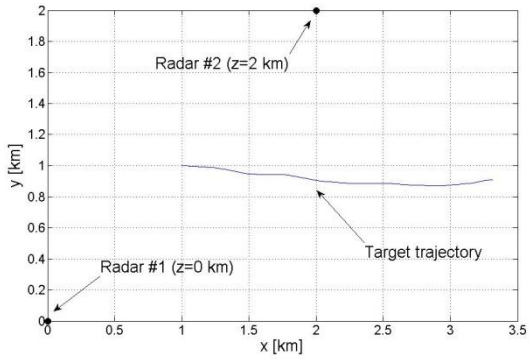

Figure 3.3 – Geometry of the single target scenario.

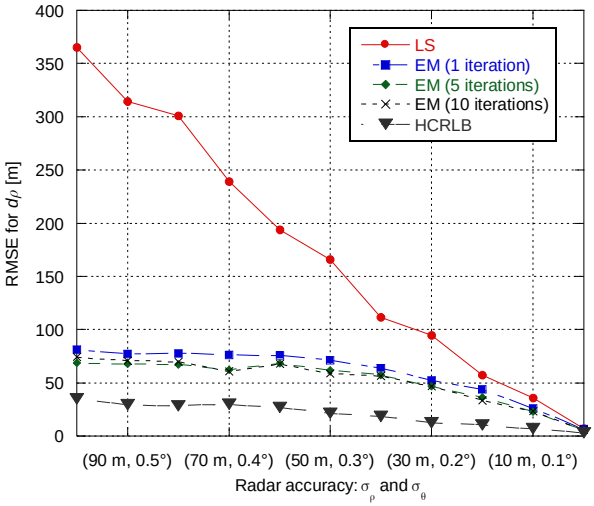

Figure 3.4 – RMSE for the estimate of the range error as function of the noise std.

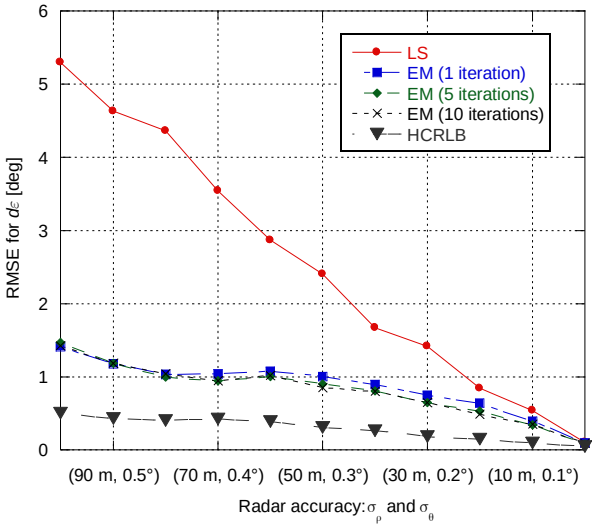

Figure 3.5 – RMSE for the estimate of the elevation error as function of the noise std.

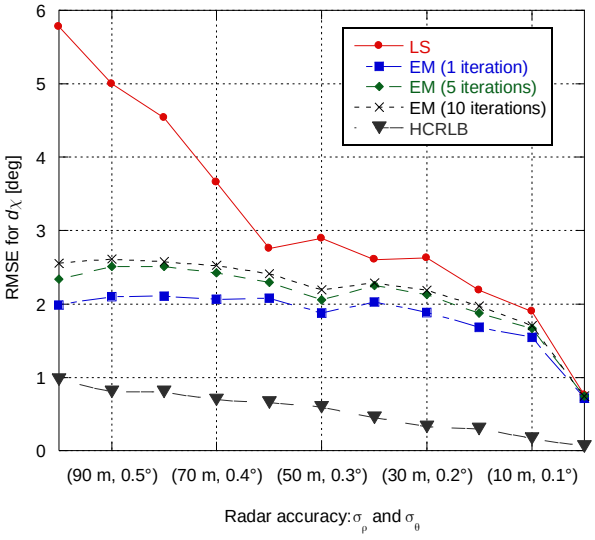

Figure 3.6 – RMSE of the roll bias error as function of the noise std.

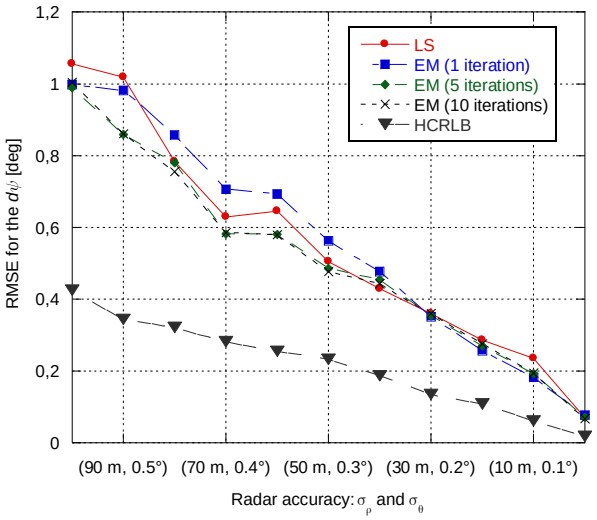

Figure 3.7 – RMSE of the pitch error as function of the noise std.

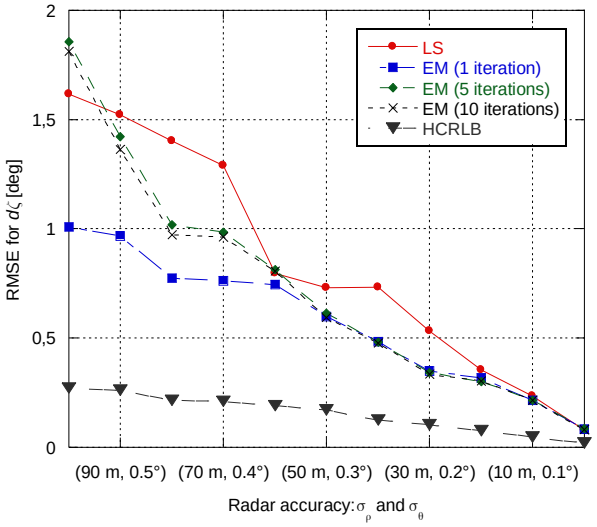

Figure 3.8 – RMSE of the yaw + azimuth error as function of the noise std.

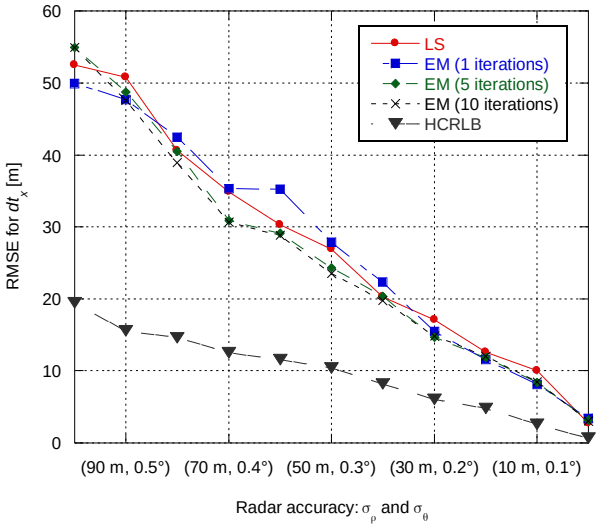

Figure 3.9 – RMSE for the estimate of the x component of the position bias error as function of the noise std.

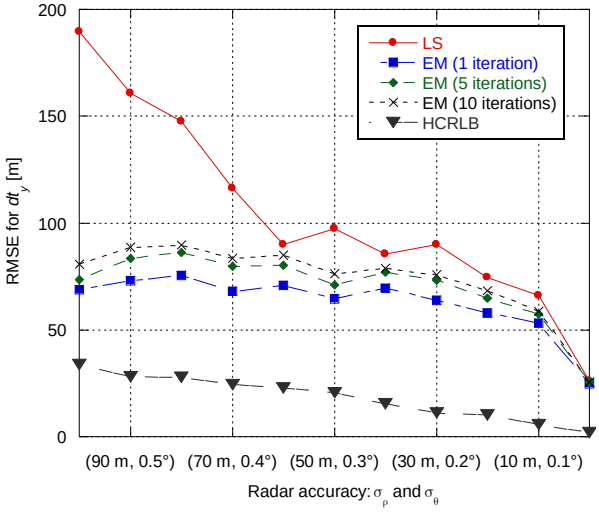

Figure 3.10 – RMSE for the estimate of the x component of the position bias error as function of the noise std.

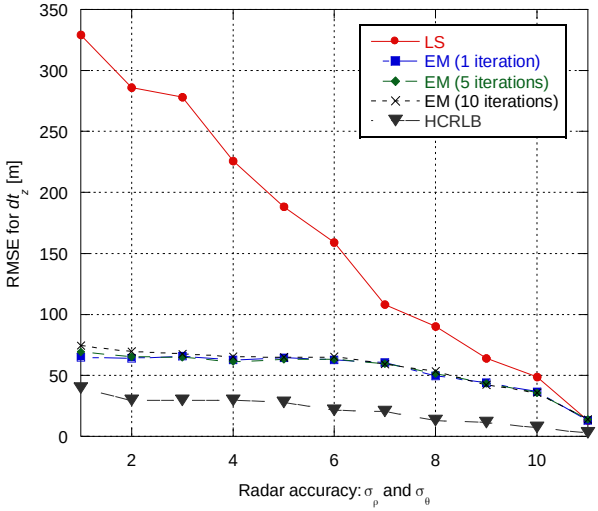

Figure 3.11 – RMSE for the estimate of the x component of the position bias error as function of the noise std.

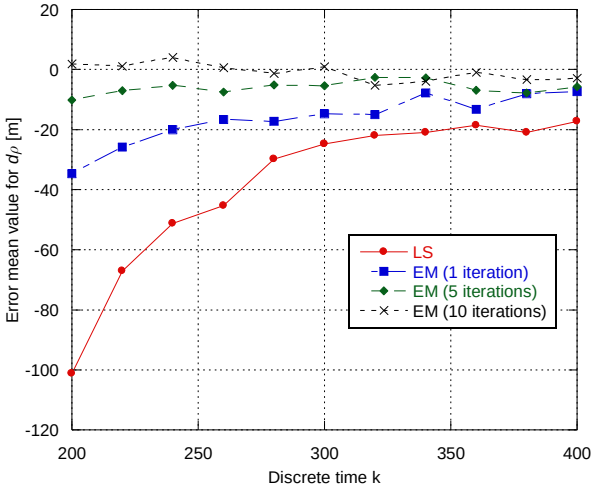

Figure 3.12 – Error mean value for the estimate of the range error.

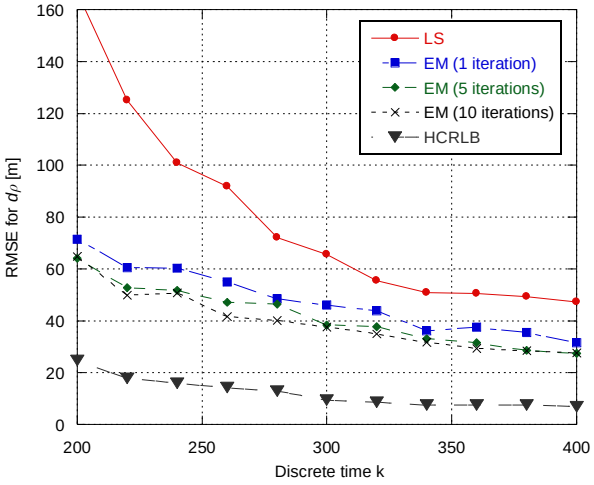

Figure 3.13 – RMSE for the estimate of the range error.

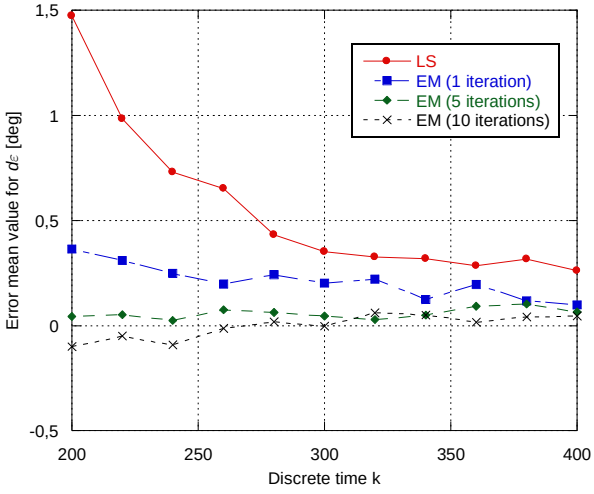

Figure 3.14 – Error mean value for the estimate of the elevation error.

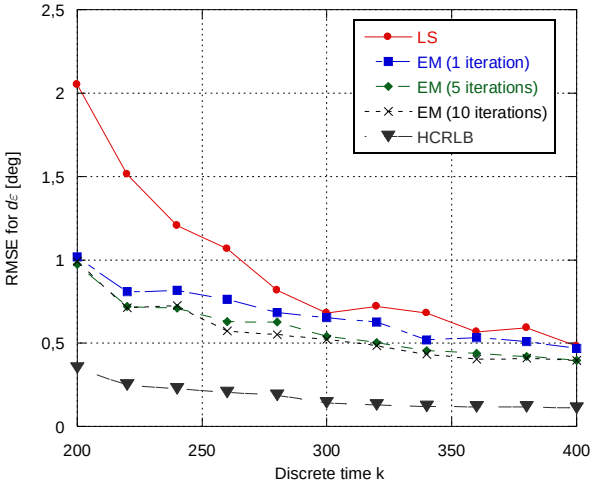

Figure 3.15 – RMSE for the estimate of the elevation error.

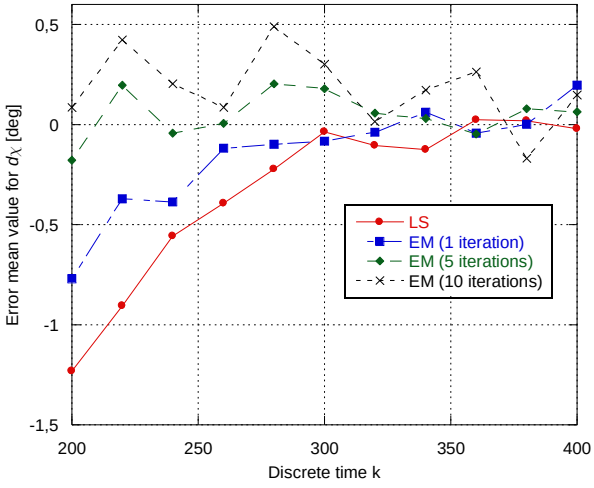

Figure 3.16 – Error mean value of the roll bias error.

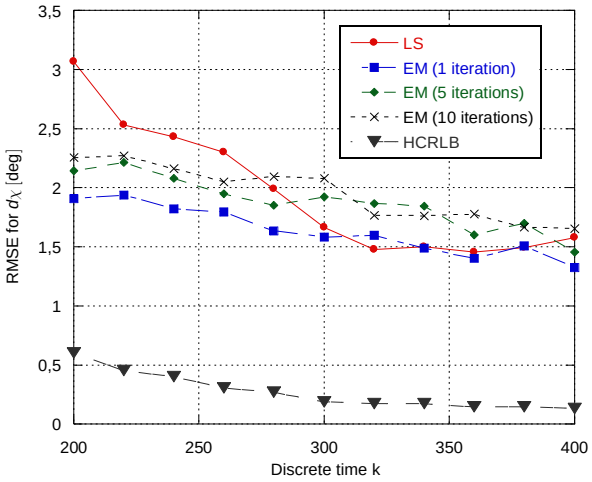

Figure 3.17 – RMSE of the roll bias error.

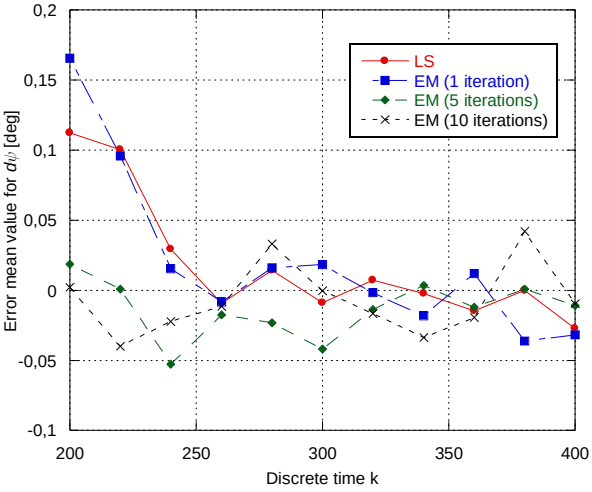

Figure 3.18 – Error mean value of the pitch error.

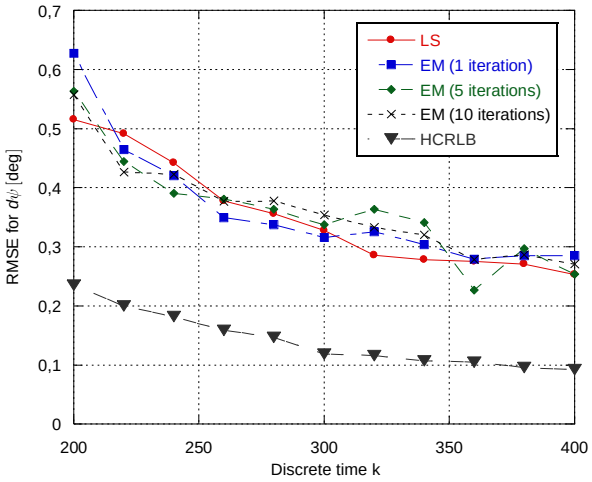

Figure 3.19 – RMSE of the pitch error.

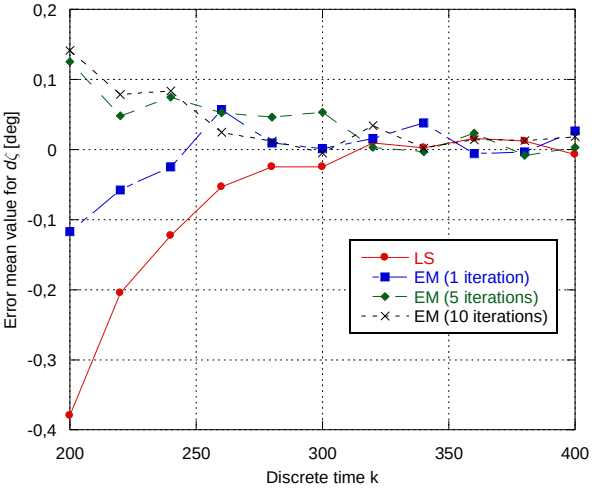

Figure 3.20 – Error mean value of the yaw+azimuth error.

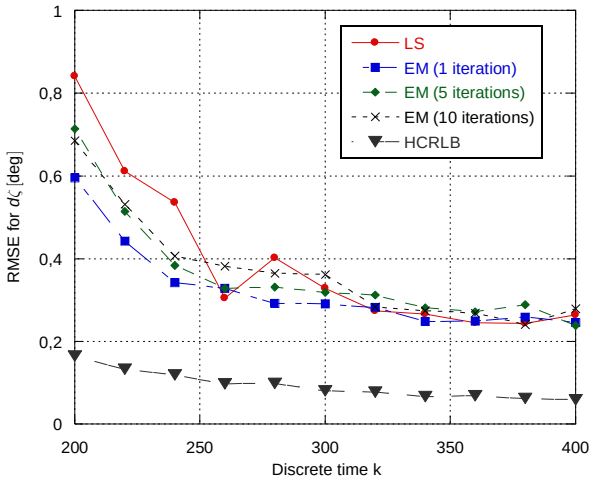

Figure 3.21 – RMSE of the yaw+azimuth error.

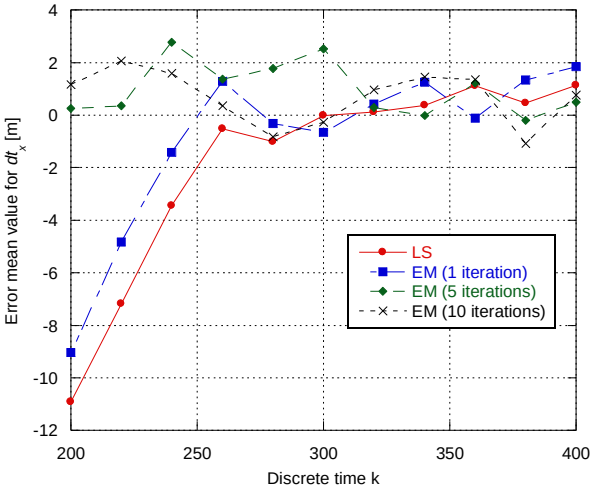

Figure 3.22 – Error mean value for the estimate of the x component of the position bias error.

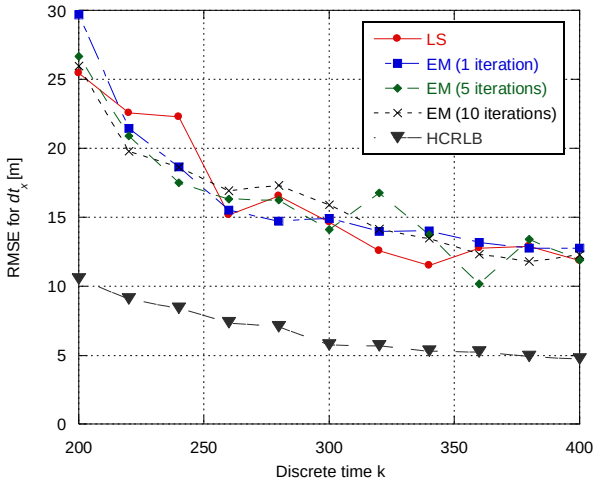

Figure 3.23 – RMSE for the estimate of the x component of the position bias error.

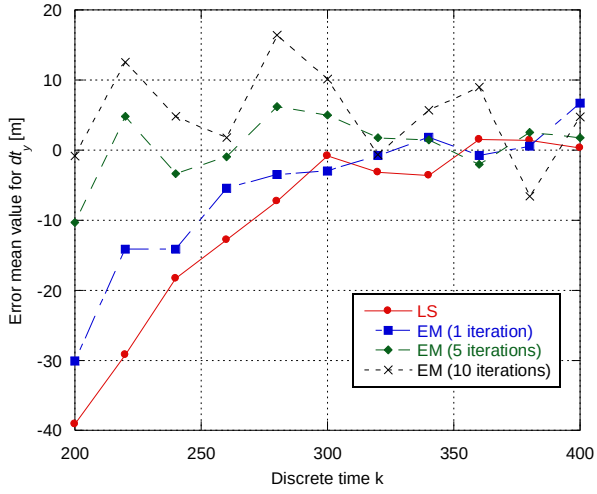

Figure 3.24 – Error mean value for the estimate of the x component of the position bias error.

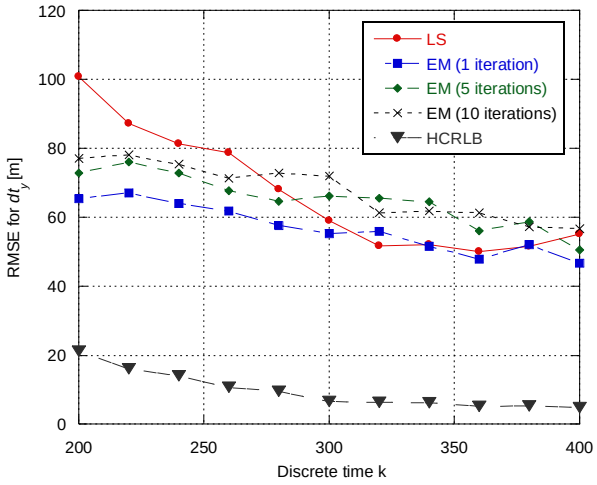

Figure 3.25 – RMSE for the estimate of the x component of the position bias error.

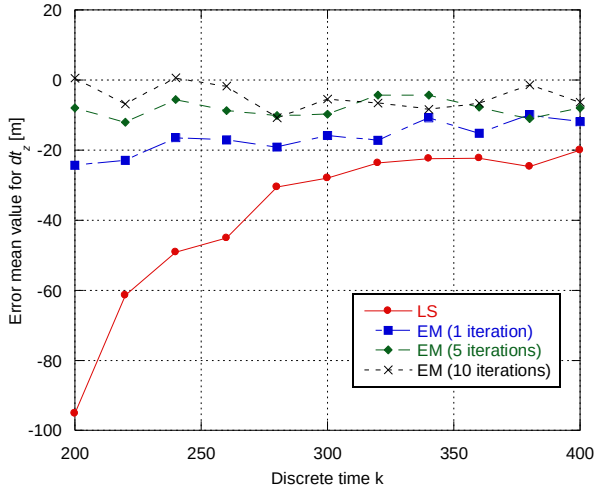

Figure 3.26 – Error mean value for the estimate of the x component of the position bias error.

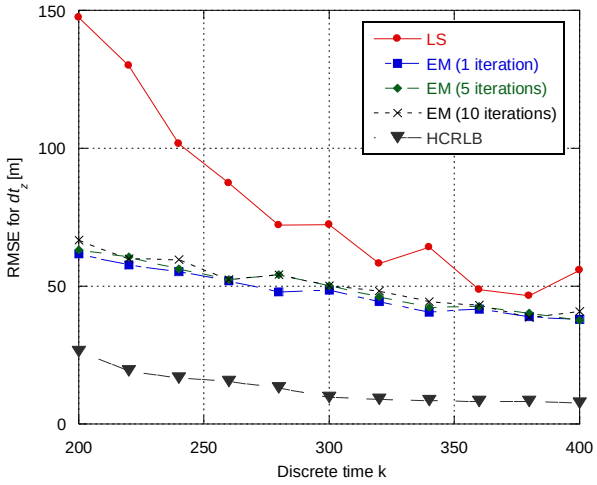

Figure 3.27 – RMSE for the estimate of the x component of the position bias error.

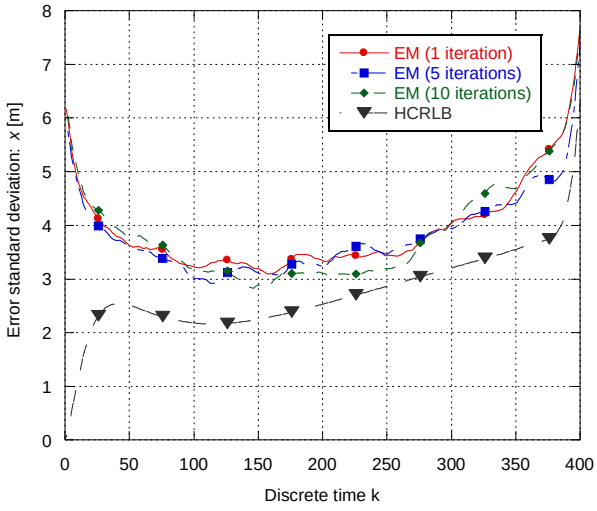

Figure 3.28 – Smoothing performance for the estimate of the *x* component of the target position vector: error std.

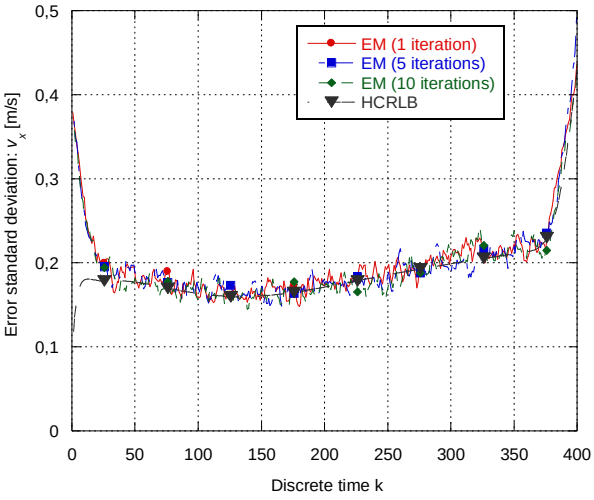

Figure 3.29 – Smoothing performance for the estimate of the *x* component of the target velocity vector: error std.

3.9 Generalization to the multi-target scenario

In this section, we generalize the proposed EM algorithm to a multi-target scenario. In the following, we assume to have N_t targets in the considered surveillance area [36]. Moreover, we assume that the data association process has been done without errors, i.e. we are able to associate exactly each radar measurement to the corresponding target. Under these assumptions, the set of complete data $\tilde{X}$ and the set of incomplete data $\tilde{Y}$ can be defined as:

$$\tilde{X} = \left(\tilde{V}_1 \quad \tilde{V}_2 \quad \tilde{S}\right), \quad \tilde{Y} = \left(\tilde{V}_1 \quad \tilde{V}_2\right), \tag{3.84}$$

where:

$$\tilde{V}_1 = \left\{V_1^l\right\}_{l=1}^{N_t}, \quad \tilde{V}_2 = \left\{V_2^l\right\}_{l=1}^{N_t}, \quad \tilde{S} = \left\{S^l\right\}_{l=1}^{N_t}, \tag{3.85}$$

and

$$V_1^l = \left\{\mathbf{v}_{1,m}^{k,l}\right\}_{k=1}^{K}, \quad V_2^l = \left\{\mathbf{v}_{2,m}^{k,l}\right\}_{k=1}^{K}, \quad S^l = \left\{\mathbf{s}_k^l\right\}_{k=1}^{K}, \tag{3.86}$$

This means that, for every scan time kT, k=1,...,K, we have at our disposal N_t measurements coming from radar #1 and radar #2, generated by N_t different targets. Each target is assumed to follow the linear stochastic model given in eq. (3.15):

$$\mathbf{s}_{k+1}^{l} = \mathbf{F}\mathbf{s}_{k}^{l} + \mathbf{w}_{k}^{l}, \tag{3.87}$$

where, as before, the index l defines the particular target. It is important to note that the discrete process noise vector $\mathbf{w}_k{}^l$ as function of the index l is an independent, zero-mean, Gaussian distributed, random vector. To apply the EM algorithm to this multi-target scenario, first we have to calculate the joint pdf of the complete data $\tilde{X}$, $p\left(\tilde{X};\mathbf{\Phi}\right)$. Following the procedure described in Section 5, we have:

$$\begin{aligned} p\left(\tilde{X};\mathbf{\Phi}\right) &= p\left(\tilde{S},\tilde{Y};\mathbf{\Phi}\right) \\ &= p\left(\tilde{V}_2\left|\tilde{S};\mathbf{\Phi}\right.\right) p\left(\tilde{V}_1\left|\tilde{S}\right.\right) p\left(\tilde{S}\right). \end{aligned} \tag{3.88}$$

In the following, we evaluate the terms $p_{\tilde{S}}(\tilde{S})$, $p_{\tilde{V}_1|\tilde{S}}(\tilde{V}_1\left|\tilde{S}\right.)$ and $p_{\tilde{V}_2|\tilde{S}}(\tilde{V}_2\left|\tilde{S};\mathbf{\Phi}\right.)$. We start with $p_{\tilde{S}}(\tilde{S})$:

$$p\left(\tilde{S}\right) = \prod_{l=1}^{N_t} p_{S^l}\left(S^l\right) = \prod_{l=1}^{N_t} p_{\mathbf{s}_1^l}\left(\mathbf{s}_1^l\right) \prod_{k=2}^{K} p_{\mathbf{s}_k^l|\mathbf{s}_{k-1}^l}\left(\mathbf{s}_k^l\left|\mathbf{s}_{k-1}^l\right.\right), \tag{3.89}$$

where:

$$\mathbf{s}_1^l \sim \mathcal{N}\left(\boldsymbol{\mu}_0^l, \mathbf{Q}_w^l\right), \tag{3.90}$$

$$\mathbf{s}_k^l\left|\mathbf{s}_{k-1}^l\right. \sim \mathcal{N}\left(\mathbf{F}\mathbf{s}_{k-1}^l, \mathbf{Q}_w^l\right). \tag{3.91}$$

The term $p_{\tilde{V}_1|\tilde{S}}(\tilde{V}_1\left|\tilde{S}\right.)$ can be evaluated as follows:

$$p\left(\tilde{V}_1\left|\tilde{S}\right.\right) = \prod_{l=1}^{N_t} p_{V_1^l|S^l}\left(V_1^l\left|S^l\right.\right) = \prod_{l=1}^{N_t}\prod_{k=1}^{K} p_{\mathbf{v}_{1,m}^{k,l}|\mathbf{s}_k^l}\left(\mathbf{v}_{1,m}^{k,l}\left|\mathbf{s}_k^l\right.\right), \tag{3.92}$$

where:

$$\mathbf{v}_{1,m}^{k,l}\left|\mathbf{s}_k^l\right. \sim \mathcal{N}\left(\tilde{\mathbf{h}}^{-1}\left(\mathbf{s}_k^l\right), \mathbf{C}_1\right). \tag{3.93}$$

Finally, the term $p\left(\tilde{V}_2\left|\tilde{S};\mathbf{\Phi}\right.\right)$ can be evaluated as:

$$p\left(\tilde{V}_2\left|\tilde{S};\boldsymbol{\Phi}\right.\right)=\prod_{l=1}^{N_t}p_{V_2^l\left|S^l\right.}\left(V_2^l\left|S^l;\boldsymbol{\Phi}\right.\right)=\prod_{l=1}^{N_t}\prod_{k=1}^{K}p_{\mathbf{v}_{2,m}^{k,l}\left|\mathbf{s}_k^l\right.}\left(\mathbf{v}_{2,m}^{k,l}\left|\mathbf{s}_k^l;\boldsymbol{\Phi}\right.\right),\tag{3.94}$$

where:

$$\mathbf{v}_{2,m}^{k,l}\left|\mathbf{s}_k^l\right.\sim\mathcal{N}\left(\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k^l,\boldsymbol{\Phi}\right),\mathbf{C}_2\right).\tag{3.95}$$

Hence, the logarithm of the pdf of the complete data $\tilde{X}$ can be expressed, ignoring constant terms, as:

$$\ln p_{\tilde{X}}\left(\tilde{X};\boldsymbol{\Phi}\right)=-\frac{1}{2}\sum_{l=1}^{N_t}\sum_{k=1}^{K}\left[\mathbf{v}_{2,m}^{k,l}-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k^l,\boldsymbol{\Phi}\right)\right]\mathbf{C}_2^{-1}\left[\mathbf{v}_{2,m}^{k,l}-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k^l,\boldsymbol{\Phi}\right)\right]^T+const.\tag{3.96}$$

Now, by defining the vector $\boldsymbol{\alpha}_k^l=\mathbf{v}_{2,m}^{k,l}-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k^l,\boldsymbol{\Phi}\right)$, the objective function $Q(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n)$, i.e. the conditional expectation of $\ln p_{\tilde{X}}(\tilde{X};\boldsymbol{\Phi})$ (neglecting constant terms w.r.t $\boldsymbol{\Phi}$), given the incomplete data set *Y* and the current estimate of the parameters vector $\hat{\boldsymbol{\Phi}}^n$, can be evaluated as:

$$Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right)=-\frac{1}{2}\sum_{l=1}^{N_t}\mathrm{tr}\left\{\mathbf{C}_2^{-1}\sum_{k=1}^{K}E\left\{\boldsymbol{\alpha}_k^l(\boldsymbol{\alpha}_k^l)^T\left|V_1^l,V_2^l;\hat{\boldsymbol{\Phi}}^n\right.\right\}\right\}=\sum_{l=1}^{N_t}Q^l\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right),\tag{3.97}$$

where $Q^l(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n),\ l=1,\ldots,N_t$, are the objective functions for each target that can be evaluated exactly as described in Section 5. Eq. (3.97) shows that the objective function obtained by applying the EM algorithm to the multi-target scenario is simply the sum of the objective functions obtained for the single targets.

3.9.1 Performance bound for the multi-target scenario

In this section, we generalize the HCRLB, described in Section 3.7, to the multi-target scenario. The hybrid parameter vector is:

$$\tilde{\boldsymbol{\Psi}}=\left[\boldsymbol{\Phi}^T,(\mathbf{s}_1^1)^T,\ldots,(\mathbf{s}_K^1)^T,\ldots,(\mathbf{s}_1^{N_t})^T,\ldots,(\mathbf{s}_K^{N_t})^T\right]^T.\tag{3.98}$$

Then, following the same procedure described in Section 6, the HFIM for this hybrid parameter estimation problem can be expressed as:

$$\begin{aligned}\left[\mathbf{I}_H\left(\tilde{\boldsymbol{\Psi}}\right)\right]_{ij} &= -E_{\tilde{V}_1,\tilde{V}_2,\tilde{S}}\left\{\frac{\partial^2 \ln p\left(\tilde{V}_1,\tilde{V}_2,\tilde{S};\boldsymbol{\Phi}\right)}{\partial\tilde{\Psi}_i\partial\tilde{\Psi}_j}\right\} \\ &= -E_{\tilde{V}_1,\tilde{V}_2,\tilde{S}}\left\{\frac{\partial^2 \ln\left(p\left(\tilde{V}_1\middle|\tilde{S}\right)p\left(\tilde{V}_2\middle|\tilde{S};\boldsymbol{\Phi}\right)p\left(\tilde{S}\right)\right)}{\partial\tilde{\Psi}_i\partial\tilde{\Psi}_j}\right\},\end{aligned} \tag{3.99}$$

From eq. (3.92), the logarithm of the conditional pdf of the measurements coming from radar#1 can be expressed as:

$$\ln p\left(\tilde{V}_1\middle|\tilde{S}\right) = \sum_{l=1}^{N_t}\sum_{k=1}^{K}\ln p_{\mathbf{v}_{1,m}^{k,l}\middle|\mathbf{s}_k^l}\left(\mathbf{v}_{1,m}^{k,l}\middle|\mathbf{s}_k^l\right), \tag{3.100}$$

where $\mathbf{v}_{1,m}^{k,l}\middle|\mathbf{s}_k^l \sim \mathcal{N}(\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k^l),\mathbf{C}_1)$. Similarly, from eq. (3.94), the logarithm of the conditional pdf of the measurements of radar #2 is:

$$\ln p\left(\tilde{V}_2\middle|\tilde{S};\boldsymbol{\Phi}\right) = \sum_{l=1}^{N_t}\sum_{k=1}^{K}\ln p_{\mathbf{v}_{2,m}^{k,l}\middle|\mathbf{s}_k^l}\left(\mathbf{v}_{2,m}^{k,l}\middle|\mathbf{s}_k^l;\boldsymbol{\Phi}\right), \tag{3.101}$$

where $\mathbf{v}_{2,m}^{k,l}\middle|\mathbf{s}_k^l \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k^l,\boldsymbol{\Phi}),\mathbf{C}_2)$. Finally, from eq. (3.89), the logarithm of the joint pdf of all the target state vectors is given by:

$$\ln p\left(\tilde{S}\right) = \sum_{l=1}^{N_t}\ln p\left(\mathbf{s}_1^l\right) + \sum_{l=1}^{N_t}\sum_{k=2}^{K}\ln p_{\mathbf{s}_k^l\middle|\mathbf{s}_{k-1}^l}\left(\mathbf{s}_k^l\middle|\mathbf{s}_{k-1}^l\right), \tag{3.102}$$

where $\mathbf{s}_1^l \sim \mathcal{N}(\boldsymbol{\mu}_0^l,\mathbf{Q}_w^l)$ and $\mathbf{s}_k^l\middle|\mathbf{s}_{k-1}^l \sim \mathcal{N}(\mathbf{F}\mathbf{s}_{k-1}^l,\mathbf{Q}_w^l)$. As before, the HFIM can be rewritten as:

$$\left[\mathbf{I}_H\left(\boldsymbol{\Phi}\right)\right]_{ij} = E_S\left\{\mathbf{I}_{H|\tilde{S}}\left(\tilde{\boldsymbol{\Psi}}\right)\right\}, \tag{3.103}$$

where the conditional HFIM is defined as:

$$\begin{aligned}\left[\mathbf{I}_{H|\tilde{S}}\left(\tilde{\boldsymbol{\Psi}}\right)\right]_{ij} &= -E_{\tilde{V}_1,\tilde{V}_2|\tilde{S}}\left\{\frac{\partial^2 \ln p\left(\tilde{V}_1,\tilde{V}_2,\tilde{S};\boldsymbol{\Phi}\right)}{\partial\tilde{\Psi}_i\partial\tilde{\Psi}_j}\right\} \\ &= \sum_{l=1}^{N_t}\left[\sum_{k=1}^{K} l_{ij}^l\left(\mathbf{s}_k^l\right) + \sum_{k=1}^{K} g_{ij}^l\left(\mathbf{s}_k^l;\boldsymbol{\Phi}\right) + u_{ij}^l\left(\mathbf{s}_k^l\right)\right],\end{aligned} \tag{3.104}$$

and

$$l_{ij}^{l}\left(\mathbf{s}_{k}^{l}\right)\triangleq -E_{\mathbf{v}_{1,m}^{k,l}\left|\mathbf{s}_{k}^{l}\right.}\left\{\frac{\partial^{2}\ln p\left(\mathbf{v}_{1,m}^{k,l}\left|\mathbf{s}_{k}^{l}\right.\right)}{\partial\tilde{\Psi}_{i}\partial\tilde{\Psi}_{j}}\right\}, \tag{3.105}$$

$$g_{ij}^{l}\left(\mathbf{s}_{k}^{l};\boldsymbol{\Phi}\right)\triangleq -E_{\mathbf{v}_{2,m}^{k,l}\left|\mathbf{s}_{k}^{l}\right.}\left\{\frac{\partial^{2}\ln p\left(\mathbf{v}_{2,m}^{k,l}\left|\mathbf{s}_{k}^{l};\boldsymbol{\Phi}\right.\right)}{\partial\tilde{\Psi}_{i}\partial\tilde{\Psi}_{j}}\right\}, \tag{3.106}$$

$$u_{ij}^{l}\left(\mathbf{s}_{k}^{l}\right)\triangleq -\frac{\partial^{2}\ln p\left(\mathbf{s}_{1}^{l}\right)}{\partial\tilde{\Psi}_{i}\partial\tilde{\Psi}_{j}}-\sum_{k=2}^{K}\frac{\partial^{2}\ln p\left(\mathbf{s}_{k}^{l}\left|\mathbf{s}_{k-1}^{l}\right.\right)}{\partial\tilde{\Psi}_{i}\partial\tilde{\Psi}_{j}}. \tag{3.107}$$

It can be noted that eqs. (3.105), (3.106), and (3.107) are exactly the same as eqs. (3.67), (3.68), and (3.69). Therefore, they can be evaluated as described in Section 3.7 and in Appendix D. In the following, we give the block-matrix form of the conditional HFIM defined in eq. (3.104):

$$\mathbf{I}_{H|\tilde{S}}\left(\tilde{\boldsymbol{\Psi}}\right)=\begin{pmatrix}\sum_{l=1}^{N_t}\sum_{k=1}^{K}\mathbf{F}_{k}^{l} & \mathbf{B}^{1} & \mathbf{B}^{2} & \cdots & \mathbf{B}^{N_t}\\ (\mathbf{B}^{1})^{T} & \mathbf{T}^{1} & & \cdots & \mathbf{0}_{6K\times 6K}\\ (\mathbf{B}^{2})^{T} & & \ddots & & \vdots\\ \vdots & \vdots & & \ddots & \vdots\\ (\mathbf{B}^{N_t})^{T} & \mathbf{0}_{6K\times 6K} & \cdots & & \mathbf{T}^{N_t}\end{pmatrix}, \tag{3.108}$$

where:

$$\mathbf{B}^{l}=\begin{pmatrix}\mathbf{B}_{1}^{l} & \cdots & \mathbf{B}_{K}^{l}\end{pmatrix}, \tag{3.109}$$

$$\mathbf{T}^{l}=\begin{pmatrix}\boldsymbol{\Pi}_{1}^{l} & \boldsymbol{\Lambda}^{l} & \cdots & \mathbf{0}_{6\times 6}\\ (\boldsymbol{\Lambda}^{l})^{T} & \boldsymbol{\Pi}_{2}^{l} & \ddots & \vdots\\ \vdots & \ddots & \ddots & \boldsymbol{\Lambda}^{l}\\ \mathbf{0}_{6\times 6} & \cdots & (\boldsymbol{\Lambda}^{l})^{T} & \boldsymbol{\Xi}^{l}\end{pmatrix}, \tag{3.110}$$

and

$$\boldsymbol{\Lambda}^{l}=-\mathbf{F}^{T}(\mathbf{Q}_{w}^{l})^{-1},\quad \boldsymbol{\Pi}_{k}^{l}=\mathbf{N}_{k}^{l}+\mathbf{M}_{k}^{l}+\boldsymbol{\Omega}^{l},\quad \boldsymbol{\Xi}^{l}=\mathbf{N}_{K}^{l}+\mathbf{M}_{K}^{l}+(\mathbf{Q}_{w}^{l})^{-1}. \tag{3.111}$$

Finally, it must be noted that matrices $\mathbf{F}_{k}^{l}$, $\mathbf{N}_{k}^{l}$, $\mathbf{M}_{k}^{l}$, $\mathbf{B}_{k}^{l}$ and $\boldsymbol{\Omega}^{l}$ are defined, for a given *l*, in Section 3.7 and the explicit calculation can be found in Appendix D. Also in this case, the mean value w.r.t. *S* in eq. (3.103) is evaluated through independent Monte Carlo

trials. As we can see from the expression of the conditional HFIM in eq. (3.108), by increasing the number of targets, the information relative to the grid-locking parameters increases, too (provided that the association of the new detections to the plots has been done correctly!). The first top-left block matrix in fact represents the amount of information relative to the deterministic vector parameter $\mathbf{\Phi}$ and it increases linearly with the number of the available target.

3.9.2 Simulation results

In this section, we show a comparison between the performance of the proposed EM algorithm and the LS algorithm for a multi-target scenario. The comparison is carried out in terms of root mean square error (RMSE) as function of the number of observations. For the sake of brevity, only the results relative to three bias errors (range bias, roll bias and the x component of the position bias) are reported here. Radar #1 is in the centre of the absolute reference system, while the position vector of radar #2 is $\mathbf{t}_t=(2,2,2)\cdot 10^3$ m (see Fig. 2). There are three targets which are supposed to follow the discrete target state model introduced in eq. (3.87) with an initial target state vector given by $\mathbf{s}_0{}^l=[(\mathbf{r}_0{}^l)^T\ (\mathbf{u}_0{}^l)^T]^T$, l=1,2,3 where:

- Target #1: $\mathbf{r}_0{}^1=[1000\ 1000\ 0]^T$, $\mathbf{u}_0{}^1=[7.63\ 0\ 0]^T$,
- Target #2: $\mathbf{r}_0{}^2=[-1000\ 2000\ 0]^T$, $\mathbf{u}_0{}^2=[3\ 7.63\ 0]^T$,
- Target #3: $\mathbf{r}_0{}^3=[2000\ -1500\ 0]^T$, $\mathbf{u}_0{}^3=[8\ 2\ 0]^T$,

The power spectral density (PSD) for each component of the continuous-time process noise vector is equal to $N_{0,x}=N_{0,y}=N_{0,z}=q^l=0.01\text{m}^2/\text{s}^3$, for l=1,2,3. The actual bias error values have been set as: *i*) measurement biases: $d\rho$=-10m, $d\theta=d\varepsilon$=-0.0573°; *ii*) attitude biases: $d\chi=d\psi$=-0.0573° and $d\zeta$=-0.1146°; *iii*) position biases: $dt_x=dt_y=dt_z$=-30 m. Radars #1 and #2 are characterized by the following accuracies: *i*) $\sigma_{\rho,1}=\sigma_{\rho,2}$=50 m; *ii*) $\sigma_{\theta,i}=\sigma_{\varepsilon,i}$=0.3°, i=1,2. Moreover, the probability of detection and false alarm are assumed to be P_D=1 and P_{FA}=0, respectively. The scan time of both radars is assumed equal to 1 sec. The target tracks plotted in Fig. 3.30 are three realizations of the discrete state model in eq. (3.87). The number of available observations varies from 200 to 400.

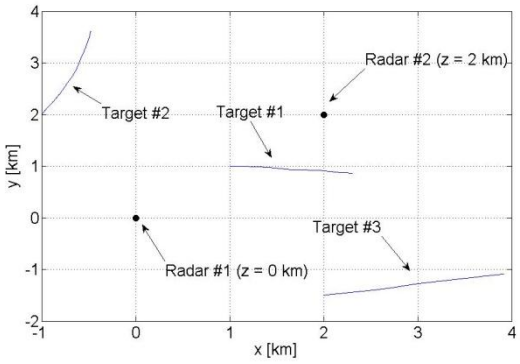

Figure 3.30 – Geometry of the multi-target scenario.

Figs 3.31-3.33, show the RMSE of the LS algorithm and of the EM algorithm and the HCRLB as a function of the number of observations, i.e. the number K of the measurements from radars #1 and #2. As starting point of the EM algorithm we chose the LS estimate of the parameter vector $\mathbf{\Phi}$ and the recursion is stopped after 10 iterations. The RMSE is evaluated using 200 independent Monte Carlo trials.

As we can see from Figs 3.31-3.33, it can be noted that, the EM algorithm outperforms the LS algorithm. In Figs. 3.34–3.36, a comparison between the RMSE on the estimate of the registration errors for the single target scenario analyzed in [8] and for the multi-target scenario is shown. This comparison allows us to evaluate the performance improvement due to the increase of the number of the available targets. From Figs. 6-8 we get that an increase of the number of the available targets correspond to an estimation performance improvement. This behaviour is explained by taking into account the expression of the conditional HFIM in eq. (3.108): the amount of information about the parameter vector $\mathbf{\Phi}$ grows linearly with the number of available targets. Finally, we observe that the EM estimator is not efficient (w.r.t. the HCRLB). However, the HCRLB for the deterministic parameter vector estimate is not always (asymptotically) tight [19].

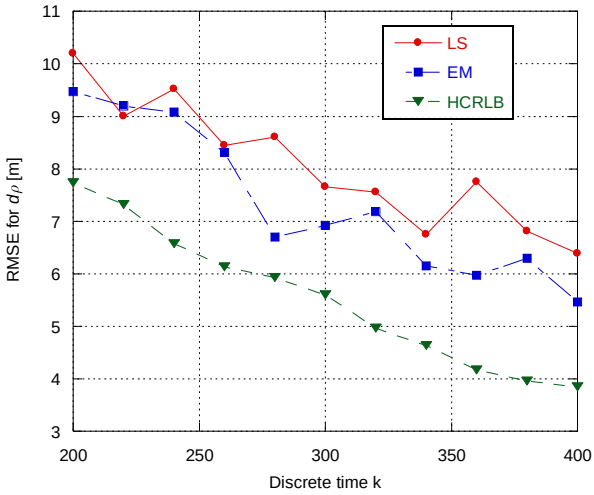

Figure 3.31 – Comparison between the RMSE for the LS and EM estimate of the range error.

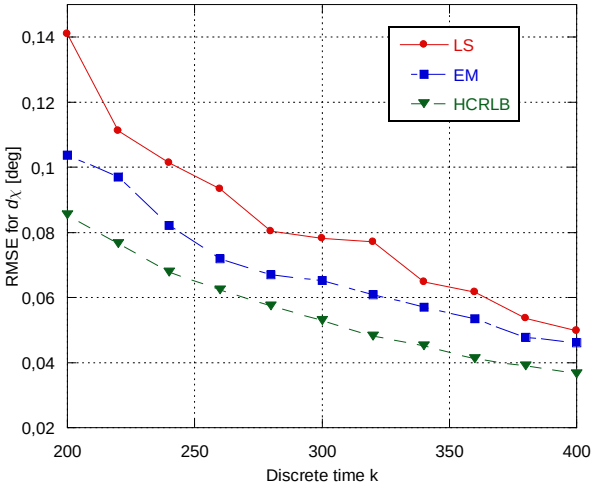

Figure 3.32 - Comparison between the RMSE for the LS and EM estimate for the estimate of the roll error.

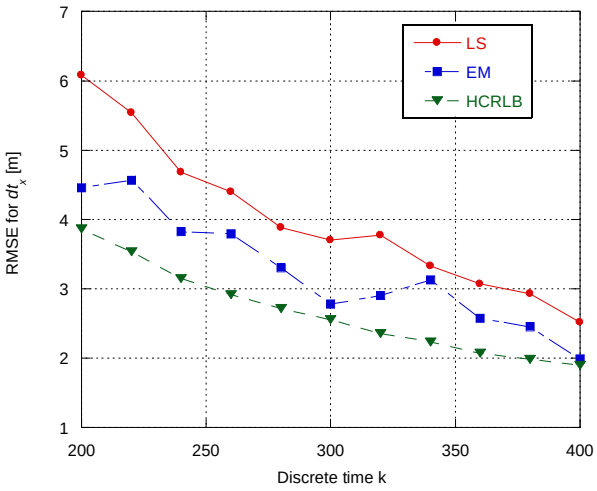

Figure 3.33 - Comparison between the RMSE for the LS and EM estimate for the estimate of dt_x.

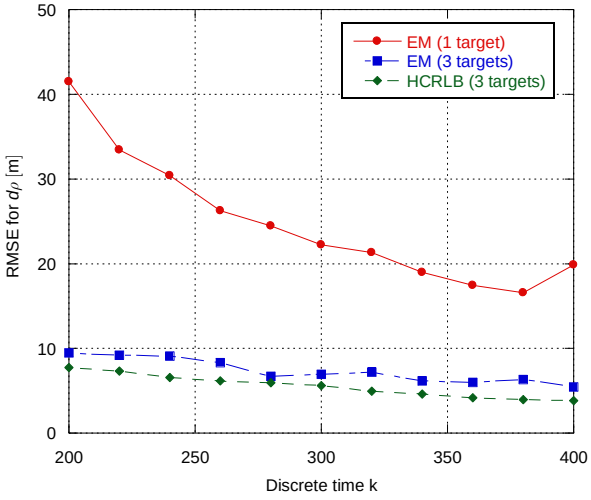

Figure 3.34 – Comparison between the RMSE of the EM estimate of the range error in a single and in a multi-target scenario.

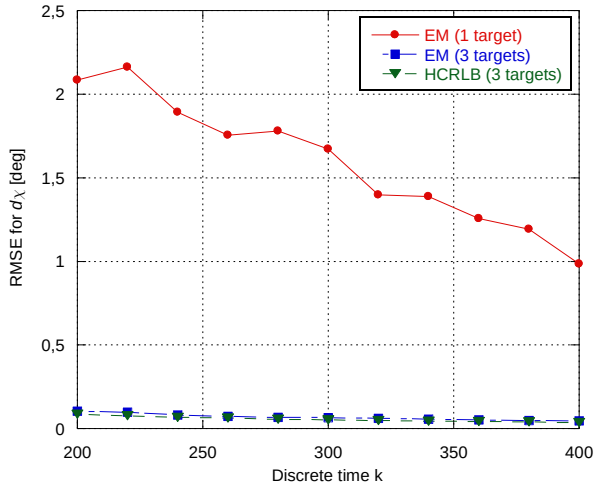

Figure 3.35 - Comparison between the RMSE of the EM estimate of the roll error in a single and in a multi-target scenario.

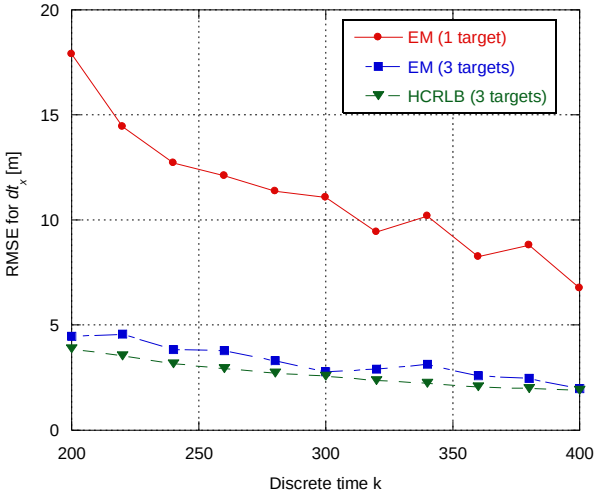

Figure 3.36 - Comparison between the RMSE of the EM estimate of dt_x in a single and in a multi-target scenario.

3.10 The identifiability problem

As discussed in the Introduction, the last Section of this Chapter is dedicated to the identifiability problem. First, we provide a general description of the identifiability problem in its classical formulation. Then, we generalize the previous results to take into account the presence of random nuisance parameters. A different definition of identifiability is provided and its relationship with the classical one is investigated. Finally, we show that this general framework can be directly used to validate the intuitive result on the impossibility to estimate separately the azimuth and the yaw bias errors.

3.10.1 General formulation of the identifiability problem

3.10.1.A Some preliminary definitions

Let $\mathbf{x} \in \mathbb{R}^n$ be a *n*-dimensional random vector, representing the outcome of some random experiment, whose probability density function (pdf) is known to belong to a family $\mathcal{F}$. A *structure T* is a set of hypotheses which implies a unique pdf in $\mathcal{F}$ for **x**. Such pdf is indicated with $p(\mathbf{x};T) \in \mathcal{F}$ [22], [24]. The set of all the a priori possible structures is called a *model* and is denoted by $\mathcal{M}$. By definition, there exist a unique pdf associated with each structure in $\mathcal{M}$.

Definition 1: *Two structures T_0 and T_1 in $\mathcal{M}$ are said to be observationally equivalent if they imply the same pdf for the observable random vector* **x**. *The structure T_0 is otherwise said to be identifiable if there is no other structure in $\mathcal{M}$ which is observationally equivalent.*

We assume that the pdf of the random vector **x** has a parametric representation , i.e. we assume that every structure *T* is described by an *m*-dimensional vector $\mathbf{\Phi}$ and that the model is described by a set $\Omega \subseteq \mathbb{R}^m$. It is possible to associate with each $\mathbf{\Phi}$ in Ω a continuous pdf $p(\mathbf{x};\mathbf{\Phi}) \in \mathcal{F}$ which is perfectly known except for the values of the parameter vector $\mathbf{\Phi}$.

Definition 2: *Two parameter vectors $\mathbf{\Phi}_0$ and $\mathbf{\Phi}_1$ (relative to two structures T_0 and T_1) are said to be observationally equivalent if* $p(\mathbf{x};\mathbf{\Phi}_0) = p(\mathbf{x};\mathbf{\Phi}_1)$ *for all* $\mathbf{x} \in \mathbb{R}^n$. $\mathbf{\Phi}_0$ *is*

otherwise said to be identifiable if there is no other $\boldsymbol{\Phi}$ *in* Ω *which is observationally equivalent.*

Since the set of the structures is a subset of $\mathbb{R}^m$ then it is possible to endow it with the same topological structure of $\mathbb{R}^m$. This allows us to consider the concept of *local* identification:

Definition 3: *A parameter vector* $\boldsymbol{\Phi}_0$ *is said to be locally identifiable is there exists an open neighborhood of* $\boldsymbol{\Phi}_0$ *containing no other* $\boldsymbol{\Phi}$ *in* Ω *which is observationally equivalent.*

To highlight the difference between the Definitions 2 and 3, in the following we indicate as *global* the identifiability in Definition 2 and as *local* the identifiability in Definition 3.

Before providing a general identifiability criterion, in the following we enumerate some necessary assumptions [1]:

Assumption 1: *The structural parameter space* Ω *is an open subset of* $\mathbb{R}^m$.

Assumption 2: *The function* $p(\mathbf{x};\boldsymbol{\Phi})$ *is a proper pdf for every* $\boldsymbol{\Phi} \in \Omega$. *In particular,* $p(\mathbf{x};\boldsymbol{\Phi})$ *is nonnegative, and the equation* $\int p(\mathbf{x};\boldsymbol{\Phi}) d\mathbf{x} = 1$ *holds for all* $\boldsymbol{\Phi} \in \Omega$.

Assumption 3: *The set* $\mathcal{X}$ *of* $\mathbf{x}$ *values for which* $p(\mathbf{x};\boldsymbol{\Phi})$ *is strictly positive is the same for all* $\boldsymbol{\Phi} \in \Omega$. *The set* $\mathcal{X}$ *is the sample space of the random vector* $\mathbf{x}$.

Assumption 4: *The function* $p(\mathbf{x};\boldsymbol{\Phi})$ *is smooth in* $\boldsymbol{\Phi} \in \Omega$. *Specifically, we assume that, for all* $\boldsymbol{\Phi}$ *in a convex set containing* Ω *and for all* $\mathbf{x}$ *in the sample space* $\mathcal{X}$, *the functions* $p(\mathbf{x};\boldsymbol{\Phi})$ *and* $\ln p(\mathbf{x};\boldsymbol{\Phi})$ *are continuously differentiable with respect to* $\boldsymbol{\Phi}$.

3.10.1.B A general identifiability criterion

In [23], a general criterion, based on the Kullback-Liebler divergence, for the identifiability of parameter vectors is proposed. Here we report only the main facts, all the proofs can be found in [23]. First of all, we recall the definition of the Kullback-Liebler divergence [37]:

Definition 4: *Let* $p(\mathbf{x};\boldsymbol{\Phi})$ *and* $p(\mathbf{x};\boldsymbol{\Phi}_0)$ *be two parametric pdfs for all* $\boldsymbol{\Phi}\in\Omega$. *The scalar function of the vector variable* $\boldsymbol{\Phi}$, $H(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$, *defined as:*

$$H\left(\boldsymbol{\Phi};\boldsymbol{\Phi}_0\right)\triangleq E\left\{\ln\frac{p\left(\mathbf{x};\boldsymbol{\Phi}\right)}{p\left(\mathbf{x};\boldsymbol{\Phi}_0\right)}\right\}=\int\ln\frac{p\left(\mathbf{x};\boldsymbol{\Phi}\right)}{p\left(\mathbf{x};\boldsymbol{\Phi}_0\right)}p\left(\mathbf{x};\boldsymbol{\Phi}_0\right)d\mathbf{x} \tag{3.112}$$

is called Kullback-Liebler divergence between $p(\mathbf{x};\boldsymbol{\Phi})$ *with* $\boldsymbol{\Phi}\in\Omega$ *and* $p(\mathbf{x};\boldsymbol{\Phi}_0)$.

One of the most important theorems on the Kullback-Liebler divergence is (the proof can be found in Lemma 1, Ch. 8 of [38]):

Theorem 1: *Let* $p(\mathbf{x};\boldsymbol{\Phi})$ *and* $p(\mathbf{x};\boldsymbol{\Phi}_0)$ *be two parametric pdfs. If* $p\left(\mathbf{x};\boldsymbol{\Phi}\right)=p\left(\mathbf{x};\boldsymbol{\Phi}_0\right)$ *for all* $\mathbf{x}\in\mathbb{R}^n$, *then* $H(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)=0$. *Otherwise, if* $H(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$ *is finite,* $H(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)<0$.

In view of the Definitions 2, 3 and 4, the link between the Kullback-Liebler divergence and the identifiability of a parameter vector is given by the following Corollary:

Corollary 1: *Let* $p(\mathbf{x};\boldsymbol{\Phi})$ *and* $p(\mathbf{x};\boldsymbol{\Phi}_0)$ *be two parametric pdfs for all* $\boldsymbol{\Phi}\in\Omega$. *Then the parameter vector* $\boldsymbol{\Phi}_0$ *is globally identifiable if and only if the equation* $H(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)=0$ *has as solution in* Ω *only* $\boldsymbol{\Phi}=\boldsymbol{\Phi}_0$. *It is locally identifiable if and only if* $\boldsymbol{\Phi}=\boldsymbol{\Phi}_0$ *is the only solution in some open neighborhood of* $\boldsymbol{\Phi}_0$.

It can be noted also that the identifiability condition is closely related to the maximum of the $H(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$. In fact, from Theorem 1 it follows that: if the maximum of $H(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$ is global and attained at $\boldsymbol{\Phi}=\boldsymbol{\Phi}_0$, then $\boldsymbol{\Phi}_0$ is globally identifiable, whereas, if there exists an open neighborhood of $\boldsymbol{\Phi}_0$ with a local maximum in $\boldsymbol{\Phi}_0$, then $\boldsymbol{\Phi}_0$ is locally identifiable. Such consideration suggests another general identification criterion that we provide in the following Corollary (the proof can be found in [23]).

Corollary 2: *Let* $p(\mathbf{x};\boldsymbol{\Phi})$ *and* $p(\mathbf{x};\boldsymbol{\Phi}_0)$ *be two parametric pdfs for all* $\boldsymbol{\Phi}\in\Omega$. *Then the parameter vector* $\boldsymbol{\Phi}_0$ *is locally identifiable if and only if the Hessian Matrix* $\mathbf{H}$ of $H(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$ evaluated at $\boldsymbol{\Phi}_0$, i.e. $\mathbf{H}(H)(\boldsymbol{\Phi}_0)$, *is a negative definite matrix. Moreover, it can be shown that:*

$$\begin{aligned}\left[\mathbf{H}(H)\right]_{ij}(\boldsymbol{\Phi}_0) &= -E\left\{\frac{\partial}{\partial\Phi_i}p(\mathbf{x};\boldsymbol{\Phi})\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0}\frac{\partial}{\partial\Phi_j}p(\mathbf{x};\boldsymbol{\Phi})\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0}\right\} \\ &= E\left\{\frac{\partial^2}{\partial\Phi_i\partial\Phi_j}p(\mathbf{x};\boldsymbol{\Phi})\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0}\right\} = -\left[\mathbf{I}(\boldsymbol{\Phi}_0)\right]_{ij},\end{aligned} \tag{3.113}$$

where $\mathbf{I}(\boldsymbol{\Phi}_0)$ *is the Fisher Information Matrix (FIM).*

Taking into account eq. (3.113), the following Corollary can be finally derived [23]:

Corollary 3: *Let* $p(\mathbf{x};\boldsymbol{\Phi}_0)$ *be a parametric pdf. Then the parameter vector* $\boldsymbol{\Phi}_0$ *is locally identifiable if and only if the Fisher Information Matrix* $\mathbf{I}(\boldsymbol{\Phi}_0)$ *is a positive definite full rank matrix.*

3.10.2 Identifiability in presence of random nuisance parameters

In practical applications, a wide class of estimation problem involves the so-called nuisance parameters, i.e. random parameters that affect the data model whose estimation is not strictly required and that are known only through their statistical distribution. The aim of this section is to generalize previous results on the identifiability to take the nuisance parameters into account.

As before, let $\mathbf{x}\in\mathbb{R}^n$ be a *n*-dimensional random vector, representing the outcome of some random experiment, let $\mathbf{a}\in\mathbb{R}^l$ be the *l*-dimensional random vector of nuisance parameters and let $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)$ the joint pdf of the random vectors **x** and **a** parameterized by the deterministic vector $\boldsymbol{\Phi}_0$ to be estimated. Such pdf is assumed perfectly known. In the rest of the paper, we assume verified, as well as the Assumptions 1-4, the following:

Assumption 5: *The pdf of the nuisance parameters* $p(\mathbf{a})$ *does not depend on the parameter vector* $\boldsymbol{\Phi}_0$. *Then, the joint pdf* $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)$ *can be always factorized as:* $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)=p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi}_0)p(\mathbf{a})$.

To apply Theorem 1 and Corollary 3 to this estimation problem, we have to evaluate the marginal pdf of the data **x**:

$$p(\mathbf{x};\boldsymbol{\Phi}_0) = \int p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)\, d\mathbf{a}. \tag{3.114}$$

Unfortunately, in many practical applications, the closed form of the integral in eq. (3.114) is extremely difficult (or impossible) to calculate and this has motivated the search for a more general identifiability criterion. When the marginal pdf of the data $p(\mathbf{x};\boldsymbol{\Phi}_0)$ is unavailable, we can use the joint pdf $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)$ to define a new identifiability criterion. To this purpose, Definition 2 can be modified as described in the following Definitions:

Definition 5: *Two parameter vectors* $\boldsymbol{\Phi}_0$ *and* $\boldsymbol{\Phi}_1$ *(relative to two structures* T_0 *and* T_1*) are said to be observationally equivalent if* $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0) = p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_1)$ *for all* $\mathbf{x} \in \mathbb{R}^n$ *and for all* $\mathbf{a} \in \mathbb{R}^l$. $\boldsymbol{\Phi}_0$ *is otherwise said to be identifiable if there is no other* $\boldsymbol{\Phi}$ *in* Ω *which is observationally equivalent.*

At this point, some considerations on the differences between Definitions 2 and 5 need to be made. According to these two definitions, a parameter vector $\boldsymbol{\Phi}_0$ is *non-identifiable* if at least another $\boldsymbol{\Phi}_1$ exists such that:

i. Definition 2: $p(\mathbf{x};\boldsymbol{\Phi}_0) = p(\mathbf{x};\boldsymbol{\Phi}_1) \quad \forall \mathbf{x} \in \mathbb{R}^n$ where $p(\mathbf{x};\boldsymbol{\Phi}_i) = \int p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_i)\, d\mathbf{a}$, for i=0,1.

ii. Definition 5: $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0) = p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_1) \quad \forall \mathbf{x} \in \mathbb{R}^n, \ \forall \mathbf{a} \in \mathbb{R}^l$.

Roughly speaking, Definition 5 requires that the parameter vector $\boldsymbol{\Phi}$ is identifiable for any realization of **a** under the implicit assumption that **x** and **a** are jointly directly observed. This is not true in reality, since the random nuisance parameter vector **a** is not directly observed, but only indirectly through the measurement vector **x**. In Definition 2, **x** is observed and **a** is unknown, but it is averaged out in the pdf, so we do not require it to be observed (known).

In the following we derive an operative procedure to verify if, in presence of random nuisance parameters, a parameter vector $\boldsymbol{\Phi}_0$ is identifiable or not under Definitions 5. To do this, we go through the same steps described in the previous section relative to the Definition 2.

3.10.2.A Identifiability condition under Definition 5

The aim of this section is to provide a condition to verify if the parameter vector $\boldsymbol{\Phi}_0$ is identifiable under Definition 5. We want to prove that there is no other parameter vector $\boldsymbol{\Phi} \in \Omega$, or at least in an open neighborhood of $\boldsymbol{\Phi}_0$ (local identifiability), such that $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}) = p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)$. First of all, we have to generalize the definition of the KL divergence under the Definition 5. This can be easily done by defining a scalar function of $\boldsymbol{\Phi}$, $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$, as:

$$H_M\left(\boldsymbol{\Phi};\boldsymbol{\Phi}_0\right) \triangleq E_{\mathbf{x},\mathbf{a}}\left\{\ln\left(\frac{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}\right)}{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0\right)}\right)\right\} = \int \ln\left(\frac{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}\right)}{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0\right)}\right) p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0\right) d\mathbf{x}d\mathbf{a}. \quad (3.115)$$

Now, we have to show that the Theorem 1 hold true under the Definition 5 with the generalized definition of KL divergence given in eq. (3.115). Under Definition 5, the Theorem 1 can be recast as follows:

Theorem 2: *Let* $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi})$ *and* $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)$ *be two parametric pdfs where* $\mathbf{a}$ *is the vector of the random nuisance parameters. If* $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}) = p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)$ *for all* $\mathbf{x} \in \mathbb{R}^n$ *and for all* $\mathbf{a} \in \mathbb{R}^l$, *then* $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0) = 0$. *Otherwise, if* $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$ *is finite,* $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0) < 0$.

Proof: The Theorem 2 can be easily proved using the Jensen inequality:

$$\begin{aligned} H_M\left(\boldsymbol{\Phi};\boldsymbol{\Phi}_0\right) &= E_{\mathbf{x},\mathbf{a}}\left\{\ln\left(\frac{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}\right)}{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0\right)}\right)\right\} \le \ln\left(E_{\mathbf{x},\mathbf{a}}\left\{\frac{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}\right)}{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0\right)}\right\}\right) \\ &= \ln\left(\int \frac{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}\right)}{p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0\right)} p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0\right) d\mathbf{x}d\mathbf{a}\right) = \ln\left(\int p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}\right) d\mathbf{x}d\mathbf{a}\right) \quad (3.116) \\ &= \ln(1) = 0. \end{aligned}$$

This conclude the proof.

At this point, following the same procedure used in the Section 2.2, it is possible to assert that $\boldsymbol{\Phi}_0$ is globally, or at least locally, identifiable if and only if it is a global, or at least a local, maximum for the KL divergence $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$ given in eq. (3.115). Then we have to show that the gradient of $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$, evaluated at $\boldsymbol{\Phi}_0$, i.e. $\nabla_{\boldsymbol{\Phi}}(H_M)(\boldsymbol{\Phi}_0)$, is equal to zero and that the Hessian matrix, also evaluated at $\boldsymbol{\Phi}_0$, i.e. $[\mathbf{H}(H_M)](\boldsymbol{\Phi}_0)$, is a negative

definite full rank matrix. As proved in Appendix E, it can be shown that the gradient is actually zero and that the Hessian matrix can be expressed as:

$$\begin{aligned}\left[\mathbf{H}\left(H_M\right)\right]_{ij}\left(\boldsymbol{\Phi}_0\right) &= \frac{\partial^2}{\partial\Phi_i\partial\Phi_j} H\left(\boldsymbol{\Phi};\boldsymbol{\Phi}_0\right)\Bigg|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \\ &= -E_{\mathbf{x},\mathbf{a}}\left\{\frac{\partial}{\partial\Phi_i} p\left(\mathbf{x}\middle|\mathbf{a};\boldsymbol{\Phi}\right)\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \frac{\partial}{\partial\Phi_j} p\left(\mathbf{x}\middle|\mathbf{a};\boldsymbol{\Phi}\right)\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0}\right\} \\ &= E_{\mathbf{x},\mathbf{a}}\left\{\frac{\partial^2}{\partial\Phi_i\partial\Phi_j} p\left(\mathbf{x}\middle|\mathbf{a};\boldsymbol{\Phi}\right)\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0}\right\} \\ &= -\left[\mathbf{I}_M\left(\boldsymbol{\Phi}_0\right)\right]_{ij},\end{aligned} \tag{3.117}$$

where $\mathbf{I}_M(\boldsymbol{\Phi}_0)$ is the so-called Modified Fisher Information Matrix (MFIM) [20], [29].

Starting from eq. (3.117), Corollary 3 can be generalized to take the random nuisance parameters into account.

Corollary 4: *Let* $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)$ *be a parametric pdf where* **x** *is the data vector and* **a** *is the vector of nuisance parameters and let* $\mathbf{I}_M(\boldsymbol{\Phi}_0)$ *be the MFIM. Then, the parameter vector* $\boldsymbol{\Phi}_0$ *is locally identifiable if and only if* $\mathbf{I}_M(\boldsymbol{\Phi}_0)$ *is a positive definite full rank matrix.*

3.10.3 Relationship among the identifiability conditions in presence of random nuisance parameters

The results obtained in the previous sections can be summarized in the following theorem:

Theorem 3: *Let* $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)$ *be a parametric pdf where* **x** *is the data random vector and* **a** *is the random vector of the nuisance parameters, let* $\mathcal{I}_2,\mathcal{I}_5\subseteq\mathbb{R}^m$ *be the sets of the parameter vectors globally observationally equivalent under Definitions 2 and 5, respectively, and let* $\mathcal{O}_2,\mathcal{O}_5\subseteq\mathbb{R}^m$ *be two open neighborhoods of* $\boldsymbol{\Phi}_0$ *that contain the parameter vectors locally observationally equivalent to* $\boldsymbol{\Phi}_0$ *under Definitions 2 and 5, respectively. Then, the following relations hold:*

$$\mathcal{I}_5\subseteq\mathcal{I}_2, \tag{3.118}$$

$$\mathcal{O}_5 \subseteq \mathcal{O}_2 \,. \tag{3.119}$$

Proof: The proof of Theorem 3 can be divided in two different parts: the first part, relative to the global identifiability, provides a proof of the relation (3.118), whereas the second part, relative to the local identifiability, provides a proof of the relation (3.119). To prove the first part, it is enough to investigate the relations between the Definitions 2 and 5. It is easy to show that Definition 5 implies Definitions 2. In fact, we have:

$$p\left(\mathbf{x};\boldsymbol{\Phi}_0\right) = \int p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0\right) d\mathbf{a} = \int p\left(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_1\right) d\mathbf{a} = p\left(\mathbf{x};\boldsymbol{\Phi}_1\right),\ \forall \mathbf{x} \tag{3.120}$$

that proves that Definition 5 implies Definition 2. On the other hand, to prove that Definition 2 does not imply Definitions 5, it is enough to show that there exist two different joint pdfs that have the same marginal pdfs of the data vector **x**. This concludes the proof of the relation (3.118) in Theorem 3.

The proof of the second part of Theorem 3 is straightforward if we take into account the results obtained in [20] and [29] about the relationships among the FIM and the MFIM. In fact, it can be proved that the following inequality holds:

$$\mathbf{I}\left(\boldsymbol{\Phi}_0\right) \leq \mathbf{I}_M\left(\boldsymbol{\Phi}_0\right) \tag{3.121}$$

where $\mathbf{A} \leq \mathbf{B}$ means that the matrix $\mathbf{B} - \mathbf{A}$ is a positive semi-definite matrix. The inequality given in eq. (3.121) has been proven in [29]. This concludes the proof of Theorem 3.

Theorem 3 states that the more restrictive identifiability condition is the Definition 2, as espected. This means that, if we use the Definitions 5 to test the identifiability of a deterministic parameter vector, it might be possible that we classify as identifiable a parameter that in reality it is not. However, in a lot of practical estimation problems that involve random nuisance parameters, it is impossible to apply the Definition 2 due to the analytical difficulties in the evaluation of the integral in eq. (3.114). In all these cases, when the classical FIM is impossible to obtain but the MFIM it is easy to evaluate, we can apply the Definition 5. Finally, by means of Theorem 3, it is possible to assert that if a parameter vector is not identifiable under Definition 5,then it is not identifiable under Definition 2 as well.

3.10.4 Identifiability in the relative grid-locking problem

In this last Section, we apply the theoretical framework on the identifiability, developed in the previous Section, to the relative grid-locking problem. In particular we show that the complete unknown parameter vector

$$\mathbf{\Phi} = \begin{bmatrix} d\rho & d\theta & d\varepsilon & d\chi & d\psi & d\xi & dt_x & dt_y & dt_z \end{bmatrix}^T \tag{3.122}$$

is not identifiable, and only a linear combination of the azimuth error $d\theta$ and of the yaw error $d\xi$ can be estimated. In Section 3.2 we discuss this point starting from geometrical considerations, here we show that the same conclusion can be drawn using the identifiability framework. Since the grid-locking estimation problem involves some nuisance parameters (i.e. the target trajectory), we could make use of the Theorem 3 introduced in Section 3.10.3. To proof that the vector in (3.122) is not (locally) identifiable, we have to show that the MFIM is rank deficient. Neglecting all the mathematical details on the evaluation of the MFIM (see Section 3.7) it can be show that the MFIM is actually rank deficient and this is caused by the coupling between $d\theta$ and $d\xi$ and the vector in (3.122) is not identifiable. As discussed in Section 3.2, to overcome this problem, we defined a new unknown parameter vector as:

$$\mathbf{\Phi}' = \begin{bmatrix} d\rho & d\varepsilon & d\chi & d\psi & d\zeta & dt_x & dt_y & dt_z \end{bmatrix}^T, \tag{3.123}$$

where $d\zeta = d\theta + d\xi$. It can be shown using exactly the same procedure as before that $\mathbf{\Phi}'$ is locally identifiable under Theorem 3.

3.11 Summary

In this Chapter we derive two algorithms for the joint estimate of all the relative gird-locking errors: a linear LS algorithm and an EM-based algorithm. The performance of the proposed algorithms is compared in terms of error mean value and RMSE. Moreover, their efficiency is investigated by comparing their RMSE with the hybrid Cramér-Rao lower bound. We used two different scenarios to assess the performance of the proposed algorithm: a single target scenario and a multi-target scenario. The numerical results show that the EM algorithm generally outperforms the LS algorithm, even if it is not tight to the

HCRLB. Finally, we provide a general framework to investigate the identifiability of a unknown parameters vector in all the estimation problems that involve random nuisance parameters and we use it to draw some additional consideration about the identifiability of the unknown parameter vector of the relative grid-locking errors.

REFERENCES

[1] M.P. Dana, "Registration: A prerequisite for multiple sensor tracking," in *Multitarget-Multisensor Tracking: Advanced Applications*, Ch. 5, pp 155-185. MA: Artech House, 1990.

[2] R. E. Helmick and T. R. Rice, "Removal of Alignment Errors in an Integrated System of Two 3-D Sensors", *IEEE Trans. on AES*, vol. 29, No. 4, pp. 1333-1343, October 1993.

[3] W. G. Bath, "Association of multisite radar data in the presence of large navigation and sensor alignment errors", *Proc. of the IEEE International Radar Conference 1982*, London, England, 18-20 October, 1982, pp. 169-173.

[4] M. J. Hirsch, P. M. Pardalos and M. G. C. Resende, "Sensor Registration in a Sensor Network by Continuous GRASP," in *Proc. IEEE Military Communications Conference, 2006*, Washington, D. C., USA, 23-25 Oct. 2006, pp.1-6.

[5] Y. Zhou, H. Leung and P. C. Yip, "An Exact Maximum Likelihood Registration Algorithm for Data Fusion," *IEEE Trans. on Signal Proc.*, vol. 45, no. 6, pp. 1560-1573, Jun 1997.

[6] A. Farina, R. Giordano, S. Pardini, "Algorithms for the compensation of alignment errors in the multiradar tracking system", *Rivista Tecnica Selenia*, vol.7, no. 2, pp. 10-15, 1980.

[7] A. Farina, F. A. Studer, *Radar Data Processing. Advanced Topics and Applications.* Researches Studies Press, John Wiley & Sons (USA), January 1986.

[8] J. J. Sudano, "A Least Square Algorithm with Covariance Weighting for Computing the Translational and Rotational Errors between Two Radar Sites," in *Proc. IEEE NAECON 1993*, Dayton, OH, USA, 24-28 May, 1993, vol. 1, pp. 383–387.

[9] I. Jónsdóttir and A. S. Hauksdóttir, "Integrity Monitoring and Estimation of Systematic Errors in Radar Data System", in *Proc. IEEE International Radar Conference*, Alexandria, VA, USA, May 8-11, 1995, pp. 310-316.

[10] B. Ristic and N. Okello, "Sensor registration in ECEF coordinates using the MLR algorithm," in *Proc. Information Fusion 2003*, Cairns, Australia, July 7-10, 2003, vol.1, pp. 135-142.

[11] B. Ristic, N. Okello and H. T. Ong, "Performance Bounds for Sensor Registration," *Proc. Information Fusion 2002*, Annapolis, USA, July 8-11, 2002, vol 1, pp 346-353.

[12] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", *Journal of the Royal Statistical Society*, vol. 39, no.1, 1977, pp. 1-38.

[13] A. Zia, T. Kirubarajan, J. P. Reilly, D. Yee, K. Punithakumar, and S. Shirani , "An EM Algorithm for Nonlinear State Estimation With Model Uncertainties," *IEEE Trans. on Signal Processing*, vol.56, no.3, pp.921-936, March 2008.

[14] D. Huang, H. Leung, "An expectation-maximization-based interacting multiple model approach for cooperative driving systems," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 6, no. 2, pp. 206- 228, June 2005.

[15] Z. Li, H. Leung, "An Expectation Maximization Based Simultaneous Registration and Fusion Algorithm for Radar Networks," *IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, pp. 31-35, Ottawa, May 2006.

[16] S. Chen, H. Leung and E. Bossè, "A Maximum Likelihood Approach to Joint Registration, Association and Fusion for Multi-Sensor Multi-Target Tracking", *12th Conference on Information Fusion*, Seattle, WA, USA, July 6-9, 2009.

[17] S. Fortunati , A. Farina, F. Gini, A. Graziano, M. S. Greco, S. Giompapa, "Least Squares Estimation and Cramér-Rao Type Lower Bounds for Relative Sensor

Registration Process," *Signal Processing, IEEE Transactions on*, vol. 59, no. 3, pp. 1075-1087, March 2011.

[18] J. Wu, "On the convergence properties of the EM algorithm", *Ann. Statist.* vol. 11, no. 1, 1983, pp. 95-103.

[19] Y. Noam, H. Messer, "Notes on the Tightness of the Hybrid Cramér–Rao Lower Bound", *Signal Processing, IEEE Transactions on* vol. 57, no. 6, pp. 2074–2084, June 2009.

[20] F. Gini and R. Reggiannini, "On the use of Cramér-Rao-Like bounds in the presence of random nuisance parameters," *Communications, IEEE Transactions on*, vol. 46, no. 7, pp. 2120-2126, December 2000.

[21] M. Pardini, F. Lombardini, F. Gini, "The Hybrid Cramér–Rao Bound on Broadside DOA Estimation of Extended Sources in Presence of Array Errors," *Signal Processing, IEEE Transactions on,* vol. 56, no. 4, pp. 1726 - 1730, April 2008.

[22] T. J. Rothenberg, "Identification in parametric models," *Econometrica*, vol. 39, no. 3, pp. 577-591, May 1971.

[23] R. Bowden, "The theory of parametric identification," *Econometrica*, vol. 41, no. 6, pp. 1069-1074, November 1973.

[24] T. C. Koopmans and O. Reiersol, "The Identification of Structural Characteristics," *The Annals of Mathematical Statistics*, vol. 21, no. 2, pp. 165-181, June 1950.

[25] B. Hochwald and A. Nehorai, "Identifiability in array processing models with vector-sensor applications," *Signal Processing, IEEE Transactions on*, vol. 44, no. 1, pp. 83-95, Jan 1996.

[26] T. Söderström and Stoica P., *System Identification*. London, UK: Prentice-Hall, 1989.

[27] H. Van Trees, *Detection, Estimation and Modulation Theory, Part. 1*. New York: Wiley, 1967, vol. 1.

[28] M. S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[29] F. Gini, R. Reggiannini, and U. Mengali, "The Modified Cramér-Rao bound in vector parameter estimation," *Communications, IEEE Transactions on*, vol. 46, no. 1, pp. 52-60, January 1998.

[30] P. Stoica and T.L. Marzetta, "Parameter estimation problems with singular information matrices," *Signal Processing, IEEE Transactions on*, vol. 49, no. 1, pp. 87-90, Jan 2001.

[31] X. R. Li, V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *Aerospace and Electronic Systems, IEEE Transactions on*, vol.39, no.4, pp. 1333- 1364, Oct. 2003.

[32] M. Lonqbin, S. Xiaoquan, Z. Yiyu, S. Z. Kanq and Y. Bar-Shalom, "Unbiased Converted Measurements for Tracking," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 34, issue 3, pp 1023-1027 July 1998.

[33] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum Likelihood Estimates of Linear Dynamic Systems", *AIAA Journal*, vol. 3, no. 8, pp. 1445-1450, August 1965.

[34] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.

[35] M. Bazaraa, H. Sherali and C. Shetty, *Nonlinear Programming, Theory and Applications*, J. Wiley & Sons, New York, (Third Edition) 2006.

[36] S. Chen, H. Leung and E. Bossè, "A Maximum Likelihood Approach to Joint Registration, Association and Fusion for Multi-Sensor Multi-Target Tracking", *12th International Conference on Information Fusion*, Seattle, WA, USA, July 6-9, 2009.

[37] S. Kullback, *Information Theory and Statistics*. New York: John Wiley, 1959.

[38] D. A. S. Fraser, *The Structure of Inference*. New York: John Wiley, 1968.

[39] S. Boyd and L. Vandenberghe, *Convex Optimization*, 1st ed.: Cambridge University Press, 2004.

[40] G. Marsaglia and G. P. H. Styan, "Equalities and inequalities for ranks of matrices," *Linear and Multilinear Algebra*, vol. 2, no. 3, pp. 269-292, 1974.

Chapter 4: The absolute grid-locking problem

4.1 Introduction

The aim of this Chapter is to extend the EM algorithm, derived in Chapter 3 for the relative grid-locking problem, to the absolute grid-locking problem. For clarity, we recall here the main difference between relative and absolute grid-locking problem. The relative grid-locking process aligns remote data to local data under the assumption that the local data are bias free and that all biases reside with the remote sensor. The problem is that, actually, also the local sensor is affected by biases that cannot be corrected by means of this approach. The absolute grid-locking process assumes that all the sensors in the scenario are affected by errors that must be corrected. First, we start to analyze a simple system composed of two biased radar and a single target. We derive the EM algorithm [1], [2] for the joint estimate of the bias error vector of the two radars and the unknown target state vectors. Then we pass to the general case of absolute grid-locking problem in a scenario with M sensors and N_t targets. In particular, we extend the EM algorithm to obtain the joint estimate of the bias error vector of all M sensors in the system and of the state vectors for all the N_t targets. The Hybrid Cramèr-Rao Lower Bound (HCRLB) [3], [4] [5] is evaluated in order to assess the efficiency of the proposed algorithm. Finally, some preliminary simulation result is provided.

4.2 Absolute grid-locking problem

The geometry of the absolute grid-locking problem for two sensors is shown in Fig. 4.1.

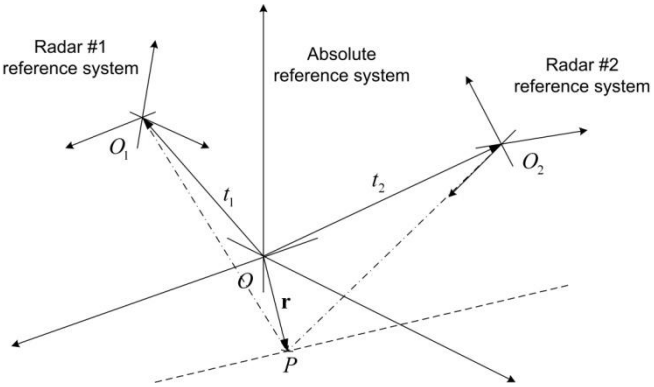

Figure 4.1 – Geometry for the absolute grid-locking problem.

Following the results obtained in Section 3.2, the measurement model of the measures coming from radars #1 and #2 can be expressed as:

$$\begin{aligned}\mathbf{v}_{1,m}^k &= \mathbf{h}^{-1}\left(\mathbf{R}^T\left(\boldsymbol{\Theta}_{1,m}+d\boldsymbol{\Theta}_1\right)\left[\mathbf{r}_k-\left(\mathbf{t}_{1,m}+d\mathbf{t}_1\right)\right]\right)-d\mathbf{v}_1+\mathbf{n}_1^k= \\ &\triangleq \boldsymbol{\mu}\left(\mathbf{r}_k,\boldsymbol{\Phi}_1\right)+\mathbf{n}_1^k,\end{aligned} \tag{4.1}$$

$$\begin{aligned}\mathbf{v}_{2,m}^k &= \mathbf{h}^{-1}\left(\mathbf{R}^T\left(\boldsymbol{\Theta}_{2,m}+d\boldsymbol{\Theta}_2\right)\left[\mathbf{r}_k-\left(\mathbf{t}_{2,m}+d\mathbf{t}_2\right)\right]\right)-d\mathbf{v}_2+\mathbf{n}_2^k= \\ &\triangleq \boldsymbol{\mu}\left(\mathbf{r}_k,\boldsymbol{\Phi}_2\right)+\mathbf{n}_2^k,\end{aligned} \tag{4.2}$$

where $\boldsymbol{\Phi}_1=(d\tilde{\mathbf{v}}_1^T \quad d\tilde{\boldsymbol{\Theta}}_1^T \quad d\mathbf{t}_1^T)^T$ and $\boldsymbol{\Phi}_2=(d\tilde{\mathbf{v}}_2^T \quad d\tilde{\boldsymbol{\Theta}}_2^T \quad d\mathbf{t}_2^T)^T$. As discussed in Section 3.2, if the rotation around *z* axis is applied first, the azimuth measurement bias $d\theta$ and the yaw attitude bias $d\xi$ for both radars cannot be distinguished and have to be merged into a single bias. In order to take this into account, we have defined the vectors $d\tilde{\boldsymbol{\Theta}}_i=\begin{bmatrix} d\chi_i & d\psi_i & d\zeta_i \end{bmatrix}^T$ and $d\tilde{\mathbf{v}}_i=\begin{bmatrix} d\rho_i & 0 & d\varepsilon_i \end{bmatrix}^T$ with i=1,2. In the rest of this Chapter, the unknown parameter vector for the absolute grid-locking problem is assumed to be $\boldsymbol{\Phi}\triangleq[\boldsymbol{\Phi}_1^T \quad \boldsymbol{\Phi}_2^T]^T$. For the target state model, we choose the simple constant velocity model described in Section 3.4.

The joint probability density function of the complete data can be rewritten as:

$$\begin{aligned}p_X\left(X;\boldsymbol{\Phi}\right) &= p_{S,Y}\left(S,Y;\boldsymbol{\Phi}_1,\boldsymbol{\Phi}_2\right)=p_{Y|S}\left(Y|S;\boldsymbol{\Phi}_1,\boldsymbol{\Phi}_2\right)p_S\left(S\right)= \\ &= p_{V_1|S}\left(V_1|S;\boldsymbol{\Phi}_1\right)p_{V_2|S}\left(V_2|S;\boldsymbol{\Phi}_2\right)p_S\left(S\right),\end{aligned} \tag{4.3}$$

The term $p_S(S)$ is given in eq. (3.43). From eq. (4.1), the term $p_{V_1|S}(V_1|S;\boldsymbol{\Phi}_1)$ can be rewritten as:

$$p_{V_1|S}\left(V_1|S;\boldsymbol{\Phi}_1\right)=\prod_{k=1}^{K}p_{\mathbf{v}_{1,m}^k|\mathbf{s}_k}\left(\mathbf{v}_{1,m}^k|\mathbf{s}_k;\boldsymbol{\Phi}_1\right), \tag{4.4}$$

where $\mathbf{v}_{1,m}^k|\mathbf{s}_k \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi}_1),\mathbf{C}_1)$. Finally, form eq. (4.2), the term $p_{V_2|S}(V_2|S;\boldsymbol{\Phi}_2)$ can be evaluated as:

$$p_{V_2|S}\left(V_2|S;\boldsymbol{\Phi}_2\right)=\prod_{k=1}^{K}p_{\mathbf{v}_{2,m}^k|\mathbf{s}_k}\left(\mathbf{v}_{2,m}^k|\mathbf{s}_k;\boldsymbol{\Phi}_2\right), \tag{4.5}$$

where $\mathbf{v}_{2,m}^k|\mathbf{s}_k \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi}_2),\mathbf{C}_2)$ and, as before, $\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi}_i)\triangleq\boldsymbol{\mu}(\mathbf{r}_k,\boldsymbol{\Phi}_i)$, $i=1,2$.

The logarithm of the pdf of the complete data X can be evaluated, ignoring constants, as ([6],[7], [8], [9]):

$$\begin{aligned}\ln p_X\left(X;\boldsymbol{\Phi}\right)&=\ln p_{V_1|S}\left(V_1|S;\boldsymbol{\Phi}_1\right)+\ln p_{V_2|S}\left(V_2|S;\boldsymbol{\Phi}_2\right)+\ln p_S\left(S\right)\\&=-\frac{1}{2}\sum_{k=1}^{K}\left[\mathbf{v}_{1,m}^k-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}_1\right)\right]\mathbf{C}_1^{-1}\left[\mathbf{v}_{1,m}^k-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}_1\right)\right]^T\\&\quad-\frac{1}{2}\sum_{k=1}^{K}\left[\mathbf{v}_{2,m}^k-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}_2\right)\right]\mathbf{C}_2^{-1}\left[\mathbf{v}_{2,m}^k-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}_2\right)\right]^T\\&\quad-\frac{1}{2}\sum_{k=2}^{K}\left[\mathbf{s}_k-\mathbf{F}\mathbf{s}_{k-1}\right]\mathbf{Q}_w^{-1}\left[\mathbf{s}_k-\mathbf{F}\mathbf{s}_{k-1}\right]^T+const.\end{aligned} \tag{4.6}$$

It can be noted that only the first two term of the sum in eq. (4.6) depend on $\boldsymbol{\Phi}$, then, for this reason, the last term can be neglected.

Using the property of the *trace* operator and by defining the column vector $\boldsymbol{\alpha}_k^i$ as

$$\boldsymbol{\alpha}_k^i=\mathbf{v}_{i,m}^k-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}_i\right),\ i=1,2, \tag{4.7}$$

eq. (4.6) can be rewritten as:

$$\ln p_X\left(X;\boldsymbol{\Phi}\right)=-\frac{1}{2}\sum_{i=1}^{2}\mathrm{tr}\left\{\mathbf{C}_i^{-1}\sum_{k=1}^{K}\boldsymbol{\alpha}_k^i(\boldsymbol{\alpha}_k^i)^T\right\}+const. \tag{4.8}$$

At this point, we are able to perform the E-step of the EM algorithm. As discussed in Section 3.6.1, the E-step consists in evaluating the objective function $Q(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n)$, i.e. the

conditional expectation of the logarithm of the pdf of the complete data, $\ln p_X(X;\boldsymbol{\Phi})$, given the incomplete data set *Y* and the current estimate of the parameters vector $\hat{\boldsymbol{\Phi}}^n$:

$$Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right)=-\frac{1}{2}\sum_{i=1}^{2}\mathrm{tr}\left\{\mathbf{C}_i^{-1}\sum_{k=1}^{K}E\left\{\boldsymbol{\alpha}_k^i(\boldsymbol{\alpha}_k^i)^T\left|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\right.\right\}\right\}. \tag{4.9}$$

We define, using the classical notation, the conditional mean and the conditional error covariance matrix as:

$$\hat{\mathbf{s}}_{k|K}^n=E\left\{\mathbf{s}_k\left|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\right.\right\}, \tag{4.10}$$

$$\mathbf{P}_{k|K}^n=E\left\{(\mathbf{s}_k-\hat{\mathbf{s}}_{k|K}^n)(\mathbf{s}_k-\hat{\mathbf{s}}_{k|K}^n)^T\left|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\right.\right\}, \tag{4.11}$$

where $V_1=\{\mathbf{v}_{1,m}^j\}_{j=1}^K$ and $V_2=\{\mathbf{v}_{2,m}^j\}_{j=1}^K$. To evaluate explicitly the terms $\boldsymbol{\alpha}_k^i(\boldsymbol{\alpha}_k^i)^T$, we follow exactly the same procedure as in Section 3.6.2. First, we take the first-order Taylor series expansion of the function $\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi}_i)$ i.e.

$$\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k,\boldsymbol{\Phi}_i\right)\simeq\tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}_i\right)+\mathbf{M}_k^T\left(\boldsymbol{\Phi}_i\right)\left(\mathbf{s}_k-\hat{\mathbf{s}}_{k|K}^n\right), \tag{4.12}$$

where $\mathbf{M}_k(\boldsymbol{\Phi}_i)=\partial\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi}_i)/\partial\mathbf{s}_k\Big|_{\mathbf{s}_k=\hat{\mathbf{s}}_{k|K}^n}$. Now, through direct calculation, it is easy to show that the conditional expectation $E\{\boldsymbol{\alpha}_k^i(\boldsymbol{\alpha}_k^i)^T\left|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\right.\}$ can be evaluated as:

$$\begin{aligned}&E\left\{\boldsymbol{\alpha}_k^i(\boldsymbol{\alpha}_k^i)^T\left|V_1,V_2;\hat{\boldsymbol{\Phi}}^n\right.\right\}\\&\quad=\left(\mathbf{v}_{i,m}^k-\tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}_i\right)\right)\left(\mathbf{v}_{i,m}^k-\tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}_i\right)\right)^T+\mathbf{M}_k^T\left(\boldsymbol{\Phi}_i\right)\mathbf{P}_{k|K}^n\mathbf{M}_k\left(\boldsymbol{\Phi}_i\right).\end{aligned} \tag{4.13}$$

Finally, collecting the previous results, we get:

$$\begin{aligned}Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right)&=-\frac{1}{2}\sum_{i=1}^{2}\mathrm{tr}\left\{\mathbf{C}_i^{-1}\sum_{k=1}^{K}\left(\mathbf{v}_{i,m}^k-\tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}_i\right)\right)\left(\mathbf{v}_{i,m}^k-\tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^n,\boldsymbol{\Phi}_i\right)\right)^T\right\}-\\&\quad-\frac{1}{2}\sum_{i=1}^{2}\mathrm{tr}\left\{\mathbf{C}_i^{-1}\sum_{k=1}^{K}\mathbf{M}_k^T\left(\boldsymbol{\Phi}_i\right)\mathbf{P}_{k|K}^n\mathbf{M}_k\left(\boldsymbol{\Phi}_i\right)\right\}\\&=\sum_{i=1}^{2}Q_i\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right),\end{aligned} \tag{4.14}$$

where $\hat{\mathbf{s}}_{k|K}^{n}$ and $\mathbf{P}_{k|K}^{n}$ are the smoothed state vector estimate and the smoothed error covariance matrix [10]. At this point, the M-step can be performed as:

$$\hat{\mathbf{\Phi}}^{n+1} = \arg\max_{\mathbf{\Phi}\in\Omega} \left\{ Q\left(\mathbf{\Phi}; \hat{\mathbf{\Phi}}^{n}\right) \right\}. \tag{4.15}$$

A close-form solution for the optimization problem in eq. (4.15) is unfeasible due to the analytical complexity of the objective function $Q(\mathbf{\Phi}; \hat{\mathbf{\Phi}}^{n})$ then we use a numerical minimization algorithm. In particular we have used the SQP method [11], [12]. The solution of the optimization problem in eq. (4.15) belongs to a subspace of a 16-dimentional Euclidean space. However, it is possible to reduce the 16-dimensional problem in eq. (4.15) in two 8-dimensional problems. By denoting with $\nabla_{\mathbf{\Phi}} = [\nabla_{\mathbf{\Phi}_1} \quad \nabla_{\mathbf{\Phi}_2}]$ the gradient operator vector, it is easy to show that:

$$\nabla_{\mathbf{\Phi}} Q\left(\mathbf{\Phi}; \hat{\mathbf{\Phi}}^{n}\right)\Big|_{\mathbf{\Phi}=\hat{\mathbf{\Phi}}^{n+1}} = \mathbf{0} \Leftrightarrow \nabla_{\mathbf{\Phi}_i} Q_i\left(\mathbf{\Phi}; \hat{\mathbf{\Phi}}^{n}\right)\Big|_{\mathbf{\Phi}_i=\hat{\mathbf{\Phi}}_i^{n+1}}, i = 1,2, \tag{4.16}$$

since

$$\nabla_{\mathbf{\Phi}_i} Q_j\left(\mathbf{\Phi}; \hat{\mathbf{\Phi}}^{n}\right) \equiv \mathbf{0}, i \neq j \tag{4.17}$$

Then, the two 8-dimensional sub-problem are exactly:

$$\hat{\mathbf{\Phi}}_i^{n+1} = \arg\max_{\mathbf{\Phi}\in\Omega_i} \left\{ Q_i\left(\mathbf{\Phi}; \hat{\mathbf{\Phi}}^{n}\right) \right\}, i = 1,2, \tag{4.18}$$

and, finally, the estimated absolute parameter vector is given by:

$$\hat{\mathbf{\Phi}}^{n+1} = \left[\left(\hat{\mathbf{\Phi}}_1^{n+1}\right)^T \quad \left(\hat{\mathbf{\Phi}}_2^{n+1}\right)^T \right]^T. \tag{4.19}$$

4.2.1 Linear Least Squares estimator for the absolute grid-locking problem

In this section, we provide a generalization of the linear Least Squares (LS) estimator derived in Section 3.5 to the case of the absolute grid-locking problem. The estimate of the unknown parameter vector given by the linear LS algorithm could be used as starting point

for the iterations of the EM algorithm. The fundamental alignment equation for the relative grid-locking problem can be easily extended to the case of absolute grid-locking problem as:

$$E\left\{\mathbf{R}\left(\boldsymbol{\Theta}_{1,m}+d\boldsymbol{\Theta}_1\right)\mathbf{h}\left(\mathbf{v}_{1,m}^k+d\mathbf{v}_1\right)+\left(\mathbf{t}_{1,m}+d\mathbf{t}_1\right)-\right.$$
$$\left.-\mathbf{R}\left(\boldsymbol{\Theta}_{2,m}+d\boldsymbol{\Theta}_2\right)\mathbf{h}\left(\mathbf{v}_{2,m}^k+d\mathbf{v}\right)-\left(\mathbf{t}_{2,m}+d\mathbf{t}_2\right)\right\}=\mathbf{0}. \tag{4.20}$$

Therefore, eq. (4.20) can be expressed as:

$$\mathbf{R}\left(\boldsymbol{\Theta}_{1,m}+d\boldsymbol{\Theta}_1\right)\mathbf{h}\left(\mathbf{v}_{1,m}^k+d\mathbf{v}_1\right)+\left(\mathbf{t}_{1,m}+d\mathbf{t}_1\right)$$
$$=\mathbf{R}\left(\boldsymbol{\Theta}_{2,m}+d\boldsymbol{\Theta}_2\right)\mathbf{h}\left(\mathbf{v}_{2,m}^k+d\mathbf{v}\right)+\left(\mathbf{t}_{2,m}+d\mathbf{t}_2\right)+\boldsymbol{\varepsilon}, \tag{4.21}$$

where $\boldsymbol{\varepsilon}$ represents the un-modelled zero mean error. The non-linear (NLLS) estimate of the absolute unknown parameter vector $\boldsymbol{\Phi}$ can be obtained by minimizing the objective function:

$$J(\boldsymbol{\Phi})=\left\|\mathbf{R}\left(\boldsymbol{\Theta}_{1,m}+d\boldsymbol{\Theta}_1\right)\mathbf{h}\left(\mathbf{v}_{1,m}^k+d\mathbf{v}_1\right)+\left(\mathbf{t}_{1,m}+d\mathbf{t}_1\right)-\right.$$
$$\left.-\mathbf{R}\left(\boldsymbol{\Theta}_{2,m}+d\boldsymbol{\Theta}_2\right)\mathbf{h}\left(\mathbf{v}_{2,m}^k+d\mathbf{v}_2\right)-\left(\mathbf{t}_{2,m}+d\mathbf{t}_2\right)\right\|_2^2 \tag{4.22}$$

with respect to $\boldsymbol{\Phi}$. Since we are interesting only to obtain a starting point for the EM iterations, we resort here to a linear least squares algorithm. To this purpose, we need to linearize the alignment equation in (4.21). Using the same procedure of [13], we get:

$$\mathbf{R}\left(\boldsymbol{\Theta}_{1,m}\right)\mathbf{h}\left(\mathbf{v}_{1,m}^k\right)+\mathbf{t}_{1,m}+\mathbf{R}\left(\boldsymbol{\Theta}_{1,m}\right)d\mathbf{r}_1^k+d\mathbf{R}_1\mathbf{h}\left(\mathbf{v}_{1,m}^k\right)+d\mathbf{t}_1=$$
$$=\mathbf{R}\left(\boldsymbol{\Theta}_{2,m}\right)\mathbf{h}\left(\mathbf{v}_{2,m}^k\right)+\mathbf{t}_{2,m}+\mathbf{R}\left(\boldsymbol{\Theta}_m\right)d\mathbf{r}_2^k+d\mathbf{R}_2\mathbf{h}\left(\mathbf{v}_{2,m}^k\right)+d\mathbf{t}_2+\boldsymbol{\varepsilon}, \tag{4.23}$$

where $d\mathbf{r}_i^k$ and $d\mathbf{R}_i$ for $i=1,2$ are defined in [13].

Eq. (4.23) can now be recast in a compact form, useful for the direct application of the linear LS algorithm. We can define a "new" measurement vector $\mathbf{z}^k$, that is a function of all available measurements, as:

$$\mathbf{z}^k \triangleq \mathbf{R}\left(\boldsymbol{\Theta}_{1,m}\right)\mathbf{h}\left(\mathbf{v}_{1,m}^k\right)+\mathbf{t}_{1,m}-\mathbf{R}\left(\boldsymbol{\Theta}_{2,m}\right)\mathbf{h}\left(\mathbf{v}_{2,m}^k\right)-\mathbf{t}_{2,m}. \tag{4.24}$$

Unfortunately, it is clear from eqs. (4.23) and (4.24) that the two position error vectors, $d\mathbf{t}_1$ and $d\mathbf{t}_2$, cannot be estimated separately but only their linear combination can be

obtained by means of this linearized approach. For this reason, we define a reduced unknown parameter vector as $\tilde{\mathbf{\Phi}} \triangleq [\tilde{\mathbf{\Phi}}_1^T \quad \tilde{\mathbf{\Phi}}_2^T]^T$ where $\tilde{\mathbf{\Phi}}_i = (d\tilde{\mathbf{v}}_i^T \quad d\tilde{\mathbf{\Theta}}_i^T)^T$ for i=1,2.

The others terms in eq. (4.23) can be rewritten in matrix form as $\mathbf{H}^k\mathbf{\Phi}$ where $\mathbf{H}^k$ is a block matrix expressed as $\mathbf{H}^k = [-\mathbf{H}_1^k \quad \mathbf{H}_2^k]$:

$$\begin{aligned}\mathbf{H}_i^k = \Bigg(& \mathbf{R}\left(\mathbf{\Theta}_{i,m}\right)\frac{\partial \mathbf{h}\left(\mathbf{v}_{i,m}^k\right)}{\partial \rho} \quad \mathbf{R}\left(\mathbf{\Theta}_{i,m}\right)\frac{\partial \mathbf{h}\left(\mathbf{v}_{i,m}^k\right)}{\partial \varepsilon}\times \\ & \times \mathbf{l}_\chi\left(\mathbf{\Theta}_{i,m}, \mathbf{v}_{i,m}^k\right) \quad \mathbf{l}_\psi\left(\mathbf{\Theta}_{i,m}, \mathbf{v}_{i,m}^k\right) \quad \mathbf{l}_\xi\left(\mathbf{\Theta}_{i,m}, \mathbf{v}_{i,m}^k\right)\Bigg),\end{aligned} \tag{4.25}$$

and i=1,2. If we assume that K measurements coming from both radars #1 and #2 are available, the LS problem can be cast in the following well-known form:

$$J_L\left(\tilde{\mathbf{\Phi}}\right) = \sum_{k=1}^{K} \left\| \mathbf{z}^k - \mathbf{H}^k\tilde{\mathbf{\Phi}} \right\|_2^2, \tag{4.26}$$

where $J_L(\tilde{\mathbf{\Phi}})$ is the linearized objective function. The LS estimate $\hat{\tilde{\mathbf{\Phi}}}$ is obtained by minimizing $J_L(\tilde{\mathbf{\Phi}})$. By defining the vector $\tilde{\mathbf{z}}$ and the matrix $\tilde{\mathbf{H}}$ as:

$$\tilde{\mathbf{z}} = \left[\left(\mathbf{z}^1\right)^T \quad \ldots \quad \left(\mathbf{z}^k\right)^T \quad \ldots \quad \left(\mathbf{z}^K\right)^T\right]^T, \tag{4.27}$$

$$\tilde{\mathbf{H}} = \left[\left(\mathbf{H}^1\right)^T \quad \ldots \quad \left(\mathbf{H}^k\right)^T \quad \ldots \quad \left(\mathbf{H}^K\right)^T\right]^T, \tag{4.28}$$

we can express the LS estimate of $\tilde{\mathbf{\Phi}}$ as:

$$\hat{\tilde{\mathbf{\Phi}}} = \tilde{\mathbf{H}}^{\#}\tilde{\mathbf{z}}, \tag{4.29}$$

where $\tilde{\mathbf{H}}^{\#}$ is the pseudo-inverse matrix of $\tilde{\mathbf{H}}$.

4.3 The absolute grid-locking problem in the multi-sensor-multi-target scenario

In this section, we provide the solution for the general case of the absolute grid-locking problem in a multi-sensor-multi-target scenario. In the following, we suppose to have M_r radars and N_t targets. As before, each one of the N_t targets is supposed to follow the linear stochastic model given in eq. (3.87):

$$\mathbf{s}_{k+1}^{l} = \mathbf{F}\mathbf{s}_{k}^{l} + \mathbf{w}_{k}^{l}, \quad l = 1,\ldots,N_t,$$

where the index l define the particular target. It's important to note that the discrete process noise vector $\mathbf{w}_k^{\,l}$ as function of the index l is an independent, zero-mean, Gaussian distributed, random vector. Starting from the previous discussion about the absolute grid-locking for two radars, it easy to show that the measurement model for each considered sensor can be expressed as:

$$\begin{aligned}\mathbf{v}_{i,m}^{k,l} &= \mathbf{h}^{-1}\left(\mathbf{R}^{T}\left(\boldsymbol{\Theta}_{i,m} + d\boldsymbol{\Theta}_{i}\right)\left[\mathbf{r}_{k}^{l} - \left(\mathbf{t}_{i,m} + d\mathbf{t}_{i}\right)\right]\right) - d\mathbf{v}_{i} + \mathbf{n}_{i}^{k} = \\ &\triangleq \boldsymbol{\mu}\left(\mathbf{r}_{k}^{l}, \boldsymbol{\Phi}_{i}\right) + \mathbf{n}_{i}^{k}, \quad i = 1,\ldots,M_r.\end{aligned} \tag{4.30}$$

Under these assumptions, the set of complete data $\tilde{X}$ and the set of incomplete data $\tilde{Y}$ can be defined as:

$$\tilde{X} = \left(\tilde{V}_1 \quad \cdots \quad \tilde{V}_{M_r} \quad \tilde{S}\right), \quad \tilde{Y} = \left(\tilde{V}_1 \quad \cdots \quad \tilde{V}_{M_r}\right), \tag{4.31}$$

where:

$$\tilde{V}_i = \left\{V_i^{l}\right\}_{l=1}^{N_t}, i = 1,\ldots,M_r \quad \tilde{S} = \left\{S^{l}\right\}_{l=1}^{N_t}, \tag{4.32}$$

and

$$V_i^{l} = \left\{\mathbf{v}_{i,m}^{k,l}\right\}_{k=1}^{K}, \quad S^{l} = \left\{\mathbf{s}_{k}^{l}\right\}_{k=1}^{K}. \tag{4.33}$$

The unknown parameter vector is defined as $\boldsymbol{\Phi} \triangleq [\boldsymbol{\Phi}_1^T \quad \cdots \quad \boldsymbol{\Phi}_{M_r}^T]^T$. To apply the EM algorithm to this multi-sensor-multi-target scenario, first we have to calculate the joint pdf of the complete data $\tilde{X}$, $p_{\tilde{X}}(\tilde{X};\boldsymbol{\Phi})$. Following the procedure described in Section 3.9, we have:

$$
\begin{aligned}
p_{\tilde{X}}\left(\tilde{X};\boldsymbol{\Phi}\right) &= p_{\tilde{S},\tilde{Y}}\left(\tilde{S},\tilde{Y};\boldsymbol{\Phi}\right) \\
&= p_{\tilde{S}}\left(\tilde{S}\right)\prod_{i=1}^{M_r} p_{\tilde{V}_i|\tilde{S}}\left(\tilde{V}_i\left|\tilde{S};\boldsymbol{\Phi}_i\right.\right).
\end{aligned}
\tag{4.34}
$$

The terms $p_{\tilde{S}}(\tilde{S})$ and $p_{\tilde{V}_i|\tilde{S}}(\tilde{V}_i\left|\tilde{S};\boldsymbol{\Phi}_i\right.)$ can be evaluated exactly as shown in Section 3.9, then the logarithm of the pdf of the complete data $\tilde{X}$ can be expressed as:

$$
\ln p_{\tilde{X}}\left(\tilde{X};\boldsymbol{\Phi}\right) = -\frac{1}{2}\sum_{i=1}^{M_r}\sum_{l=1}^{N_t}\sum_{k=1}^{K}\left[\mathbf{v}_{i,m}^{k,l}-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k^l,\boldsymbol{\Phi}_i\right)\right]\mathbf{C}_i^{-1}\left[\mathbf{v}_{i,m}^{k,l}-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k^l,\boldsymbol{\Phi}_i\right)\right]^T + const. \tag{4.35}
$$

Now, by defining the vector $\boldsymbol{\alpha}_{k,l}^i = \mathbf{v}_{i,m}^{k,l}-\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k^l,\boldsymbol{\Phi}_i\right)$, the objective function $Q(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n)$, i.e. the conditional expectation of $\ln p_{\tilde{X}}(\tilde{X};\boldsymbol{\Phi})$ (neglecting constant terms w.r.t $\boldsymbol{\Phi}$), given the incomplete data set $\tilde{Y}$ and the current estimate of the parameters vector $\hat{\boldsymbol{\Phi}}^n$, can be evaluated as:

$$
\begin{aligned}
Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right) &= -\frac{1}{2}\sum_{l=1}^{N_t}\sum_{i=1}^{M_r}\mathrm{tr}\left\{\mathbf{C}_i^{-1}\sum_{k=1}^{K}E\left\{\boldsymbol{\alpha}_{k,l}^i(\boldsymbol{\alpha}_{k,l}^i)^T\left|V_1^l,\ldots,V_i^l;\hat{\boldsymbol{\Phi}}^n\right.\right\}\right\} \\
&= \sum_{l=1}^{N_t}\sum_{i=1}^{M_r}Q_i^l\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right),
\end{aligned}
\tag{4.36}
$$

where $Q_i^l(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n)$ is the objective function for a given target and for a given sensor defined as:

$$
\begin{aligned}
Q_i^l\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right) = &-\frac{1}{2}\mathrm{tr}\left\{\mathbf{C}_i^{-1}\sum_{k=1}^{K}\left(\mathbf{v}_{i,m}^{k,l}-\tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^{n,l},\boldsymbol{\Phi}_i\right)\right)\left(\mathbf{v}_{i,m}^{k,l}-\tilde{\boldsymbol{\mu}}\left(\hat{\mathbf{s}}_{k|K}^{n,l},\boldsymbol{\Phi}_i\right)\right)^T\right\}- \\
&-\frac{1}{2}\mathrm{tr}\left\{\mathbf{C}_i^{-1}\sum_{k=1}^{K}\mathbf{M}_k^T\left(\boldsymbol{\Phi}_i\right)\mathbf{P}_{k|K}^{n,l}\mathbf{M}_k\left(\boldsymbol{\Phi}_i\right)\right\}.
\end{aligned}
\tag{4.37}
$$

where, as before, the first-order Taylor series expansion of the function $\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi}_i)$ is used.

The M-step can be performed as:

$$
\hat{\boldsymbol{\Phi}}^{n+1} = \arg\max_{\boldsymbol{\Phi}\in\Omega}\left\{Q\left(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}^n\right)\right\}. \tag{4.38}
$$

Also in this case, the optimization problem is solved using the SQP numerical minimization algorithm. The solution of the optimization problem in eq. (4.38) belongs to a subspace of a $8M_r$-dimentional Euclidean space. However, it is possible to reduce the $8M_r$-dimensional

problem in M_r 8-dimensional problems. By denoting with $\nabla_{\mathbf{\Phi}} = [\nabla_{\mathbf{\Phi}_1} \quad \cdots \quad \nabla_{\mathbf{\Phi}_{M_r}}]$ the gradient operator vector, it is easy to show that:

$$\nabla_{\mathbf{\Phi}} Q\left(\mathbf{\Phi};\hat{\mathbf{\Phi}}^n\right)\Big|_{\mathbf{\Phi}=\hat{\mathbf{\Phi}}^{n+1}} = \mathbf{0} \Leftrightarrow \nabla_{\mathbf{\Phi}_i} \sum_{l=1}^{N_t} Q_i^l\left(\mathbf{\Phi};\hat{\mathbf{\Phi}}^n\right)\Big|_{\mathbf{\Phi}_i=\hat{\mathbf{\Phi}}_i^{n+1}}, i = 1,\ldots,M_r, \tag{4.39}$$

since

$$\nabla_{\mathbf{\Phi}_i} \sum_{l=1}^{N_t} Q_i^l\left(\mathbf{\Phi};\hat{\mathbf{\Phi}}^n\right) \equiv \mathbf{0}, i \neq j. \tag{4.40}$$

Then, the M_r 8-dimensional sub-problem are given by:

$$\hat{\mathbf{\Phi}}_i^{n+1} = \arg\max_{\mathbf{\Phi}\in\Omega_i} \left\{ \sum_{l=1}^{N_t} Q_i^l\left(\mathbf{\Phi};\hat{\mathbf{\Phi}}^n\right) \right\}, i = 1,\ldots,M_r \tag{4.41}$$

and, finally, the estimated absolute parameter vector is given by:

$$\hat{\mathbf{\Phi}}^{n+1} = \left[\left(\hat{\mathbf{\Phi}}_1^{n+1}\right)^T \quad \cdots \quad \left(\hat{\mathbf{\Phi}}_{M_r}^{n+1}\right)^T \right]^T. \tag{4.42}$$

4.4 Performance bound for multi-sensor-multi-target scenario

In this section, we evaluate the HCRLB [3], [4], [5] for the general case of multi-sensor-multi-target scenario. First of all, we have to define a new hybrid parameter vector as:

$$\tilde{\mathbf{\Psi}} = \left[\mathbf{\Phi}_1^T,\ldots,\mathbf{\Phi}_{M_r}^T, (\mathbf{s}_1^1)^T,\ldots,(\mathbf{s}_K^1)^T,\ldots,(\mathbf{s}_1^{N_t})^T,\ldots,(\mathbf{s}_K^{N_t})^T \right]^T, \tag{4.43}$$

then, following the same procedure described in Section 3, the HFIM for this hybrid parameter estimation problem can be expressed as:

$$\begin{aligned} \left[\mathbf{I}_H\left(\tilde{\mathbf{\Psi}}\right)\right]_{ij} &= -E_{\tilde{V}_1,\ldots,\tilde{V}_{M_r},\tilde{S}} \left\{ \frac{\partial^2 \ln p\left(\tilde{V}_1,\ldots,\tilde{V}_{M_r},\tilde{S};\mathbf{\Phi}\right)}{\partial\tilde{\Psi}_i \partial\tilde{\Psi}_j} \right\} \\ &= -E_{\tilde{V}_1,\ldots,\tilde{V}_{M_r},\tilde{S}} \left\{ \frac{\partial^2 \ln\left(p\left(\tilde{S}\right)\prod_{v=1}^{M_r} p\left(\tilde{V}_v\middle|\tilde{S};\mathbf{\Phi}_v\right)\right)}{\partial\tilde{\Psi}_i \partial\tilde{\Psi}_j} \right\}. \end{aligned} \tag{4.44}$$

The logarithm of the conditional pdf of the measurements coming from a given sensor *i* is:

$$\ln p\left(\tilde{V}_i \left| \tilde{S};\boldsymbol{\Phi}_i \right.\right) = \sum_{l=1}^{N_t}\sum_{k=1}^{K} \ln p_{\mathbf{v}_{i,m}^{k,l}\left|\mathbf{s}_k^l\right.}\left(\mathbf{v}_{i,m}^{k,l}\left|\mathbf{s}_k^l;\boldsymbol{\Phi}_i\right.\right), \tag{4.45}$$

where $\mathbf{v}_{i,m}^{k,l}\left|\mathbf{s}_k^l \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k^l,\boldsymbol{\Phi}_i),\mathbf{C}_i)\right.$ while, as shown in eq. (3.102), the logarithm of the joint pdf of all the target state vector is given by:

$$\ln p\left(\tilde{S}\right) = \sum_{l=1}^{N_t} \ln p\left(\mathbf{s}_1^l\right) + \sum_{l=1}^{N_t}\sum_{k=2}^{K} \ln p_{\mathbf{s}_k^l\left|\mathbf{s}_{k-1}^l\right.}\left(\mathbf{s}_k^l\left|\mathbf{s}_{k-1}^l\right.\right), \tag{4.46}$$

where $\mathbf{s}_1^l \sim \mathcal{N}(\boldsymbol{\mu}_0^l,\mathbf{Q}_w^l)$ and $\mathbf{s}_k^l\left|\mathbf{s}_{k-1}^l \sim \mathcal{N}(\mathbf{F}\mathbf{s}_{k-1}^l,\mathbf{Q}_w^l)\right.$. As before, the Hybrid FIM can be rewritten as:

$$\left[\mathbf{I}_H\left(\tilde{\boldsymbol{\Psi}}\right)\right]_{ij} = E_S\left\{\mathbf{I}_{H\left|\tilde{S}\right.}\left(\tilde{\boldsymbol{\Psi}}\right)\right\}, \tag{4.47}$$

where the conditional Hybrid FIM can be calculated as:

$$\begin{aligned}\left[\mathbf{I}_{H\left|\tilde{S}\right.}\left(\boldsymbol{\Psi}\right)\right]_{ij} &= -E_{\tilde{V}_1,\ldots,\tilde{V}_{M_r},\tilde{S}}\left\{\frac{\partial^2 \ln p\left(\tilde{V}_1,\ldots,\tilde{V}_{M_r},\tilde{S};\boldsymbol{\Phi}\right)}{\partial\tilde{\Psi}_i\partial\tilde{\Psi}_j}\right\} = \\ &= \sum_{l=1}^{N_t}\left[\sum_{v=1}^{M_r}\sum_{k=1}^{K} g_{ij}^{v,l}\left(\mathbf{s}_k^l;\boldsymbol{\Phi}\right) + u_{ij}^l\left(\mathbf{s}_k^l\right)\right],\end{aligned} \tag{4.48}$$

where:

$$g_{ij}^{v,l}\left(\mathbf{s}_k^l;\boldsymbol{\Phi}\right) \triangleq -E_{\mathbf{v}_{v,m}^{k,l}\left|\mathbf{s}_k^l\right.}\left\{\frac{\partial^2 \ln p\left(\mathbf{v}_{v,m}^{k,l}\left|\mathbf{s}_k^l;\boldsymbol{\Phi}_v\right.\right)}{\partial\tilde{\Psi}_i\partial\tilde{\Psi}_j}\right\}, \tag{4.49}$$

$$u_{ij}^l\left(\mathbf{s}_k^l\right) \triangleq -\frac{\partial^2 \ln p\left(\mathbf{s}_1^l\right)}{\partial\tilde{\Psi}_i\partial\tilde{\Psi}_j} - \sum_{k=2}^{K}\frac{\partial^2 \ln p\left(\mathbf{s}_k^l\left|\mathbf{s}_{k-1}^l\right.\right)}{\partial\tilde{\Psi}_i\partial\tilde{\Psi}_j}. \tag{4.50}$$

It can be noted that the eqs. (4.49) and (4.50) are exactly the same of the eqs. (3.68) and (3.69), then they can be evaluated as described in Section 3.7 and in Appendix D. The block-matrix form of the conditional HFIM defined in eq. (4.48) is:

$$\mathbf{I}_{H|\tilde{S}}\left(\tilde{\boldsymbol{\Psi}}\right)=\begin{pmatrix}\sum_{l=1}^{N_t}\tilde{\mathbf{F}}_l & \tilde{\mathbf{B}}_1 & \tilde{\mathbf{B}}_2 & \cdots & \tilde{\mathbf{B}}_{N_t}\\ \mathbf{B}_1^T & \mathbf{T}_1 & & \cdots & \mathbf{0}_{6K\times 6K}\\ \mathbf{B}_2^T & & \ddots & & \vdots\\ \vdots & \vdots & & \ddots & \vdots\\ \mathbf{B}_{N_t}^T & \mathbf{0}_{6K\times 6K} & \cdots & & \mathbf{T}_{N_t}\end{pmatrix}, \tag{4.51}$$

where

$$\tilde{\mathbf{F}}_l=\mathrm{diag}\left(\sum_{k=1}^{K}\mathbf{F}_{k,l}^1,\ldots,\sum_{k=1}^{K}\mathbf{F}_{k,l}^{M_r}\right), \tag{4.52}$$

$$\tilde{\mathbf{B}}_l=\begin{pmatrix}\mathbf{B}_{1,l}^1 & \mathbf{B}_{2,l}^1 & \cdots & \mathbf{B}_{K,l}^1\\ \vdots & \vdots & \vdots & \vdots\\ \mathbf{B}_{1,l}^{M_r} & \mathbf{B}_{2,l}^{M_r} & \cdots & \mathbf{B}_{K,l}^{M_r}\end{pmatrix}, \tag{4.53}$$

$$\mathbf{T}_l=\begin{pmatrix}\boldsymbol{\Pi}_{1,l} & \boldsymbol{\Lambda}_l & \cdots & \mathbf{0}_{6\times 6}\\ \boldsymbol{\Lambda}_l^T & \boldsymbol{\Pi}_{2,l} & \ddots & \vdots\\ \vdots & \ddots & \ddots & \boldsymbol{\Lambda}_l\\ \mathbf{0}_{6\times 6} & \cdots & \boldsymbol{\Lambda}_l^T & \boldsymbol{\Xi}_l\end{pmatrix}, \tag{4.54}$$

and

$$\boldsymbol{\Lambda}^l=-\mathbf{F}^T(\mathbf{Q}_w^l)^{-1},\quad \boldsymbol{\Pi}_k^l=\sum_{i=1}^{M_r}\mathbf{M}_{k,l}^i+\boldsymbol{\Omega}^l,\quad \boldsymbol{\Xi}^l=\sum_{i=1}^{M_r}\mathbf{M}_{K,l}^i+(\mathbf{Q}_w^l)^{-1}. \tag{4.55}$$

It must be noted that all the matrices $\mathbf{F}_{k,l}^i$, $\mathbf{M}_{k,l}^i$, $\mathbf{B}_{k,l}^i$ and $\boldsymbol{\Omega}^l$ are defined, for a given l and for a given i, in Section 3.7 and the explicit calculation can be found in Appendix D. Also in this case, the mean value w.r.t. S in eq. (4.47) is evaluated through independent Monte Carlo trials.

Finally, some consideration about the identifiability problem in the absolute grid-locking problem can be provided. In particular, the Theorem 2 given in Section 3.10.3 can be directly applied to the Hybrid FIM of eq. (4.47). With the same procedure used for the relative grid-locking problem and described in Section 3.10.4, it can be proved that the unknown parameter vector $\boldsymbol{\Phi}\triangleq[\boldsymbol{\Phi}_1^T\ \cdots\ \boldsymbol{\Phi}_{M_r}^T]^T$ is identifiable.

4.5 Simulation results

In this section, some preliminary simulation results are reported. Before introducing the simulation scenario, some comment on the initialization of the EM algorithm need to be done. As in the case of relative grid-locking problem, we use the linear LS estimate of the unknown parameter vector, given in eq. (4.29), as starting point for the EM algorithm. Nevertheless, the LS algorithm, described in Section 4.2.1, are not able to estimate the relative position errors vectors $d\mathbf{t}_i$. For this reason in our simulations we consider, for the i^{th} biased radar, only the measurement error vector $d\mathbf{v}_i$ and the attitude error vector $d\boldsymbol{\Theta}_i$. The generalization of the initialization procedure to the case of a full bias error vector that includes also the relative position error vector is left to future investigation.

In our simulation, we assume a scenario with three biased radars and five targets (see Fig. 4.2). Two of these five targets are cooperative targets, i.e. targets whose trajectory are perfectly know. In the following, we give all the geometrical details of the assumed scenario.

- The position vector of the radar #1 is $\mathbf{t}_{t,1} = (-2,4,1)\cdot 10^3$ m.
- The position vector of the radar #2 is $\mathbf{t}_{t,1} = (2,2,2)\cdot 10^3$ m.
- The position vector of the radar #2 is $\mathbf{t}_{t,1} = (-2,-4,-2)\cdot 10^3$ m.

In all the simulations, the following parameters have been used:

- The actual bias errors values for both radar #1 and radar #2 are: i) measurement biases: *i*) measurement biases: $d\rho$=-10m, $d\theta$=$d\varepsilon$=-0.0573°; *ii*) attitude biases: $d\chi$=$d\psi$=-0.0573° and $d\xi$=-0.1146°.
- Radars #1, #2 and #3 are characterized by the following accuracies: *i*) $\sigma_{\rho,1}$=$\sigma_{\rho,2}$=50 m; *ii*) $\sigma_{\theta,i}$=$\sigma_{\varepsilon,i}$=0.3°, i=1,2,3.
- Scan time of the radars: 1 sec.
- The tolerance ε, i. e. absolute value of the difference between the objectivefunction evaluated at the current estimate of the parameter vector and the objective function evaluated at the previous estimate: $|Q(\hat{\boldsymbol{\Phi}}^{n+1}) - Q(\hat{\boldsymbol{\Phi}}^{n})| \geq \varepsilon$: ε=10^{-6}.

- The number of independent Monte Carlo trials: MC=100.
- Searching volume Ω for the optimization algorithm, i. e. the region of $\mathbb{R}^{d_{\mathbf{\Phi}}}$ which the solution is supposed to belong to. We describe this volume whit the following set of inequality $\mathbf{a} \leq \mathbf{\Phi} \leq -\mathbf{a}$: where **a**=(-100m, -5.7296°, -5.7296°,-5.7296°,-5.7296°).
- The maximum number of iteration of the EM algorithm: 10.

We assume to have five targets that move according to the stochastic discrete random model in eq. (3.87) with the following parameters:

- Target #1:

  1) Initial position and velocity vectors: $\mathbf{r}_0{}^1$=[0 2000 0]^T, $\mathbf{v}_0{}^1$=[7.63 -5 1]^T

  2) The PSD of the three components of the continuous-time process noise vector is $N_{0,x}^1 = N_{0,y}^1 = N_{0,z}^1 = q_1 = 0.01\,\mathrm{m}^2/\mathrm{s}^3$

- Target #2:

  1) Initial position vector: $\mathbf{r}_0{}^2$=[-6000 0 0]^T, $\mathbf{v}_0{}^2$=[-4.63 7 -1]^T

  2) The PSD of the three components of the continuous-time process noise vector is equal to $N_{0,x}^2 = N_{0,y}^2 = N_{0,z}^2 = q_2 = 0.01\,\mathrm{m}^2/\mathrm{s}^3$.

- Target #3:

  1) Initial position vector: $\mathbf{r}_0{}^3$=[2000 -1500 0]^T, $\mathbf{v}_0{}^3$=[8 5 0.5]^T

  2) The PSD of the three components of the continuous-time process noise vector is equal to $N_{0,x}^3 = N_{0,y}^3 = N_{0,z}^3 = q_3 = 10^{-15}\ \mathrm{m}^2/\mathrm{s}^3$.

- Target #4:

  1) Initial position vector: $\mathbf{r}_0{}^4$=[-4000 -4000 0]^T, $\mathbf{v}_0{}^4$=[-6 3 2]^T

  2) The PSD of the three components of the continuous-time process noise vector is equal to $N_{0,x}^4 = N_{0,y}^4 = N_{0,z}^4 = q_4 = 10^{-15}\ \mathrm{m}^2/\mathrm{s}^3$.

- Target #5:

  1) Initial position vector: $\mathbf{r}_0{}^5$=[6000 6000 0]^T, $\mathbf{v}_0{}^5$=[-6 4 -2.5]^T

2) The PSD of the three components of the continuous-time process noise vector is equal to $N_{0,x}^5 = N_{0,y}^5 = N_{0,z}^5 = q_5 = 0.01\,\mathrm{m}^2/\mathrm{s}^3$.

It can be noted that the two cooperative targets are Target #3 and Target #4, since their process noise variance is almost 0.

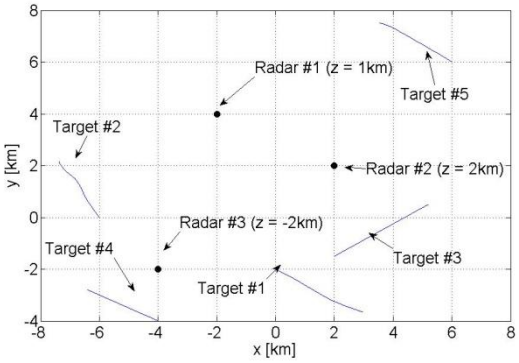

Figure 4.2 – Geometry of the assumed scenario.

As in the previous Chapters, the performance of the EM algorithm is evaluated in terms of error mean value e standard deviation (std) of the estimation error for each parameter to be estimated (see figs. 4.3-4.12). From the numerical results that we derived, some general consideration can be drawn:

- Except for the estimate of the joint (yaw+azimuth) error (see fig. 4.11), all the other estimates result to be biased. This can be due to the low number of iterations allowed to the EM algorithm. However, a larger number of iterations tends to increase dramatically the computational time making the use of the EM algorithm unfeasible.
- Since most of the estimate are biased, nothing can be said about efficiency of the proposed EM algorithm. Also the strange progress of the std curves could be due to a low number of iteration.

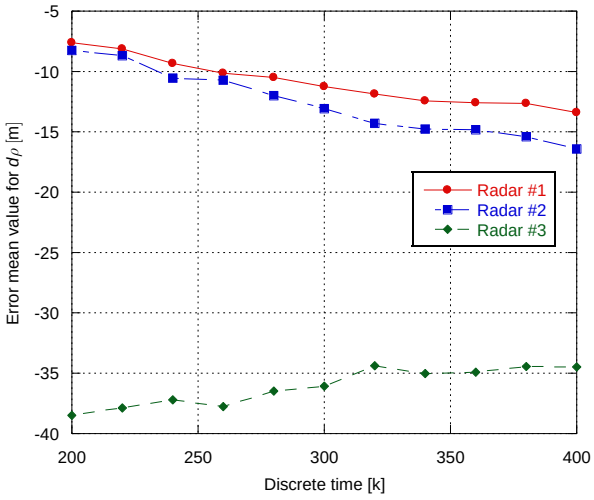

Figure 4.3 – Error mean value for the estimate of the range error.

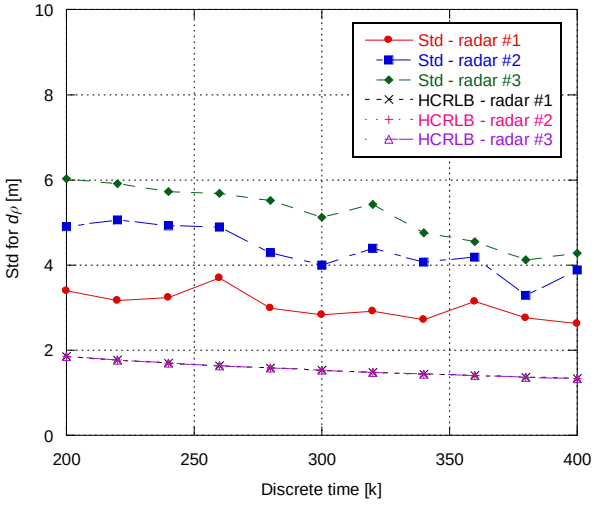

Figure 4.4 – Std for the estimate of the range error.

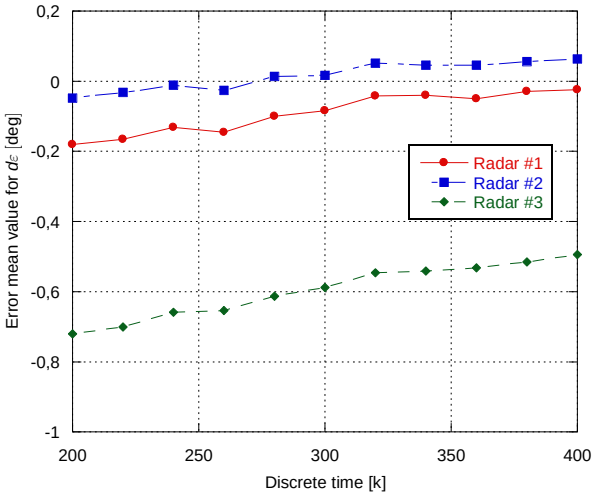

Figure 4.5 – Error mean value for the estimate of the elevation error.

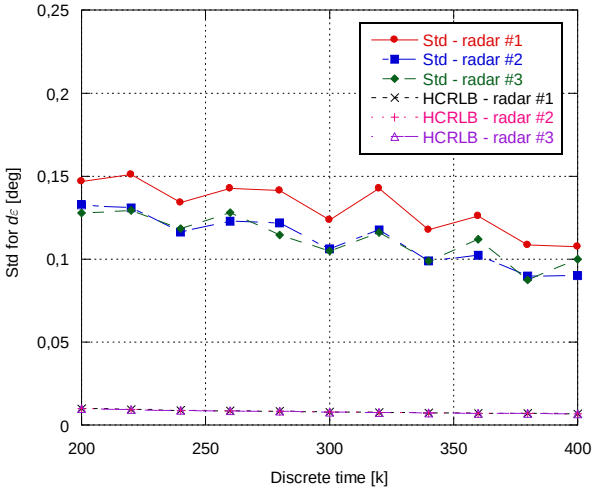

Figure 4.6 – RMSE for the estimate of the elevation error.

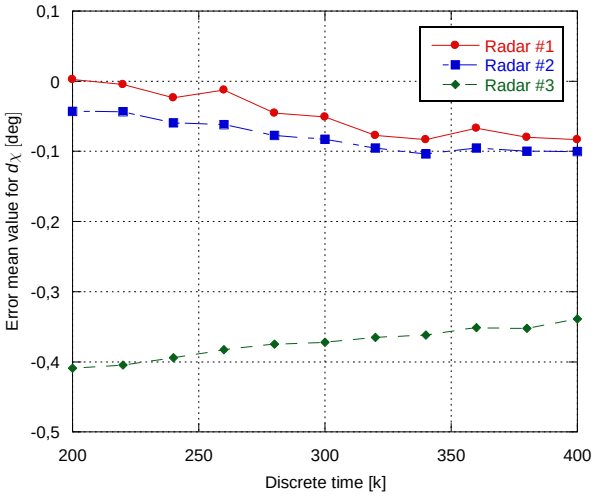

Figure 4.7 – Error mean value of the roll bias error.

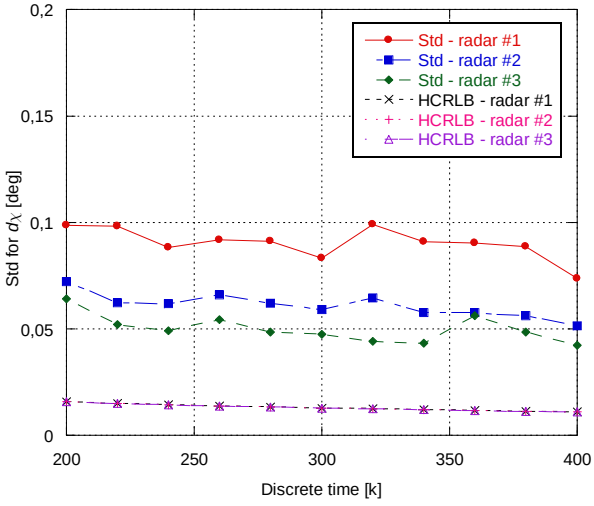

Figure 4.8 – Std of the roll bias error.

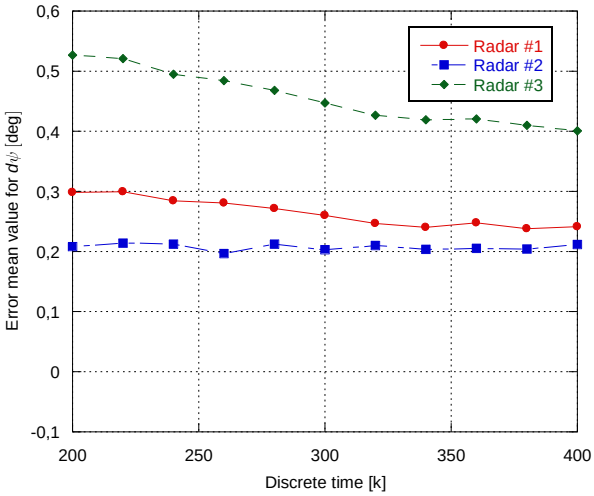

Figure 4.9 – Error mean value of the pitch error.

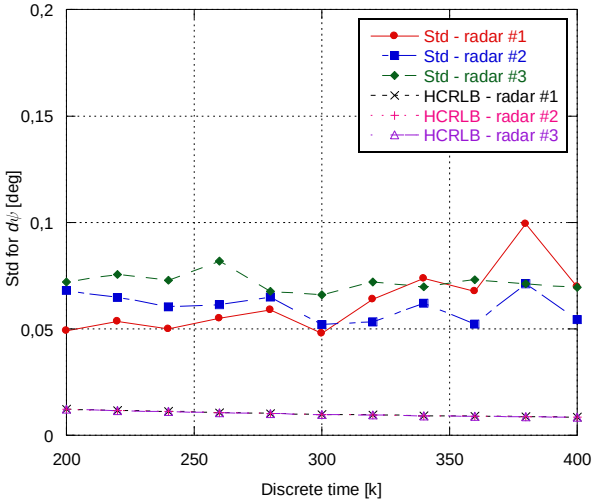

Figure 4.10 – RMSE of the pitch error.

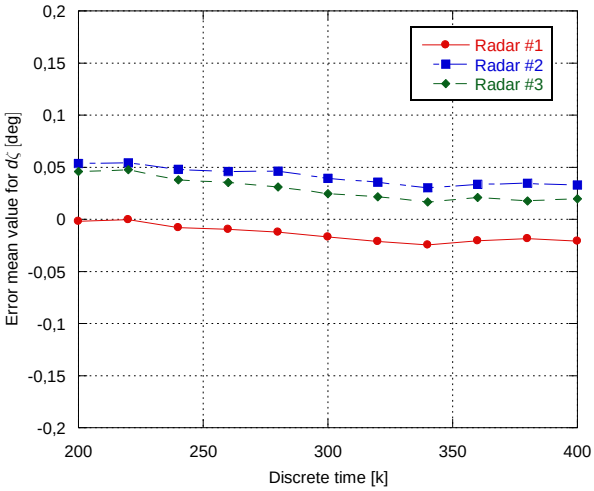

Figure 4.11 – Error mean value of the yaw+azimuth error.

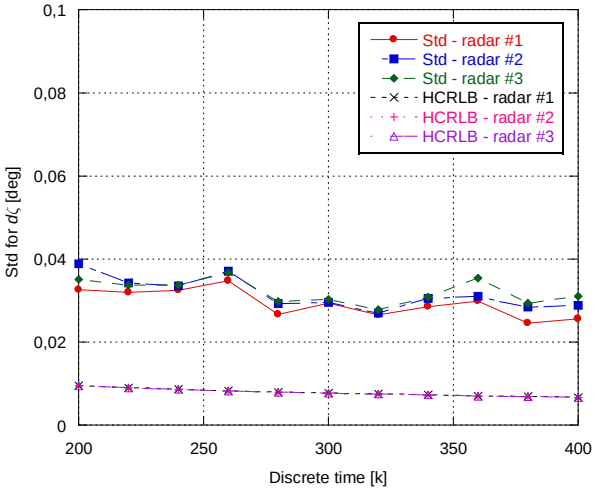

Figure 4.12 – RMSE of the yaw+azimuth error.

4.6 Summary

In this Chapter we generalize the EM algorithm, derived in Chapter 3 for the relative grid-locking problem, to the absolute grid-locking problem. In particular, we solve the general problem of the absolute grid-locking with M biased radars and N_t targets. Moreover, we evaluate the HCRLB for such general case. However, a certain number of problem is still open. The first problem is the initialization of the EM algorithm. As discussed in Section 4.2.1, the linear LS algorithm is not able to estimate the relative position error vector, then it cannot be used to initialize the EM algorithm if we want to estimate jointly all the registration bias errors (i.e. measurement errors, attitude errors and relative position errors). The second problem are the not satisfactory performance of the EM algorithm. In fact, as shown in Section 4.5, most of the estimates of the bias errors are biased. Finally, a fundamental problem seems to be the computational time. To make the application of the EM algorithm in real environments feasible, its computational time need to be drastically reduced. All these problems will be addressed in future works.

References

[1] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm”, *Journal of the Royal Statistical Society*, vol. 39, no.1, 1977, pp. 1-38.

[2] J. Wu, “On the convergence properties of the EM algorithm”, *Ann. Statist.* vol. 11, no. 1, 1983, pp. 95-103.

[3] Y. Noam, H. Messer, "Notes on the Tightness of the Hybrid Cramér–Rao Lower Bound", *Signal Processing, IEEE Transactions on* vol. 57, no. 6, pp. 2074–2084, June 2009.

[4] F. Gini and R. Reggiannini, "On the use of Cramér-Rao-Like bounds in the presence of random nuisance parameters," *Communications, IEEE Transactions on*, vol. 46, no. 7, pp. 2120-2126, December 2000.

[5] M. Pardini, F. Lombardini, F. Gini, "The Hybrid Cramér–Rao Bound on Broadside DOA Estimation of Extended Sources in Presence of Array Errors," *Signal Processing, IEEE Transactions on,* vol. 56, no. 4, pp. 1726 - 1730, April 2008.

[6] D. Huang, H. Leung, "An expectation-maximization-based interacting multiple model approach for cooperative driving systems," *Intelligent Transportation Systems, IEEE Transactions on,* vol. 6, no. 2, pp. 206- 228, June 2005.

[7] Z. Li, H. Leung, "An Expectation Maximization Based Simultaneous Registration and Fusion Algorithm for Radar Networks," *IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, pp. 31-35, Ottawa, May 2006.

[8] S. Chen, H. Leung and E. Bossè, "A Maximum Likelihood Approach to Joint Registration, Association and Fusion for Multi-Sensor Multi-Target Tracking", *12th Conference on Information Fusion*, Seattle, WA, USA, July 6-9, 2009.

[9] S. Chen, H. Leung and E. Bossè, "A Maximum Likelihood Approach to Joint Registration, Association and Fusion for Multi-Sensor Multi-Target Tracking", *12th International Conference on Information Fusion*, Seattle, WA, USA, July 6-9, 2009.

[10] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum Likelihood Estimates of Linear Dynamic Systems", *AIAA Journal*, vol. 3, no. 8, pp. 1445-1450, August 1965.

[11] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.

[12] M. Bazaraa, H. Sherali and C. Shetty, *Nonlinear Programming, Theory and Applications*, J. Wiley & Sons, New York, (Third Edition) 2006.

[13] S. Fortunati , A. Farina, F. Gini, A. Graziano, M. S. Greco, S. Giompapa, "Least Squares Estimation and Cramér-Rao Type Lower Bounds for Relative Sensor Registration Process," *Signal Processing, IEEE Transactions on*, vol. 59, no. 3, pp. 1075-1087, March 2011.

Appendix C

Evaluation of the matrix $\mathbf{M}_k(\mathbf{\Phi})$

In this Appendix, we evaluate the entries of the matrix $\mathbf{M}_k(\mathbf{\Phi})$ defined as:

$$\left[\mathbf{M}_k\left(\mathbf{\Phi}\right)\right]_{ij} = \left.\frac{\partial \tilde{\mu}_j\left(\mathbf{s}_k,\mathbf{\Phi}\right)}{\partial s_{k,i}}\right|_{\mathbf{s}^k=\hat{\mathbf{s}}^n_{k|K}}, \tag{C.1}$$

where the vector function $\tilde{\boldsymbol{\mu}}(\mathbf{s}^k;\mathbf{\Phi})$ is given by:

$$\tilde{\boldsymbol{\mu}}\left(\mathbf{s}_k;\mathbf{\Phi}\right) \triangleq \boldsymbol{\mu}\left(\mathbf{r}_k;\mathbf{\Phi}\right) \triangleq \mathbf{h}^{-1}\left(\mathbf{R}^T\left(\mathbf{\Theta}_m + d\mathbf{\Theta}\right)\left[\mathbf{r}_k - \left(\mathbf{t}_m + d\mathbf{t}\right)\right]\right) - d\mathbf{v}. \tag{C.2}$$

It is easy to show that the matrix $\mathbf{M}_k(\mathbf{\Phi})$ can be expressed as a block matrix:

$$\mathbf{M}_k\left(\mathbf{\Phi}\right) = \begin{pmatrix}\mathbf{B}_k\left(\mathbf{\Phi}\right)\\ \mathbf{0}_{3\times 3}\end{pmatrix}, \tag{C.3}$$

where the first 3×3 block matrix $\mathbf{B}_k(\mathbf{\Phi})$ is defined as:

$$\left[\mathbf{B}_k\left(\mathbf{\Phi}\right)\right]_{ij} = \left.\frac{\partial \mu_j\left(\mathbf{r}_k,\mathbf{\Phi}\right)}{\partial r_{k,i}}\right|_{\mathbf{r}_k=\hat{\mathbf{r}}^n_{k|K}}. \tag{C.4}$$

To evaluate each entry of the matrix in eq. (C.4), we use the chain rule of derivation. The derivative of each entry of the three-dimensional vector function in (C.2) can be expressed as:

$$\frac{\partial \mu_j\left(\mathbf{r}_k;\mathbf{\Phi}\right)}{\partial r_{k,i}} = \nabla h_j^{-1}(\mathbf{u})\cdot\frac{\partial \mathbf{u}\left(\mathbf{r}_k;\mathbf{\Phi}\right)}{\partial r_{k,i}},\ i,j=1,2,3, \tag{C.5}$$

where, ∇h_j^{-1} is the gradient of the jth entries of the vector function $\mathbf{h}^{-1}$. The vector function $\mathbf{u}(\mathbf{r}_k;\mathbf{\Phi})$ is defined, from (C.2), as:

$$\mathbf{u}\left(\mathbf{r}_k;\mathbf{\Phi}\right) = \mathbf{R}^T\left(\mathbf{\Theta}_m + d\mathbf{\Theta}\right)\cdot\left[\mathbf{r}_k - \left(\mathbf{t}_m + d\mathbf{t}\right)\right]. \tag{C.6}$$

The entries of the 3d vector function $\partial\mathbf{u}\left(\mathbf{r}_k;\mathbf{\Phi}\right)/\partial r_{k,i}$, for i=1,2,3, can be evaluated as:

$$\frac{\partial \mathbf{u}\left(\mathbf{r}_k;\boldsymbol{\Phi}\right)}{\partial \mathbf{r}_k} = \mathbf{R}^T\left(\boldsymbol{\Theta}_m + d\boldsymbol{\Theta}\right). \tag{C.7}$$

Finally, $\mathbf{B}_k(\boldsymbol{\Phi})$ can be rewritten in a compact matrix form as:

$$\mathbf{B}_k\left(\boldsymbol{\Phi}\right) = \left[\mathbf{J}\left(\mathbf{h}^{-1}\left(\mathbf{u}(\mathbf{r}_k;\boldsymbol{\Phi})\right)\right)\Big|_{\mathbf{r}_k=\hat{\mathbf{r}}_{k|K}^{m}} \cdot \mathbf{R}^T\left(\boldsymbol{\Theta}_m + d\boldsymbol{\Theta}\right)\right]^T, \tag{C.8}$$

where $\mathbf{J}(\cdot)$ is the Jacobian matrix whose entries are given by:

$$\begin{aligned}
&\left[\mathbf{J}\left(\mathbf{h}^{-1}\left(\mathbf{u}\right)\right)\right]_{1j} = \frac{u_j}{\sqrt{u_1^2+u_2^2+u_3^2}}, \quad j=1,2,3\\
&\left[\mathbf{J}\left(\mathbf{h}^{-1}\left(\mathbf{u}\right)\right)\right]_{21} = \frac{u_2}{u_1^2+u_2^2}, \quad \left[\mathbf{J}\left(\mathbf{h}^{-1}\left(\mathbf{u}\right)\right)\right]_{22} = \frac{-u_1}{u_1^2+u_2^2}, \quad \left[\mathbf{J}\left(\mathbf{h}^{-1}\left(\mathbf{u}\right)\right)\right]_{23} = 0\\
&\left[\mathbf{J}\left(\mathbf{h}^{-1}\left(\mathbf{u}\right)\right)\right]_{31} = \frac{-u_3}{\sqrt{u_1^2+u_2^2}}\frac{u_1}{u_1^2+u_2^2+u_3^2},\\
&\left[\mathbf{J}\left(\mathbf{h}^{-1}\left(\mathbf{u}\right)\right)\right]_{31} = \frac{-u_3}{\sqrt{u_1^2+u_2^2}}\frac{u_2}{u_1^2+u_2^2+u_3^2},\\
&\left[\mathbf{J}\left(\mathbf{h}^{-1}\left(\mathbf{u}\right)\right)\right]_{33} = \frac{\sqrt{u_1^2+u_2^2}}{u_1^2+u_2^2+u_3^2},
\end{aligned} \tag{C.9}$$

where, for ease of notation, the dependence on $\mathbf{r}_k$ and $\boldsymbol{\Phi}$ of the vector function $\mathbf{u}(\mathbf{r}_k;\boldsymbol{\Phi})$ is omitted.

Appendix D

Evaluation of the matrices $\mathbf{L}\left(\mathbf{s}_k\right)$ and $\mathbf{G}\left(\mathbf{s}_k;\boldsymbol{\Phi}\right)$ for the HCRLB.

Calculation of $\mathbf{L}\left(\mathbf{s}_k\right)$.

Here we evaluate the entries of the matrix $\mathbf{L}(\mathbf{s}_k)$ defined in eq. (3.73) as:

$$l_{ij}\left(\mathbf{s}_k\right) = \left(\frac{\partial \tilde{\mathbf{h}}^{-1}\left(\mathbf{s}_k\right)}{\partial \Psi_i}\right)^T \mathbf{C}_1^{-1}\left(\frac{\partial \tilde{\mathbf{h}}^{-1}\left(\mathbf{s}_k\right)}{\partial \Psi_j}\right), \tag{D.1}$$

where the vector function $\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k)$ is the Cartesian-to-spherical transformation, defined as:

$$\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k) = \begin{pmatrix} \sqrt{s_{k,1}^2 + s_{k,2}^2 + s_{k,3}^2} \\ \operatorname{atan}\left(s_{k,1}/s_{k,2}\right) \\ \operatorname{atan}\left(s_{k,3}/\sqrt{s_{k,1}^2 + s_{k,2}^2}\right) \end{pmatrix}. \tag{D.2}$$

Here we have to calculate the derivative terms of the vector function $\partial\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k)/\partial\Psi_i$ where $\boldsymbol{\Psi} = [\boldsymbol{\Phi}^T \quad \mathbf{s}_1^T \quad \ldots \quad \mathbf{s}_K^T]^T$ is the vector of the hybrid parameters with dimension $d = d_{\boldsymbol{\Phi}} + 6K$ where $d_{\boldsymbol{\Phi}} = 8$. We need to calculate the derivative terms with respect to $\Psi_i, i = d_{\boldsymbol{\Phi}} + (6k-5), \ldots, d_{\boldsymbol{\Phi}} + 6k$ can be evaluated as follows:

$$\frac{\partial\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k)}{\partial\Psi_{d_{\boldsymbol{\Phi}}+(6k-5),\ldots,d_{\boldsymbol{\Phi}}+6k}} = \left(\mathbf{J}\left(\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k)\right) \quad \mathbf{0}_{3\times 3}\right), \tag{D.3}$$

where $\mathbf{J}(\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k))$ is the Jacobian matrix of $\tilde{\mathbf{h}}^{-1}(\cdot)$ evaluated at $\mathbf{s}_k$. For all the others indices different from the previous ones, the derivative vector is a zero-vector. For this reason, the matrix $\mathbf{L}(\mathbf{s}_k)$ can be rewritten in the block matrix form given in eq. (3.75), where the matrix $\mathbf{N}_k$ is the following block matrix:

$$\mathbf{N}_k = \begin{pmatrix} \mathbf{J}\left(\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k)\right)^T \mathbf{C}_1^{-1} \mathbf{J}\left(\tilde{\mathbf{h}}^{-1}(\mathbf{s}_k)\right) & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \end{pmatrix}. \tag{D.4}$$

Calculation of $\mathbf{G}(\mathbf{s}_k;\boldsymbol{\Phi})$.

Here we evaluate the entries of the matrix $\mathbf{G}(\mathbf{s}_k;\boldsymbol{\Phi})$ defined in eq. (3.74) as:

$$g_{ij}(\mathbf{s}_k;\boldsymbol{\Phi}) = \left(\frac{\partial\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi})}{\partial\Psi_i}\right)^T \mathbf{C}_2^{-1} \left(\frac{\partial\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi})}{\partial\Psi_j}\right), \tag{D.5}$$

where the vector function $\tilde{\boldsymbol{\mu}}(\mathbf{s}_k,\boldsymbol{\Phi})$ is defined in eq. (C.2). The first two derivative terms with respect to the range bias error $d\rho$ (i.e. Ψ_1) and to the elevation bias error $d\varepsilon$ (i.e. Ψ_2) are given by:

$$\frac{\partial\tilde{\boldsymbol{\mu}}(\mathbf{s}_k;\boldsymbol{\Phi})}{\partial\Psi_1} = -\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad , \quad \frac{\partial\tilde{\boldsymbol{\mu}}(\mathbf{s}_k;\boldsymbol{\Phi})}{\partial\Psi_2} = -\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{D.6}$$

Then, as before, we use the chain rule of derivation to evaluate all other terms:

$$\frac{\partial \tilde{\mu}_i(\mathbf{s}_k;\boldsymbol{\Phi})}{\partial \Psi_j} = \nabla h_i^{-1}(\tilde{\mathbf{u}}) \cdot \frac{\partial \tilde{\mathbf{u}}(\mathbf{s}_k;\boldsymbol{\Phi})}{\partial \Psi_j},\ i = 1,2,3;\ j = 3,\ldots,d_{\boldsymbol{\Phi}} + 6K \tag{D.7}$$

where the vector function $\tilde{\mathbf{u}}(\mathbf{s}_k;\boldsymbol{\Phi})$ is defined in eq. (C.6). The entries of the vector function $\partial\tilde{\mathbf{u}}(\mathbf{s}_k;\boldsymbol{\Phi})/\partial\Psi_i$, for $i = 3,\ldots,d_{\boldsymbol{\Phi}} = 8$, can be evaluated as:

$$\frac{\partial \tilde{\mathbf{u}}(\mathbf{s}_k;\boldsymbol{\Phi})}{\partial \Psi_3} = \left(\frac{\partial \mathbf{R}(d\boldsymbol{\Theta})}{\partial d\chi}\right)^T \cdot \mathbf{R}^T(\boldsymbol{\Theta}_m) \cdot \left[\mathbf{r}_k - (\mathbf{t}_m + d\mathbf{t})\right], \tag{D.8}$$

$$\frac{\partial \tilde{\mathbf{u}}(\mathbf{s}_k;\boldsymbol{\Phi})}{\partial \Psi_4} = \left(\frac{\partial \mathbf{R}(d\boldsymbol{\Theta})}{\partial d\psi}\right)^T \cdot \mathbf{R}^T(\boldsymbol{\Theta}_m) \cdot \left[\mathbf{r}_k - (\mathbf{t}_m + d\mathbf{t})\right], \tag{D.9}$$

$$\frac{\partial \tilde{\mathbf{u}}(\mathbf{s}_k;\boldsymbol{\Phi})}{\partial \Psi_5} = \left(\frac{\partial \mathbf{R}(d\boldsymbol{\Theta})}{\partial d\xi}\right)^T \cdot \mathbf{R}^T(\boldsymbol{\Theta}_m) \cdot \left[\mathbf{r}_k - (\mathbf{t}_m + d\mathbf{t})\right], \tag{D.10}$$

$$\frac{\partial \tilde{\mathbf{u}}(\mathbf{s}_k;\boldsymbol{\Phi})}{\partial \Psi_{6,7,8}} = -\mathbf{R}^T(\boldsymbol{\Theta}_m + d\boldsymbol{\Theta}). \tag{D.11}$$

Now, using the same procedure as before, the derivative terms of $\tilde{\boldsymbol{\mu}}(\mathbf{s}_k;\boldsymbol{\Phi})$ with respect to $\Psi_i, i = d_{\boldsymbol{\Phi}} + (6k-5),\ldots,d_{\boldsymbol{\Phi}} + 6k$ can be evaluated as follows:

$$\frac{\partial \tilde{\boldsymbol{\mu}}(\mathbf{s}_k;\boldsymbol{\Phi})}{\partial \Psi_{d_{\boldsymbol{\Phi}}+(6k-5),\ldots,d_{\boldsymbol{\Phi}}+6k}} = \left(\mathbf{J}\left(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k;\boldsymbol{\Phi})\right)\mathbf{R}^T(\boldsymbol{\Theta}_m + d\boldsymbol{\Theta}) \quad \mathbf{0}_{3\times 3}\right), \tag{D.12}$$

where $\mathbf{J}(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k;\boldsymbol{\Phi}))$ is the Jacobian matrix of $\tilde{\boldsymbol{\mu}}(\cdot;\boldsymbol{\Phi})$ evaluated at $\mathbf{s}_k$. For all the others indices different from the previous ones, the derivative vector is a zero-vector. For this reason, the matrix $\mathbf{G}(\mathbf{s}_k;\boldsymbol{\Phi})$ can be rewritten in the block matrix form given in eq. (3.77), where the matrix $\mathbf{M}_k$ is the following block matrix:

$$\mathbf{M}_k = \begin{pmatrix} \mathbf{R}(\boldsymbol{\Theta}_m + d\boldsymbol{\Theta})\mathbf{J}\left(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k;\boldsymbol{\Phi})\right)^T \mathbf{C}_2^{-1}\mathbf{J}\left(\tilde{\boldsymbol{\mu}}(\mathbf{s}_k;\boldsymbol{\Phi})\right)\mathbf{R}(\boldsymbol{\Theta}_m + d\boldsymbol{\Theta})^T & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \end{pmatrix}. \tag{D.13}$$

Appendix E

Evaluation of the gradient and of the Hessian matrix of $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$

The aim this Appendix is to show that the gradient of $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$ is equal to zero and that the Hessian matrix is exactly the Modified FIM given in eq. (3.117). We recall here for clarity the definition of the Kullback-Leibler divergence $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$ under Definition 5:

$$\begin{aligned} H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0) &= \int \ln\left(\frac{p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi})}{p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)}\right) p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)\, d\mathbf{x}d\mathbf{a} \\ &= \int \ln\left(\frac{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})\, p(\mathbf{a})}{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi}_0)\, p(\mathbf{a})}\right) p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)\, d\mathbf{x}d\mathbf{a} \\ &= \int \ln\left(\frac{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})}{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi}_0)}\right) p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)\, d\mathbf{x}d\mathbf{a}. \end{aligned} \tag{E.1}$$

where we used the fact that $p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0) = p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi}_0)\, p(\mathbf{a})$ (Assumption 5).

To start, we need the following lemma:

Lemma 1: Let $p(\mathbf{x};\boldsymbol{\Phi})$ *be the pdf of the random vector* $\mathbf{x} \in \mathbb{R}^n$ *parameterized by the deterministic vector* $\boldsymbol{\Phi} \in \Omega$. *By assuming that the usual regularity conditions on* $p(\mathbf{x};\boldsymbol{\Phi})$ *are verified and that* $\int p(\mathbf{x};\boldsymbol{\Phi})\, d\mathbf{x} = 1,\ \forall \boldsymbol{\Phi} \in \Omega$, *then the following relations hold:*

1. $\int \frac{\partial}{\partial \Phi_i} p(\mathbf{x};\boldsymbol{\Phi})\, d\mathbf{x} = 0,\ \forall i = 1,\ldots,\dim(\boldsymbol{\Phi})$,
2. $\int \frac{\partial}{\partial \Phi_i \partial \Phi_j} p(\mathbf{x};\boldsymbol{\Phi})\, d\mathbf{x} = 0,\ \forall i, j = 1,\ldots,\dim(\boldsymbol{\Phi})$.

Proof: The first statement of Lemma 1 can be easily proven as follows:

$$\int \frac{\partial}{\partial \Phi_i} p(\mathbf{x};\boldsymbol{\Phi})\, d\mathbf{x} = \frac{\partial}{\partial \Phi_i} \int p(\mathbf{x};\boldsymbol{\Phi})\, d\mathbf{x} = \frac{\partial}{\partial \Phi_i} 1 = 0,\ \forall i = 1,\ldots,\dim(\boldsymbol{\Phi}) \tag{E.2}$$

where the order change is justified by the regularity assumptions made on $p(\mathbf{x};\boldsymbol{\Phi})$. The second statement can be proven exactly in the same way:

$$\int \frac{\partial}{\partial\Phi_i \partial\Phi_j} p(\mathbf{x};\boldsymbol{\Phi})\, d\mathbf{x} = \frac{\partial}{\partial\Phi_i \partial\Phi_j} \int p(\mathbf{x};\boldsymbol{\Phi})\, d\mathbf{x} = \frac{\partial}{\partial\Phi_i \partial\Phi_j} 1 = 0,\ \forall i, j = 1,\ldots,\dim(\boldsymbol{\Phi}) \text{ .(E.3)}$$

This conclude the proof of the Lemma 1.

Now we return to the main problem. First we evaluate the derivative terms of $H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$ as:

$$\begin{aligned}
\frac{\partial}{\partial\Phi_i} H_M(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} &= \int \frac{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi}_0)}{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})} \frac{\partial p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})/\partial\Phi_i}{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi}_0)} p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)\, d\mathbf{x}d\mathbf{a}\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \\
&= \int \frac{\partial}{\partial\Phi_i} p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})\, p(\mathbf{a})\, d\mathbf{x}d\mathbf{a}\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \qquad \text{(E.4)} \\
&= \int \frac{\partial}{\partial\Phi_i} p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi})\, d\mathbf{x}d\mathbf{a}\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \overset{\text{Lemma 1}}{=} 0. \quad \forall i = 1,\ldots,\dim(\boldsymbol{\Phi})
\end{aligned}$$

Eq. (E.4) shows that the vector parameter $\boldsymbol{\Phi}_0$ is a stationary point for $H(\boldsymbol{\Phi};\boldsymbol{\Phi}_0)$, as function of $\boldsymbol{\Phi}$. Now, we pass to evaluate the entries of the Hessian matrix. The second-order derivative terms can be evaluated as:

$$\begin{aligned}
&\frac{\partial^2}{\partial\Phi_i \partial\Phi_j} H_M(\boldsymbol{\Phi},\boldsymbol{\Phi}_0)\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} = \int \frac{\partial}{\partial\Phi_j} \left[\frac{\partial p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})/\partial\Phi_i}{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})} \right] p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)\, d\mathbf{x}d\mathbf{a}\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \\
&= \int \left[\frac{\partial^2 p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})/\partial\Phi_i \partial\Phi_j}{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})} - \frac{\partial p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})/\partial\Phi_i\, \partial p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})/\partial\Phi_j}{p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})^2} \right] p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)\, d\mathbf{x}d\mathbf{a}\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \qquad \text{(E.5)} \\
&\overset{\text{Lemma 1}}{=} -\int \left(\frac{\partial}{\partial\Phi_i} \ln p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \frac{\partial}{\partial\Phi_j} \ln p(\mathbf{x}|\mathbf{a};\boldsymbol{\Phi})\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \right) p(\mathbf{x},\mathbf{a};\boldsymbol{\Phi}_0)\, d\mathbf{x}d\mathbf{a},
\end{aligned}$$

$\forall i, j = 1,\ldots,\dim(\boldsymbol{\Phi})$, where the last equality is obtained by using the second relation given in Lemma 1. Then, we get that each entry of the Hessian matrix evaluated at $\boldsymbol{\Phi}_0$, i.e. $[\mathbf{H}(H_M)]_{ij}(\boldsymbol{\Phi}_0)$, is:

$$
\begin{aligned}
\left[\mathbf{H}\left(H_M\right)\right]_{ij}\left(\boldsymbol{\Phi}_0\right) &= \frac{\partial^2}{\partial\Phi_i\partial\Phi_j} H_M\left(\boldsymbol{\Phi};\boldsymbol{\Phi}_0\right)\Bigg|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0} \\
&= -E_{\mathbf{x},\mathbf{a}}\left\{\frac{\partial}{\partial\Phi_i} p\left(\mathbf{x}\middle|\mathbf{a};\boldsymbol{\Phi}\right)\Big|_{\boldsymbol{\Gamma}=\boldsymbol{\Phi}} \frac{\partial}{\partial\Phi_j} p\left(\mathbf{x}\middle|\mathbf{a};\boldsymbol{\Phi}\right)\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0}\right\} \\
&= E_{\mathbf{x},\mathbf{a}}\left\{\frac{\partial^2}{\partial\Phi_i\partial\Phi_j} p\left(\mathbf{x}\middle|\mathbf{a};\boldsymbol{\Phi}\right)\Big|_{\boldsymbol{\Phi}=\boldsymbol{\Phi}_0}\right\} \\
&= -\left[\mathbf{I}_M\left(\boldsymbol{\Phi}_0\right)\right]_{ij},
\end{aligned}
\tag{E.6}
$$

where $\mathbf{I}_M(\boldsymbol{\Phi}_0)$ is the MFIM. This concludes the proof.

Part III: Intrinsic covariance matrix estimation and its application to the radar target detection

Chapter 5: Intrinsic estimation in the manifold of the symmetric positive-definite matrices and its applications to the radar target detection

5.1 Introduction

The goal of this Chapter is to describe a new approach to the radar target detection and clutter suppression problem. This new approach, introduced and discussed for the first time in [1], [2], [3], is based on a non-Euclidean description of the geometry of the set of the covariance matrices that is the set of the symmetric (Hermitian) positive-definite matrices. The non-Euclidean nature of this set is a well known topic and there is a lot of mathematical works on it, e. g. [4]-[10], but the application of these geometrical concepts to the radar signal processing is still an open problem.

First, we give a very short explanation of the main concepts of the differential geometry on a generic differentiable manifold. After this brief introduction, we define the geometrical structure, given by the Riemann metric and by the geodesic equation on the manifold of the symmetric positive-definite matrices ([4], [5] and [11]). Starting from the obtained results, we show the intrinsic distance between two generic symmetric positive-definite matrices [11]. This distance is of fundamental importance to define the concept of "intrinsic mean", or Karcher-Fréchet mean or Riemann barycentre. We show a matrix formula that characterizes the Karcher-Fréchet mean on the manifold of the symmetric positive definite matrices [5], but such mean cannot be evaluated in closed form. Then, we describe a gradient descent algorithm in order to evaluate recursively the mean value [1], [10].

After this first general part, the detection problem in radar system is addressed. The two fundamental aspects of a radar detection algorithm are the decision rule and the clutter covariance matrix estimation. There is a huge radar literature on these two topics, some of the fundamental works are: L. E. Brennan and L. S. Reed [12] and of E. J. Kelly [13] for the decision rule and the paper of I. S. Reed, J. D. Mallet and L. E. Brennan [14] for the clutter covariance matrix estimation.

The differential geometry provides a new method for both the decision rule and for the clutter covariance matrix estimation. In this work, we try to show a theoretical comparison between the classical and the new decision rule based on the Riemann geometry. Then, we pay attention on the clutter covariance matrix estimation algorithms. A comparison among various estimators is performed in terms of error mean value and Root Mean Squares Errors (RMSE) compared with the Flat and the Intrinsic CRLB evaluated in [11]. Finally, we give the overall performance of the detection algorithm in terms of ROC (Receiver Operating Characteristics) for different matrix mean estimators.

5.2 Preliminaries: the exponential mapping

It can be defined an "exponential" (bijective) mapping between the tangent space and the manifold M as follows:

$$\begin{aligned} &\exp_{\mathbf{A}} : T_{\mathbf{A}}M \to M, \\ &\exp_{\mathbf{A}}(\boldsymbol{\Omega}) : \boldsymbol{\Omega} \mapsto \mathbf{A}. \end{aligned} \tag{5.1}$$

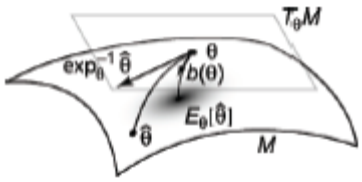

Figure 5.1. A manifold M, its tangent space and the exponential map.

Intuitively, the exponential map $\exp_{\mathbf{A}}$ equates points on the manifold with point in the tangent space $T_{\mathbf{A}}P$ at $\mathbf{A}$. The inverse mapping is the logarithmic map defied as:

$$\begin{aligned} \log_{\mathbf{A}} &\triangleq \exp_{\mathbf{A}}^{-1} : M \to T_{\mathbf{A}}M, \\ \log_{\mathbf{A}}(\mathbf{A}) &: \mathbf{A} \mapsto \mathbf{\Omega}. \end{aligned} \tag{5.2}$$

If M is a matrix Lie group, then the exponential map coincides with the matrix exponential and is given by the series expansion:

$$\exp_{\mathbf{A}}(\mathbf{\Omega}) = \sum_{k=0}^{\infty} \frac{1}{k!} \mathbf{\Omega}^k. \tag{5.3}$$

The logarithm of a matrix $\mathbf{A}$ is the solution of the matrix equation $\exp(\mathbf{\Omega})=\mathbf{A}$. When matrix $\mathbf{A}$ does not possess any negative eigenvalues, there exists its unique real logarithm, termed "principal logarithm". Furthermore, if, for any given matrix norm $\|\cdot\|$, it holds that $\|\mathbf{I}-\mathbf{A}\|<1$, then:

$$\log(\mathbf{A}) = -\sum_{k=1}^{\infty} \frac{(\mathbf{I}-\mathbf{A})^k}{k}. \tag{5.4}$$

In practise, matrix exponential and logarithm may be computed efficiently by making use of the spectral decomposition or by various approximate methods. In the case of symmetric $n \times n$ matrices $\mathbf{A}$, for example, matrix exponential can be computed by using the spectral decomposition:

$$\mathbf{A} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T, \tag{5.5}$$

where $\mathbf{V} \in O(n)$ is the column vector matrix of eigenvectors and $\mathbf{\Sigma}$ is the diagonal matrix of eigenvalues. Thus, the exponential matrix $\exp(\mathbf{A})$ is given by:

$$\begin{aligned} \exp(\mathbf{A}) &= \mathbf{V}\exp(\mathbf{\Sigma})\mathbf{V}^T = \\ &= \mathbf{V}\mathrm{diag}\left(\exp(\lambda_1),\ldots,\exp(\lambda_n)\right)\mathbf{V}^T. \end{aligned} \tag{5.6}$$

Similarly, the matrix logarithm, for a symmetric $n \times n$ matrices $\mathbf{A}$ can be computed as:

$$\begin{aligned} \log(\mathbf{A}) &= \mathbf{V}\log(\mathbf{\Sigma})\mathbf{V}^T = \\ &= \mathbf{V}\mathrm{diag}\left(\log(\lambda_1),\ldots,\log(\lambda_n)\right)\mathbf{V}^T. \end{aligned} \tag{5.7}$$

Proposition 1: *The exponential function* $\exp : S(n) \to P(n)$ *given by*

$$\exp(\mathbf{A}) = \mathbf{V}\exp(\mathbf{\Sigma})\mathbf{V}^T \tag{5.8}$$

is a bijection. In particular, a symmetric matrix is positive-definite if and only if it is the exponential of a symmetric matrix [7].

Preposition 1 allows us to defined the tangent space of the manifold *P* as the vector space of the symmetric real matrices:

$$T_{\mathbf{A}}P = S(n), \quad \mathbf{A} \in P. \tag{5.9}$$

In the rest of this work we assume that all the consideration and theorems stated for the on the real field, can be directly extended to the complex field.

5.3 Geometrical characterization of a Riemann manifolds

To define the geometrical properties of a manifold we need two fundamental structures: the Riemann metric g and an affine connection ∇ (this is not a gradient operator) [11].

A Riemann metric g is defined as an inner (or scalar) product on the manifold tangent space, i. e.:

$$g : T_{\mathbf{A}}M \times T_{\mathbf{A}}M \to \mathbb{R}. \tag{5.10}$$

If $\mathbf{\Omega}$ is a tangent vector, then the squared length of $\mathbf{\Omega}$ is given by:

$$\|\mathbf{\Omega}\|^2 = \langle \mathbf{\Omega}, \mathbf{\Omega} \rangle_{\mathbf{A}} \triangleq g_{\mathbf{A}}(\mathbf{\Omega}, \mathbf{\Omega}). \tag{5.11}$$

Note that this inner product depends on the location of the tangent space.

The affine connection ∇, intuitively, allows one to "connect" different tangent space and compare objects defined separately at each point. In other words [4], an affine connection is a continuous collection of scalar products on the tangent space at each point of the manifold. Thus, if we consider a curve on the manifold, we can compute at each point its instantaneous velocity vector and its norm, the instantaneous speed. To compute the length of the curve, we can proceed as usual by integrating this value along the curve. Each affine connection ∇ has an associated Christoffel operator defined as $\Gamma(\cdot,\cdot)$ [11].

The distance between two points of a connected Riemann manifold is the minimum length curve among the curves joining these points. The curve realizing this minimum for any two points of the manifold are called geodesics. The mathematical definition of geodesic is:

$$\begin{gathered}\gamma:[0,1]\to P,\\ \gamma(0)=\mathbf{A}\in P\,;\,\gamma(1)=\mathbf{B}\in P;\dot{\gamma}(0)=\boldsymbol{\Omega}\in T_{\mathbf{A}}P,\\ d(\mathbf{A},\mathbf{B})=\min_{\gamma}\int_0^1\left\|\dot{\gamma}(t)\right\|_{\gamma(t)}dt=\min_{\gamma}\int_0^1\left\langle\dot{\gamma}(t),\dot{\gamma}(t)\right\rangle_{\gamma(t)}^{\frac{1}{2}}dt.\end{gathered}\tag{5.12}$$

The calculus of variations shows that geodesics are the solution of a second order differential equation depending on the Riemann metric [11]:

$$\begin{cases}\ddot{\gamma}(t)+\Gamma_{\gamma(t)}\left(\dot{\gamma}(t),\dot{\gamma}(t)\right)=0\\ \gamma(0)=\mathbf{A}\\ \dot{\gamma}(0)=\boldsymbol{\Omega}\end{cases},\tag{5.13}$$

where $\Gamma_{\cdot}(\cdot,\cdot)$ denotes the Christoffel operator [11]. In a flat vector space, the solution of the Cauchy problem in eq. (5.13) is trivial, since the Christoffel operator is identically zero, than we have:

$$\begin{cases}\ddot{\gamma}(t)=0\\ \gamma(0)=\mathbf{A}\\ \dot{\gamma}(0)=\boldsymbol{\Omega}\end{cases}\quad\Rightarrow\quad\gamma(t)=\mathbf{A}+\boldsymbol{\Omega}t.\tag{5.14}$$

The eq. in eq. (5.14) can be easily rewritten as function on the point **B** by setting:

$$\gamma(1)=\mathbf{B}=\mathbf{A}+\boldsymbol{\Omega}\quad\Rightarrow\quad\boldsymbol{\Omega}=\mathbf{B}-\mathbf{A}.\tag{5.15}$$

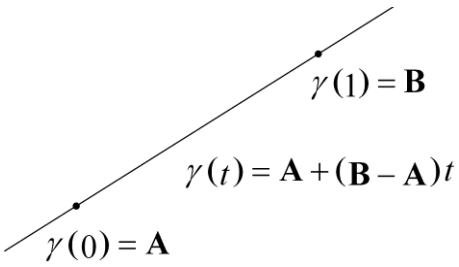

Figure 5.2. Geodetic curve in a flat Euclidean space.

5.4 Manifold of the symmetric positive-definite matrices

The manifold of the positive-definite symmetric matrices, *P*, can be defined in two different ways [11], but both of them yields the same Riemann structure defined as the scalar product in the tangent space at any point **A**. We can define *P* as:

1. Quotient space of the General Linear group with respect to the Orthogonal group:

$$P(n) = GL(n)/O(n). \tag{5.16}$$

2. Invariance under coordinate transformations. Formally we can defined *P* as the set of the equivalence class under the transitive action of *GL*(*n*) defined as:

$$\begin{gathered}\mathbf{A} \simeq \mathbf{A}' \quad \Leftrightarrow \quad \mathbf{A}' = \mathbf{R} * \mathbf{A} \triangleq \mathbf{R}\mathbf{A}\mathbf{R}^T, \\ \mathbf{R} \in GL(n).\end{gathered} \tag{5.17}$$

It can be proved that, the scalar product at a point **A** between two vectors $\mathbf{\Omega}, \mathbf{\Theta} \in T_{\mathbf{A}}P$ is given by:

$$\langle \mathbf{\Omega}, \mathbf{\Theta} \rangle_{\mathbf{A}} = \mathrm{tr}\left(\mathbf{\Omega}\mathbf{A}^{-1}\mathbf{\Theta}\mathbf{A}^{-1}\right) = \mathrm{tr}\left(\mathbf{A}^{-1/2}\mathbf{\Omega}\mathbf{A}^{-1}\mathbf{\Theta}\mathbf{A}^{-1/2}\right). \tag{5.18}$$

The equation of the geodetic in the manifold *P* can be obtain as solution of the Cauchy problem in eq. (5.13) with a Christoffel symbol given by [11]:

$$\Gamma_{\mathbf{A}}(\mathbf{\Omega}, \mathbf{\Theta}) = -\frac{1}{2}\left(\mathbf{\Omega}\mathbf{A}^{-1}\mathbf{\Theta} + \mathbf{\Theta}\mathbf{A}^{-1}\mathbf{\Omega}\right). \tag{5.19}$$

Finally, the closed form of the geodetic equation on *P* is:

$$\begin{gathered}\gamma(0) = \mathbf{A} \in P; \dot{\gamma}(0) = \mathbf{\Omega} \in T_{\mathbf{A}}P, \\ \gamma(t) = \mathbf{A}^{1/2} \exp\left(t\mathbf{A}^{-1/2}\mathbf{\Omega}\mathbf{A}^{-1/2}\right)\mathbf{A}^{1/2}.\end{gathered} \tag{5.20}$$

As before, the geodetic can be expressed as function of the ending point **B**, by setting:

$$\gamma(1) = \mathbf{B} = \mathbf{A}^{1/2} \exp\left(\mathbf{A}^{-1/2}\mathbf{\Omega}\mathbf{A}^{-1/2}\right)\mathbf{A}^{1/2}, \tag{5.21}$$

and then, through some algebra, we get:

$$\mathbf{\Omega} = \mathbf{A}^{1/2} \log\left(\mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}\right)\mathbf{A}^{1/2}, \tag{5.22}$$

and finally, by substituting eq. (5.22) in eq. (5.20), we obtain:

$$\gamma(t) = \mathbf{A}^{1/2} \exp\left(t \log\left(\mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}\right)\right)\mathbf{A}^{1/2}. \tag{5.23}$$

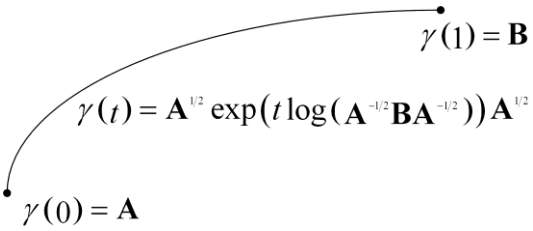

Figure 5.3. Gedetic on the manifold *P*.

5.5 Distance between two points on *P*

Using the eq. (5.12) and the scalar product in eq. (5.18), we can define the distance between two points **A** and **B**. Noticing that $\dot{\gamma}(t) \in T_{\mathbf{A}}P$, we have:

$$d(\mathbf{A},\mathbf{B}) = \int_0^1 \left\|\dot{\gamma}(t)\right\|_{\gamma(t)} dt = \int_0^1 \left\langle \dot{\gamma}(t), \dot{\gamma}(t)\right\rangle_{\gamma(t)}^{\frac{1}{2}} dt, \tag{5.24}$$

where $\left\langle \dot{\gamma}(t), \dot{\gamma}(t)\right\rangle_{\gamma(t)}$ is defined in eq. (5.18). From eq. (5.20), $\dot{\gamma}(t)$ can be calculated as:

$$\dot{\gamma}(t) = \mathbf{\Omega}\mathbf{A}^{-1/2} \exp\left(t \log\left(\mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}\right)\right)\mathbf{A}^{1/2} = \mathbf{\Omega}\mathbf{A}^{-1}\gamma(t), \tag{5.25}$$

then

$$\begin{aligned}
\left\langle \dot{\gamma}(t), \dot{\gamma}(t)\right\rangle_{\gamma(t)} &= \operatorname{tr}\left(\dot{\gamma}(t)\gamma^{-1}(t)\dot{\gamma}(t)\gamma^{-1}(t)\right) = \\
&= \operatorname{tr}\left(\mathbf{\Omega}\mathbf{A}^{-1}\gamma(t)\gamma^{-1}(t)\mathbf{\Omega}\mathbf{A}^{-1}\gamma(t)\gamma^{-1}(t)\right) = \\
&= \operatorname{tr}\left(\mathbf{\Omega}\mathbf{A}^{-1}\mathbf{\Omega}\mathbf{A}^{-1}\right) \overset{\text{eq. 22}}{=} \\
&= \operatorname{tr}\left(\log\left(\mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}\right)\log\left(\mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}\right)\right) = \\
&= \left\|\log\left(\mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}\right)\right\|_F^2.
\end{aligned} \tag{5.26}$$

By substituting eq. (5.26) in eq. (5.24), we get:

$$d(\mathbf{A},\mathbf{B}) = \left\|\log\left(\mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}\right)\right\|_F = \sqrt{\sum_{i=1}^{n}\left(\log\lambda_i\right)^2}, \tag{5.27}$$

where λ_i are the eigenvalues of the generalized problem **A**-λ**B** or **B**-λ**A** or equivalently the eigenvalues of the matrix $\mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}$.

It can be noted that the integral in eq. (5.24) can be rewritten in terms of the infinitesimal arc length of *ds* as:

$$d\left(\mathbf{A},\mathbf{B}\right)=\int_{\mathbf{A}}^{\mathbf{B}} ds, \tag{5.28}$$

where the infinitesimal arc length can be obtained as:

$$ds^2=\left\langle \dot{\gamma}(t),\dot{\gamma}(t)\right\rangle_{\gamma(t)} dt^2=\mathrm{tr}\left(\left(\gamma^{-1}(t)\dot{\gamma}(t)\right)^2\right)=\mathrm{tr}\left(\left(\gamma^{-1}(t)d\gamma(t)\right)^2\right)dt^2, \tag{5.29}$$

that is called Siegel metric [1], [2], [3].

5.6 Characterization of two intrinsic mean operators

5.6.1 The Karcher-Fréchet (KF) mean

Given a Riemann manifold *M* and a distance $d(\cdot,\cdot)$ on *M*, the KF mean is characterized by the variational property [5], [6]: it minimizes the sum of the squared distances to the given points $\mathbf{m}_k$:

$$\bar{\mathbf{m}}=\arg\min_{\mathbf{m}\in M}\sum_{k=1}^{n} d^2\left(\mathbf{m},\mathbf{m}_k\right). \tag{5.30}$$

Proposition 2: *The Riemann barycentre of n points* $\mathbf{m}_1,\ldots,\mathbf{m}_n$ *of a manifold M with non positive sectional curvature always exists and it's unique* [5] [6].

In our case the manifold is *P* and the distance is defined in eq. (5.27), then:

$$\begin{aligned}\bar{\mathbf{A}}&=\arg\min_{\mathbf{A}\in P}\sum_{k=1}^{n} d^2\left(\mathbf{A},\mathbf{B}_k\right)=\arg\min_{\mathbf{A}\in P}\sum_{k=1}^{n}\left\|\log\left(\mathbf{A}^{-1/2}\mathbf{B}_k\mathbf{A}^{-1/2}\right)\right\|_F^2=\\&=\arg\min_{\mathbf{A}\in P}\sum_{k=1}^{n}\mathbf{tr}\left(\log^2\left(\mathbf{A}^{-1/2}\mathbf{B}_k\mathbf{A}^{-1/2}\right)\right).\end{aligned} \tag{5.31}$$

In the following, we try to get a closed form solution for the minimum problem in eq. (5.31). First, we start to define the gradient of a function defined on a manifold *M*.

For a real valued function $f(\mathbf{A})$ defined on a Riemann manifold M (in our case P), the gradient ∇f at point $\mathbf{A} \in M$ is the unique tangent vector defined implicitly as:

$$\left\langle \mathbf{\Omega}, \nabla f \right\rangle_{\mathbf{A}} = \frac{d}{dt} f\left(\gamma(t)\right)\Big|_{t=0}, \tag{5.32}$$

where $\gamma(t)$ is a geodetic emanating from $\mathbf{A}$ in the direction of $\mathbf{\Omega}$, and $\left\langle \cdot, \cdot \right\rangle_{\mathbf{A}}$ denotes the Riemann inner product on the tangent space (in our case, it is given in eq. (5.18)).

Now, we can define an objective function as:

$$f\left(\mathbf{A}\right) = \sum_{k=1}^{n} \mathbf{tr}\left(\log^2\left(\mathbf{A}^{-1/2}\mathbf{B}_k\mathbf{A}^{-1/2}\right)\right). \tag{5.33}$$

In order to find out the minimum of eq. (5.31), we have to minimize the function $f(\mathbf{A})$ defined in (5.33), then we have to set to zero its gradient:

$$\nabla f\left(\mathbf{A}\right)\Big|_{\mathbf{A}=\bar{\mathbf{A}}} = \mathbf{0}. \tag{5.34}$$

It can be proofed ([5], [10]) that, in our case, the gradient at a point $\mathbf{A}$ is given by:

$$\nabla f\left(\mathbf{A}\right) = \mathbf{A}^{1/2}\sum_{k=1}^{n}\log\left(\mathbf{A}^{-1/2}\mathbf{B}_k\mathbf{A}^{-1/2}\right)\mathbf{A}^{1/2}, \tag{5.35}$$

then, forcing eq. (5.35) to zero, we get:

$$\nabla f\left(\mathbf{A}\right)\Big|_{\mathbf{A}=\bar{\mathbf{A}}} = \mathbf{0} \quad \Rightarrow \quad \sum_{k=1}^{n}\log\left(\bar{\mathbf{A}}^{-1/2}\mathbf{B}_k\bar{\mathbf{A}}^{-1/2}\right) = \mathbf{0}. \tag{5.36}$$

The solution of the non linear matrix equation cannot be given explicitly except for the case with n=2. In such a case, the Riemann barycentre between two symmetric positive-definite matrices is given by any of the six equivalent expressions [5] [10]:

$$\begin{aligned}\bar{\mathbf{A}} &= \mathbf{B}_1\left(\mathbf{B}_1^{-1}\mathbf{B}_2\right)^{1/2} = \mathbf{B}_2\left(\mathbf{B}_2^{-1}\mathbf{B}_1\right)^{1/2} = \left(\mathbf{B}_2^{-1}\mathbf{B}_1\right)^{1/2}\mathbf{B}_1 = \left(\mathbf{B}_1^{-1}\mathbf{B}_2\right)^{1/2}\mathbf{B}_2 = \\ &= \mathbf{B}_1^{1/2}\left(\mathbf{B}_1^{-1/2}\mathbf{B}_2\mathbf{B}_1^{-1/2}\right)\mathbf{B}_1^{1/2} = \mathbf{B}_2^{1/2}\left(\mathbf{B}_2^{-1/2}\mathbf{B}_1\mathbf{B}_2^{-1/2}\right)\mathbf{B}_2^{1/2}.\end{aligned} \tag{5.37}$$

In the case of n>2, an iterative gradient descent algorithm can be used to evaluate the KF mean. Let $f(\mathbf{A})$ the objective function to minimize (in our case it is eq. (5.33)), and let $\bar{\mathbf{A}}_t$ be the current estimation of the minimum point $\bar{\mathbf{A}}$ and $\nabla f\left(\bar{\mathbf{A}}\right)$ the gradient at point $\bar{\mathbf{A}}$.

The principle of a first order gradient descent algorithm is to go toward the steepest descent, in the direction opposite to the gradient for a short time-step ε, and iterate the process. However, the standard operator, given by:

$$\bar{\mathbf{A}}_{t+1} = \bar{\mathbf{A}}_t - \varepsilon \nabla f\left(\bar{\mathbf{A}}_t\right) \tag{5.38}$$

is only valid for very short time steps in the flat Euclidean matrix space, and we could easily go out of the cone of symmetric positive definite matrices. A much more interesting numerical operator is given by following the geodesic backward starting at $\bar{\mathbf{A}}_t$ with tangent vector $\nabla f\left(\bar{\mathbf{A}}_t\right)$ during a time ε. This intrinsic gradient descent ensures that we cannot leave the manifold *P*. The recursive formula can be easily expressed using the exponential map as [1]:

$$\begin{aligned} \bar{\mathbf{A}}_{t+1} = \gamma(-\varepsilon) &= \bar{\mathbf{A}}_t^{1/2} \exp\left(\varepsilon \bar{\mathbf{A}}_t^{-1/2} \nabla f\left(\bar{\mathbf{A}}_t\right) \bar{\mathbf{A}}_t^{-1/2}\right) \bar{\mathbf{A}}_t^{1/2} = \\ &= \bar{\mathbf{A}}_t^{1/2} \exp\left(\varepsilon \sum_{k=1}^{n} \log\left(\bar{\mathbf{A}}_t^{-1/2} \mathbf{B}_k \bar{\mathbf{A}}_t^{-1/2}\right)\right) \bar{\mathbf{A}}_t^{1/2}. \end{aligned} \tag{5.39}$$

It can be proofed that the manifold of symmetric positive definite matrices *P* has a non positive curvature, so that there is one and only one mean value $\bar{\mathbf{A}}$ (Prop. 2).

5.6.1.A The KF mean as ML estimator for the mean value of a set of i.i.d. Gaussian distributed matrices

The Gaussian probability density function (pdf) on *P* is defined as [4] [22]:

$$f_{\mathbf{A}}\left(\mathbf{A}\right) = h \exp\left(-\frac{1}{2} \mu_{\Omega}^2\left(\mathbf{A}\right)\right), \tag{5.40}$$

where *h* is a normalization constant and the quadratic form μ_{Ω}^2 can be expressed as:

$$\mu_{\Omega}^2\left(\mathbf{A}\right) = \mathrm{vec}[\mathbf{\Omega}_{\mathbf{MA}}]^H \mathbf{C}^{-1} \mathrm{vec}\left[\mathbf{\Omega}_{\mathbf{MA}}\right], \tag{5.41}$$

and

$$\Omega_{\mathbf{MA}} = \mathbf{M}^{1/2} \log\left(\mathbf{M}^{-1/2} \mathbf{A} \mathbf{M}^{-1/2}\right) \mathbf{M}^{1/2} \tag{5.42}$$

is the tangent vector at the mean value **M** to a point **A** on *P*.

The symbol "vec" defines the operator that arranges in an unique column vector the columns of a matrix and **C** is the covariance matrix.

Now we proof that the Karcher mean defined in eq. (5.36) represent the ML estimator for the mean value **M**.

Theorem 1: *Let* $\{\mathbf{A}_k\}_{k=1}^K$ *be a set of K Hermitian positive definite matrices sampled from a Gaussian pdf defined on P, with mean value* **M** *and covariance matrix* $\mathbf{C}=\mathbf{M}\otimes\mathbf{M}$. *Then the ML estimator* $\hat{\mathbf{M}}_{ML}$ *of the mean value* **M**, *is the Karcher mean of the set* $\{\mathbf{A}_k\}_{k=1}^K$.

Proof.

First, we define a set of *K* independent samples from the Gaussian pdf in eq. (5.40), i. e. $\{\mathbf{A}_k\}_{k=1}^K$. The joint pdf of this set of sample is the product of the marginal pdf:

$$f\left(A_1,\ldots A_K\right)=\prod_{k=1}^{K}h\exp\left(-\frac{1}{2}\mu_{\boldsymbol{\Omega}}^2\left(\mathbf{A}_k\right)\right)=h^K\exp\left(-\frac{1}{2}\sum_{k=1}^{K}\mu_{\boldsymbol{\Omega}}^2\left(\mathbf{A}_k\right)\right). \tag{5.43}$$

Through some matrix manipulation and with the particular choice of the covariance matrix **C** such that $\mathbf{C}=\mathbf{M}\otimes\mathbf{M}$, eq. (5.43) can be rewritten as:

$$f\left(A_1,\ldots A_K\right)=h^K\exp\left(-\frac{1}{2}\sum_{k=1}^{K}\mathrm{tr}\left(\log\left(\mathbf{M}^{-1/2}\mathbf{A}_k\mathbf{M}^{-1/2}\right)^2\right)\right). \tag{5.44}$$

From the joint pdf in eq. (5.44), the log-likelihood function of the mean value **M** is (ignoring constants):

$$l\left(\mathbf{M}\right)=-\frac{1}{2}\sum_{k=1}^{K}\mathrm{tr}\left(\log\left(\mathbf{M}^{-1/2}\mathbf{A}_k\mathbf{M}^{-1/2}\right)^2\right). \tag{5.45}$$

The ML estimator of the mean value **M** can be found by setting to zero the derivative of the log-likelihood function with respect to **M**:

$$\nabla l\left(\mathbf{M}\right)\Big|_{\mathbf{M}=\hat{\mathbf{M}}_{ML}}=\mathbf{0}. \tag{5.46}$$

This problem is formally identical to the one expressed in eq. (5.34) that defines the Karcher mean. In fact, the log-likelihood function defined in eq. (5.45) and the matrix function defined in eq. (5.33) are equal. Then, we can write:

$$\nabla l(\mathbf{M})\Big|_{\mathbf{M}=\hat{\mathbf{M}}_{ML}} = \mathbf{0} \quad \Rightarrow \quad \sum_{k=1}^{n} \log\left(\hat{\mathbf{M}}_{ML}^{-1/2}\mathbf{A}_k\hat{\mathbf{M}}_{ML}^{-1/2}\right) = \mathbf{0}. \tag{5.47}$$

■

This concludes the proof.

Other work is needed to generalize this Theorem to Gaussian matrices with arbitrary covariance matrix **C**.

5.6.2 The Log-Euclidean (LE) mean

In this section we introduce another algorithm to compute the mean on *P*. To derive the recursive formula in eq. (5.39), we have used the geometrical property of the manifold *P*. The problem of finding the mean of a set of symmetric positive-definite (spd) matrices could be solved using, instead of the geometrical structure, the Lie Group structure of *P* [8], [9], [15]. This approach leads us to a closed form for the mean expression that is slightly different from the optimal solution in eq. (5.39) but that can be computed approximately 20 times faster [8]. The basic idea is to give to the group of the symmetric positive-definite (spd) matrices a vector space structure. Of course, this doesn't mean that the space of the sdp matrices is a vector subspace of the vector space of the squared matrices. The idea is to use the logarithm function as an isomorphism between *P* and the vector subspace of the symmetric matrices as shown in the Preliminaries. The question of whether or not the sdp matrix space is a vector space depends on the vector space structure we are considering, and not on the space itself.

A vector space can be defined as $(V, \oplus, \odot)$ where *V* is a set, $\oplus$ is an inner operator, i.e. given $v_1, v_2 \in V$ then $v_1 \oplus v_2 \in V$, and $\odot$ is the scalar multiplication, i. e. $\lambda \in \mathbb{R}, v \in V$, then $\lambda \odot v \in V$. An example is given by the classical vector space in $\mathbb{R}^n$ as $(\mathbb{R}^n, +, \cdot)$ where + is the classical addition between two vector and λ is the classical multiplication between a scalar and a vector. Now, we want to define a vector space on *P*. We can endow *P* with a vector space structure $(P, \oplus, \odot)$ by defining a inner operator as [8]:

$$\mathbf{A}, \mathbf{B} \in P, \ \mathbf{A} \oplus \mathbf{B} \triangleq \exp\left(\log(\mathbf{A}) + \log(\mathbf{B})\right). \tag{5.48}$$

With this definition of "addition" we have that the neutral element is the usual identity matrix and the inverse is the inverse in the matrix sense. Moreover, whenever two sdp matrices commute in the matrix sense, the sum defined in eq. (5.48) is equal to their matrix product. Finally, such addition is commutative.

The scalar multiplication can be defined as:

$$\lambda \in \mathbb{R}, \mathbf{A} \in P\ ,\ \lambda \odot \mathbf{A} = \exp\left(\lambda \cdot \log\left(\mathbf{A}\right)\right) = \mathbf{A}^{\lambda}. \tag{5.49}$$

Now we can define the Log-Euclidean mean. It is well known that, in a vector space structure, the mean can be evaluated as:

$$\overline{v} = \frac{1}{N}\sum_{i=1}^{N} v_i. \tag{5.50}$$

Since we have endowed the set of the sdp matrices with a vector space structure, we can use the Euclidean mean to compute the mean on *P*. Then, we have:

$$\overline{\mathbf{A}} = \exp\left(\frac{1}{N}\sum_{i=1}^{N}\log\left(\mathbf{A}_i\right)\right). \tag{5.51}$$

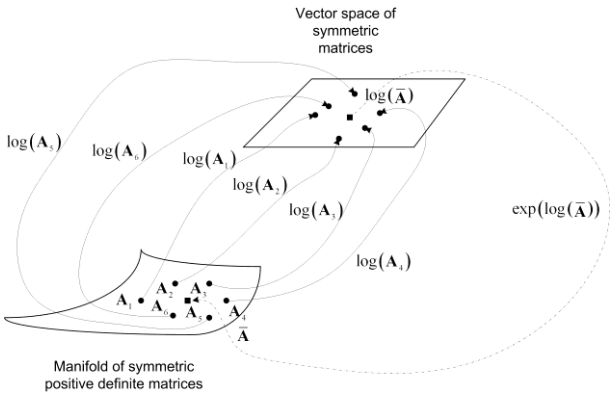

Figure 5.4: Log-Euclidean mean.

5.6.3 Comparison between the KF and the LE means

In this section, we give a brief explanation about the difference between the KF mean in eq. (5.39) and the Log-Euclidean mean in eq. (5.51). The main difference is that the KF mean is affine invariant, i. e. for all invertible matrix **C**, we have:

$$\forall i \; \mathbf{C}\mathbf{B}_i\mathbf{C}^T \rightarrow \mathbf{C}\bar{\mathbf{A}}_{Riemann}\mathbf{C}^T, \tag{5.52}$$

that means that if we make a change of coordinates for all the matrices $\mathbf{B}_i$ to be averaged, then the resulting KF mean will be affected by the same change of coordinates.

It can be proofed ([8],[9]) that the Log-Euclidean mean is not full affine invariant, but is only invariant to rotations and scaling, i. e. if we indicate with **R** a rotation matrix and with α>0 a scaling factor, we have:

$$(\alpha\mathbf{R})\mathbf{B}_i(\alpha\mathbf{R})^T \rightarrow (\alpha\mathbf{R})\bar{\mathbf{A}}_{Log-Euclidean}(\alpha\mathbf{R})^T. \tag{5.53}$$

Finally, we give a criterion for the equality of the two means:

Preposition 3: *Let* $\{\mathbf{B}_i\}_{i=1}^N$ *be N sdp matrices, and let* $\bar{\mathbf{L}}$ *be the Euclidean mean of their logarithms, i.e.* $\bar{\mathbf{L}} = 1/N\sum_{i=1}^N \log(\mathbf{B}_i)$, *if* $\bar{\mathbf{L}}$ *commutes with all* $\log(\mathbf{B}_i)$, *then the Log-Euclidean and the KF means are identical* [8].

5.7 Radar target detection in the presence of additive clutter

Here, we give a short explanation of a possible application of the theoretical concepts discussed in this Chapter. In particular, we try to apply this new geometric concept to the radar detection in additive clutter. First, we briefly recall this well-known detection problem.

The problem of detecting a target signal in additive clutter can be stated in terms of the following binary hypotheses test for *k*th cell as:

$$\begin{cases} H_0: & \mathbf{z}^k = \mathbf{c}^k \\ H_1: & \mathbf{z}^k = \mathbf{s} + \mathbf{c}^k \end{cases}, \quad k = 1, \ldots, K, \tag{5.54}$$

where $\mathbf{z}^k$, $\mathbf{s}$ and $\mathbf{c}^k$ are N-dimensional complex vectors that represent the temporal samples from the kth cell. The desired signal $\mathbf{s}$ can be modelled as $\mathbf{s}=\alpha\mathbf{p}$ where $\mathbf{p}$ is the target steering vector and α is a random parameter accounting for the channel propagation effect and the target radar cross section. From (5.54) we can calculate the covariance matrix of the temporal sample vector $\mathbf{z}^k$ for each cell as:

$$\mathbf{R}^k_{\mathbf{z}^k|H_0} = E\left\{\mathbf{z}^k \left(\mathbf{z}^k\right)^H\right\} = E\left\{\mathbf{c}^k \left(\mathbf{c}^k\right)^H\right\} = \mathbf{M}^k, \tag{5.55}$$

$$\begin{aligned} \mathbf{R}^k_{\mathbf{z}^k|H_1} &= E\left\{\mathbf{z}^k \left(\mathbf{z}^k\right)^H\right\} = E\left\{\left(\mathbf{s}+\mathbf{c}^k\right)\left(\mathbf{s}+\mathbf{c}^k\right)^H\right\} = \\ &= E\left\{\alpha\alpha^*\right\}\mathbf{p}\mathbf{p}^H + E\left\{\mathbf{c}^k \left(\mathbf{c}^k\right)^H\right\} = \sigma_\alpha^2 \mathbf{p}\mathbf{p}^H + \mathbf{M}^k. \end{aligned} \tag{5.56}$$

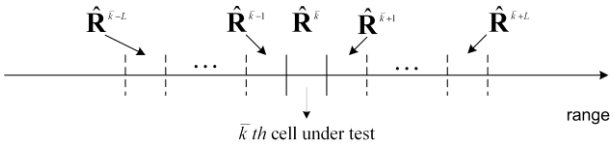

Figure 5.5. Scenario's geometry

There are two key points in all the radar detection algorithms:

1. The decision rule,
2. The estimation of the clutter covariance matrix.

The classical approach to the radar detection problem is based on two fundamental papers [12], [14]. In the first paper, the authors show the optimal detection rule (i. e. a rule that maximizes the probability of detection given a certain value of the false alarm probability) under the hypothesis of Gaussian clutter. They show that the optimum filter is composed of two cascaded operations: the (linear) transformation of the disturbance correlated component (clutter) into white noise; the result of this operation is then multiplied by the steering vector to enhance the useful signal (matched filter), if present. The second fundamental paper describe how to estimate the clutter covariance matrix from the received data, necessary to decorrelate the disturbance.

Using the theoretical concepts discussed in the previous sections, in [1], [2] and [3] a new approach both for the decision rule and for the clutter covariance matrix estimation is introduced. In the next Section, a comparison between the classical decision rule and a new rule based on the Riemann geometry is made in the ideal case i. e. when the clutter covariance matrix is assumed known. Then, we discuss various estimation algorithms for the clutter covariance matrix.

5.7.1 Comparison between the classical decision criterion and the Riemann distance based criterion

In this section, we provide a comparison between the classical decision rule, based on the maximization of the signal to noise (SNR) ratio, and the new criterion based on the structure of the covariance matrix set. Using the notation introduced in the previous section, we define with $\mathbf{z} \in \mathbb{C}^N$, $\mathbf{s} = \alpha \mathbf{p} \in \mathbb{C}^N$ and $\mathbf{c} \in \mathbb{C}^N$ the N-dimensional complex vectors that represent the received data, the signal and the interference noise (clutter). For ease of notation, in the following discussion, the index of the cell under test is neglected.

A. Classical decision rule

In the classical decision theory, the decision rule is defined as follows:

$$|y| \underset{H_1}{\overset{H_0}{\lessgtr}} \lambda, \tag{5.57}$$

where the statistic y is the output of a linear filter given by [12]:

$$y = \sum_{n=1}^{N} w_n^* z_n = \mathbf{w}^H \mathbf{z}. \tag{5.58}$$

The mean value and the variance of y, given H_1 or H_0, can be expressed as:

$$E\{y\} = \begin{cases} H_0: & 0 \\ H_1: & E\{\alpha\}\mathbf{w}^H\mathbf{p} \end{cases}, \tag{5.59}$$

$$\text{var}\{y\} = \mathbf{w}^H \mathbf{M} \mathbf{w}. \tag{5.60}$$

The optimum decision criterion for detecting the signal **s** in presence of clutter **c** is the likelihood ratio test:

$$\frac{f\left(y|H_1\right)}{f\left(y|H_0\right)} \underset{H_1}{\overset{H_0}{\lessgtr}} \lambda, \tag{5.61}$$

where $f(y|H_1)$ and $f(y|H_0)$ are the probability density functions (pdfs) of the statistic *y* given the hypotheses H_1 or H_0 respectively. It can be proved [12] that, if the clutter is zero-mean, Gaussian-distributed complex random vector, the likelihood ratio test in eq. (5.61) is equivalent to maximize the output signal to noise ratio given by [12]:

$$SNR_{out} = \frac{\left|\mathbf{w}^H\mathbf{s}\right|^2}{\mathbf{w}^H\mathbf{M}\mathbf{w}}. \tag{5.62}$$

It can be shown [12] that the maximum of the SNR_{out} is obtained for a weight vector

$$\mathbf{w} = q\mathbf{M}^{-1}\mathbf{s} = q\alpha\mathbf{M}^{-1}\mathbf{p}. \tag{5.63}$$

where *q* is a complex constant not equal to zero and **p** is the steering vector. In the following, *q* is supposed to have unit value. By substituting the weight vector in eq. (5.62) and by assuming known the clutter covariance matrix **M**, we get the maximum of the output signal to noise ratio as:

$$\max_{\mathbf{w}}\left\{SNR_{out}\right\} = \left|\alpha\right|^2\mathbf{p}^H\mathbf{M}^{-1}\mathbf{p} \tag{5.64}$$

By using the weight vector in eq. (5.63), we can write the expression of the statistic *y* as:

$$y = \begin{cases} H_0: & \mathbf{w}^H\mathbf{z} = \alpha^*\mathbf{p}^H\mathbf{M}^{-1}\mathbf{c} \\ H_1: & \mathbf{w}^H\mathbf{z} = \left|\alpha\right|^2\mathbf{p}^H\mathbf{M}^{-1}\mathbf{p} + \alpha^*\mathbf{p}^H\mathbf{M}^{-1}\mathbf{c} \end{cases}. \tag{5.65}$$

In [12], it is shown that the linear filter in eq. (5.58) with a weight vector given by eq. (5.63), is the filter that gives the maximum probability of detection P_D for a prescribed probability of false alarm P_{FA}. The probability of detection can be written in closed form as function of the P_{FA} (for a Swerling 1 target model) as [16]:

$$P_D = Q\left(\sqrt{2\sigma_\alpha^2\mathbf{p}^H\mathbf{M}^{-1}\mathbf{p}}, \sqrt{2\ln\frac{1}{P_{FA}}}\right), \tag{5.66}$$

where $Q(\cdot)$ is the Marcum function and σ_α^2 is the variance of the parameter α.

B. Riemann distance-based decision rule

As proposed in [1], [2] and [3], the decision rule based on the Riemann distance is:

$$d\left(\bar{\mathbf{M}}, \hat{\mathbf{R}}^{\bar{k}}\right) \underset{H_1}{\overset{H_0}{\lessgtr}} \lambda, \tag{5.67}$$

where $d(\cdot,\cdot)$ is defined in eq. (5.27). Now, we want to evaluate the statistic under the two hypotheses H_0 and H_1. By assuming known the clutter covariance matrix **M**, we have:

- Hypothesis H_0. In this case, the average clutter covariance matrix evaluated on the cells neighbouring to the one under test and the clutter covariance matrix for the cell under test are equal: $\bar{\mathbf{M}} = \mathbf{R}^{\bar{k}} = \mathbf{M}$. Then the decision rule becomes:

$$d\left(\bar{\mathbf{M}}, \mathbf{R}^{\bar{k}}\right) = d\left(\mathbf{M}, \mathbf{M}\right) = 0. \tag{5.68}$$

- Hypothesis H_1. In this case we have that the average covariance matrix is equal to the nominal one, $\bar{\mathbf{M}} = \mathbf{M}$, but, according to eq. (5.56), the covariance matrix for the cell under test is $\mathbf{R}^{\bar{k}} = \sigma_\alpha^2 \mathbf{p}\mathbf{p}^H + \mathbf{M}$. Then the decision rule becomes:

$$\begin{aligned} d\left(\bar{\mathbf{M}}, \mathbf{R}^{\bar{k}}\right) &= \left\| \log\left(\bar{\mathbf{M}}^{-1/2} \mathbf{R}^{\bar{k}} \bar{\mathbf{M}}^{-1/2}\right) \right\|_F = \\ &= \left\| \log\left(\mathbf{M}^{-1/2} \left(\sigma_\alpha^2 \mathbf{p}\mathbf{p}^H + \mathbf{M}\right) \mathbf{M}^{-1/2}\right) \right\|_F = \\ &= \left\| \log\left(\sigma_\alpha^2 \mathbf{M}^{-1/2} \mathbf{p}\mathbf{p}^H \mathbf{M}^{-1/2} + \mathbf{I}\right) \right\|_F = \\ &= \log\left(\sigma_\alpha^2 \mathbf{p}^H \mathbf{M}^{-1} \mathbf{p} + 1\right). \end{aligned} \tag{5.69}$$

Finally, the previous result can be summarized as follows:

$$d\left(\bar{\mathbf{M}}, \mathbf{R}^{\bar{k}}\right) = \begin{cases} H_0: & 0 \\ H_1: & \log\left(\sigma_\alpha^2 \mathbf{p}^H \mathbf{M}^{-1} \mathbf{p} + 1\right) \end{cases}. \tag{5.70}$$

In order to compare the two algorithms, we can define an output signal plus noise-to-noise ratio as:

$$SNNR_{out} = \frac{E\left\{\left|y_{H_1}\right|^2\right\}}{E\left\{\left|y_{H_0}\right|^2\right\}}, \tag{5.71}$$

that gives us an idea about how much 'larger' is the decision statistic when a target occurs with respect to the case in which only the noise is present. In the classical case we have:

$$\begin{aligned} SNNR_{out} &= \frac{E\left\{\left|y_{H_1}\right|^2\right\}}{E\left\{\left|y_{H_0}\right|^2\right\}} = \frac{E\left\{\left|\left|\alpha\right|^2 \mathbf{p}^H\mathbf{M}^{-1}\mathbf{p} + \alpha^* \mathbf{p}^H\mathbf{M}^{-1}\mathbf{c}\right|^2\right\}}{E\left\{\left|\alpha^* \mathbf{p}^H\mathbf{M}^{-1}\mathbf{c}\right|^2\right\}} = \\ &= \frac{\sigma_\alpha^4 \left|\mathbf{p}^H\mathbf{M}^{-1}\mathbf{p}\right|^2 + \sigma_\alpha^2 \mathbf{p}^H\mathbf{M}^{-1}\mathbf{p}}{\sigma_\alpha^2 \mathbf{p}^H\mathbf{M}^{-1}\mathbf{p}} = \sigma_\alpha^2 \mathbf{p}^H\mathbf{M}^{-1}\mathbf{p} + 1. \end{aligned} \tag{5.72}$$

We cannot calculate this statistic for the Riemann-distance-based algorithm because of the value of statistic, in the ideal case, doesn't depend on the clutter data. Keeping in mind that the two expectation operators in eq. (5.72) represent the power of the decision's statistic, we can link the Riemann distance with these two quantities:

$$E\left\{\left|y_{H_1}\right|^2\right\} \rightleftarrows d_{H_1}, \qquad E\left\{\left|y_{H_0}\right|^2\right\} \rightleftarrows d_{H_0}. \tag{5.73}$$

If we use the (monotonic) exponential function to link the quantities in eq. (5.73), we achieve following result. So, we have:

$$\begin{aligned} SNNR_{out} &= \frac{\exp\left(d_{H_1}\right)}{\exp\left(d_{H_0}\right)} = \exp\left(d_{H_1} - d_{H_0}\right) = \\ &= \exp\left(\log\left(\sigma_\alpha^2 \mathbf{p}^H\mathbf{M}^{-1}\mathbf{p} + 1\right)\right) = \sigma_\alpha^2 \mathbf{p}^H\mathbf{M}^{-1}\mathbf{p} + 1. \end{aligned} \tag{5.74}$$

As we can see, using the exponential (then monotonic) transformation, the two algorithms have the same performance in terms of output signal plus noise–to-noise ratio. This means that, instead of the decision rule in eq. (5.67), we must use the equivalent decision rule:

$$\exp\left(d\left(\bar{\mathbf{M}}, \hat{\mathbf{R}}^{\bar{k}}\right)\right) \underset{H_1}{\overset{H_0}{\lessgtr}} \lambda'. \tag{5.75}$$

5.8 Comparison among covariance matrix estimation algorithms

In this Section, five different algorithms for the estimate of the clutter covariance matrix are discussed. The first is the classical one, the Sample Covariance Matrix (SCM) estimators, then three other estimator, based on a different definition of matrix mean, are introduced. Finally, an estimator based on the MEM algorithm is discussed. Moreover, an explicit formula of the Implicit (i. e. based on the intrinsic geometrical characteristics of the manifold of the Hermitian positive-definite matrices) Cramér-Rao Lower Bound (CRLB) and the flat (i. e. based on the assumption of that the manifold of the Hermitian positive-definite matrices is an Euclidean space) CRLB is given. Finally, all the estimators are compared with these two bounds.

5.8.1. Sample Covariance Matrix (SCM)

The SCM method is the most used method to estimate the covariance matrix of a set of data and, in our specific case, of clutter data. This algorithm assumes implicitly the stationarity of the clutter, i. e. the clutter covariance matrix is the same for the all the data. In formula, the SCM estimator can be expressed as:

$$\hat{\mathbf{M}}_{SCM} = \frac{1}{K}\sum_{k=1}^{K}\mathbf{z}^k(\mathbf{z}^k)^H, \tag{5.76}$$

where $\mathbf{z}^k$ is the *k*th data vector coming from the *k*th cell (see fig. 5.6). In order to assure that the estimated covariance matrix is positive-definite the number of data used for the estimation must be greater than the dimension of the data vector, then according to the notation introduced before, $K \geq N$. However, as show in [14], the condition $K \geq N$ doesn't assure a good output SNR. The loss in output SNR due to the use of the SCM instead of the true clutter covariance matrix, expressed in decibels, is:

$$\text{loss} = -10\log_{10}\left[(K+2-N)/(K+1)\right]. \tag{5.77}$$

If one wishes to maintain an average loss less than 3 db, by eq. (5.77) the number of data must be $K = 2N-3 \cong 2N$.

5.8.2 Matrix Means

The next three estimators are based on the idea of averaging the clutter covariance matrix relative to each cell of the secondary data, i. e. $\{\hat{\mathbf{M}}^k\}_{k=1,k\neq\bar{k}}^{K}$ where $\bar{k}$ is the index of the cell under test. The following three estimators differ for the particular definition of matrix mean: the first estimator is based on the classical Euclidean definition of matrix mean while the others are based on a non-Euclidean definition. Before applying these three algorithms, we need an estimation of the clutter covariance matrix for each cell. In the next subsection we discuss a method to get such estimation, then the three algorithms to compute the matrix mean are described.

5.8.2.1 Estimation of the covariance matrix for each cell through MEM algorithm.

For each cell, we have at disposal N complex samples, collected in the vector $\mathbf{z}^k$. The idea is to estimate the Power Spectral Density (and then, the autocorrelation function) of the clutter in a given cell k from the data vector $\mathbf{z}^k$. In [17], the Maximum Entropy Method (MEM) is used to estimate the clutter's PSD. In this paper, the authors suggest to use the Burg's algorithm for the clutter spectrum estimation in order to design the optimum filter for clutter cancellation. It can be shown [17] that, applying the MEM algorithm to estimate the PSD of a process whose only N samples, i. e. $\{z_n^k\}_{n=1}^{N}$, are known, is formally equivalent to assume an autoregressive (AR) model of order O for the discrete random process z_n^k and then to estimate the set of coefficients $\{c_p\}_{p=1}^{O}$ and the order O from the process known samples $\{z_n^k\}_{n=1}^{N}$. In formula, the AR model can be expressed as:

$$z_n^k = \sum_{p=1}^{O} c_p z_{n-p}^k + e_n, \tag{5.78}$$

where e_n is a discrete, zero-mean, white Gaussian-distributed random process with variance σ_e^2. As discussed in [17], an estimate of the set of coefficients $\{c_p\}_{p=1}^{O}$ and of the noise variance σ_e^2 can be obtained through the recursive Burg's algorithm [18]. Particular attention has to be paid on the choice of the order O of the autoregressive model in eq.

(5.78). An often used criterion for the selection of O is due to Akaike [18], [19] and is known as the Final Prediction Error (FPE). It can be shown that the FPE relative to N samples of an AR model of order O is:

$$FPE(O) = \frac{N+P+1}{N-P-1}\hat{\sigma}_e^2(O), \tag{5.79}$$

where $\hat{\sigma}_e^2(O)$ is the estimated noise variance for a given model order O. The choice of O can be made by applying the Burg algorithm repeatedly to the same group of N samples increasing the model order O up to $N/2$ (in order not to use overlapping data) and selecting the value that minimizes the FPE, i. e.

$$\hat{O} = \arg\min_{O}\{FPE(O)\}. \tag{5.80}$$

However, in literature are present a lot of possible improvement for the Akaike criterion. As an example we refer to [23], [24] and [25].

Now, we can return to the main problem: the estimation of the clutter covariance matrix. It is possible to show that the inverse of the clutter covariance matrix $\hat{\mathbf{M}}^k$ can be directly calculated from the estimated coefficients $\{\hat{c}_p\}_{p=1}^{\hat{O}}$ and from the estimated noise variance $\hat{\sigma}_e^2(\hat{O})$ through the *Gohberg-Semencul formula* [20], [21]. Such formula can be expressed as:

$$\left(\hat{\mathbf{M}}^k\right)^{-1} = \left(\mathbf{A}_1\mathbf{A}_1^H - \mathbf{A}_2\mathbf{A}_2^H\right)\Big/\sigma_e^2, \tag{5.81}$$

where $\mathbf{A}_1$ and $\mathbf{A}_2$ are two triangular $N \times N$ Toeplitz matrices given by:

$$\mathbf{A}_1 = \begin{pmatrix} 1 & & & & & \mathbf{0} \\ c_1 & \ddots & & & & \\ \vdots & \ddots & \ddots & & & \\ c_O & & \ddots & \ddots & & \\ & \ddots & & \ddots & \ddots & \\ \mathbf{0} & & c_O & \cdots & c_1 & 1 \end{pmatrix} \tag{5.82}$$

$$\mathbf{A}_2 = \begin{pmatrix} & & & & \mathbf{0} \\ 0 & & & & \\ c_O & \ddots & & & \\ \vdots & \ddots & \ddots & & \\ c_2 & & \ddots & \ddots & \\ c_1 & c_2 & \cdots & c_O & 0 \end{pmatrix} \tag{5.83}$$

The processing chain for the estimation of the covariance matrix based on the matrix means is shown in fig. 5.6.

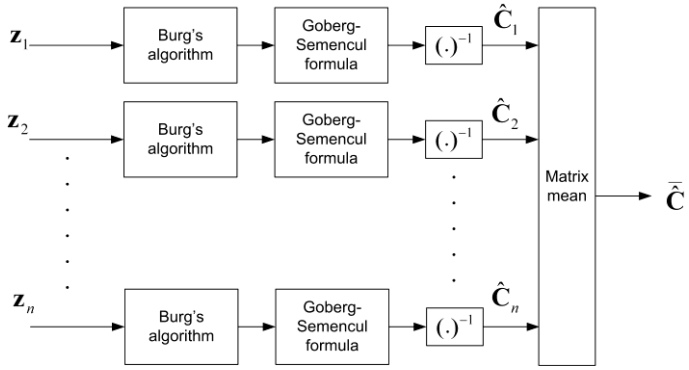

Figure 5.6 Processing chain for the estimation of the covariance matrix based on the matrix means.

5.8.2.2 Euclidean Mean.

The Euclidean mean (EM)-based estimator is the first of three different covariance matrix estimator based on three different definitions of "mean" for a set of covariance matrices. The Euclidean mean-based algorithm assumes an Euclidean (flat) space for the covariance matrices, then the resulting definition of mean is the classical one:

$$\hat{\bar{\mathbf{M}}}_E = \frac{1}{K}\sum_{k=1}^{K}\hat{\mathbf{M}}^k, \tag{5.84}$$

where $\hat{\mathbf{M}}^k$ is the estimated covariance matrix of the *k*th cell obtained as described in subsection B.1.

5.8.2.3 Log-Euclidean Mean.

The Log-Euclidean Mean algorithm is based on the Lie Group structure of the set of the positive-definite Hermitian matrices. The entire procedure to evaluate such mean is addressed in Section 5.6.2. The closed form of the Log-Euclidean Mean is:

$$\hat{\bar{\mathbf{M}}}_{LE} = \exp\left(\frac{1}{K} \sum_{k=1}^{K} \log\left(\hat{\mathbf{M}}^k \right) \right), \tag{5.85}$$

where $\hat{\mathbf{M}}^k$ is the estimated covariance matrix of the *k*th cell. Such estimate can be obtained through the procedure discussed in subsection B.1.

5.8.2.4 Karcher–Fréchet (KF) Mean.

The KF mean algorithm exploits the geometrical structure of the set of the positive-definite Hermitian matrices to get a recursive formula for the evaluation of the average clutter covariance matrices. Such formula can be expressed as follow:

$$\hat{\bar{\mathbf{M}}}_{t+1} = \hat{\bar{\mathbf{M}}}_t^{1/2} \exp\left(\varepsilon \sum_{k=1}^{K} \log\left(\hat{\bar{\mathbf{M}}}_t^{-1/2} \hat{\mathbf{M}}^k \hat{\bar{\mathbf{M}}}_t^{-1/2} \right) \right) \hat{\bar{\mathbf{M}}}^{1/2}, \tag{5.86}$$

where, as before $\hat{\mathbf{M}}^k$ is the estimated covariance matrix of the *k*th cell. In the following, we describe how the gradient descent algorithm in eq. (5.86) is implemented.

- Input data
    1. Set of estimated covariance matrices $\{\hat{\mathbf{M}}^k\}_{k=1}^K$,
    2. Step size ε,
    3. Maximum number of iterations *N_max*,
    4. Stop condition δ.
- Initialization:

In [5], it is shown that a close form solution for the eq. (5.36) can be obtained in the case of $K = 2$ (eq. (5.37)). Then, as starting point, we choose the Riemann barycentre of two estimated matrices, e. g. $\bar{\mathbf{M}}_0 = \hat{\mathbf{M}}^1((\hat{\mathbf{M}}^1)^{-1}\hat{\mathbf{M}}^2)^{1/2}$.

5.8.3 Burg-based estimator

Another covariance matrix estimator can be easily obtained by applying to all the secondary data the procedure used to estimate the clutter covariance matrix for each cell. In fact, under the assumption of clutter homogeneity, we can set up all the secondary data in a unique vector as:

$$\mathbf{z}' = \left((\mathbf{z}^1)^T \quad \dots \quad (\mathbf{z}^k)^T \quad \dots \quad (\mathbf{z}^K)^T \right)^T, \tag{5.87}$$

and then apply exactly the same procedure as described in Section B.1. In this case we have at disposal $N \times K$ complex samples, unlike the previous case where we have only N complex samples. First, we apply the Burg algorithm on the $N \times K$ complex samples to get an estimate of the AR coefficients. The order of the model is selected according to the Akaike information method as before. Finally, through the Gohberg-Semencul formula, we get the estimate of the clutter covariance matrix.

5.8.4 Intrinsic and Flat Cramér-Rao Lower Bounds

In [11], the Intrinsic and the Flat Cramér-Rao Lower Bounds for the covariance matrix estimation is evaluated. Here we recall only the main theorems, for all the mathematical details we refer the fundamental paper [11].

Let $\mathbf{Z} = \left(\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K \right)$ be an $N \times K$ matrix whose columns are independent and identically distributed (iid) zero-mean complex Gaussian random vector with covariance matrix **M**. The pdf of **Z** is:

$$f\left(\mathbf{Z}|\mathbf{M}\right) = \frac{1}{\pi^{NK} \left|\mathbf{M}\right|^K} e^{-\mathrm{tr}\left(\mathbf{Z}\mathbf{Z}^H \mathbf{M}^{-1}\right)}. \tag{5.88}$$

The log-likelihood of this function is (ignoring constants):

$$L\left(\mathbf{Z}|\mathbf{R}\right) = -\mathrm{tr}\left(\mathbf{Z}\mathbf{Z}^H\right) - K \ln\left|\mathbf{M}\right|, \tag{5.89}$$

and the SCM, i. e. $\hat{\mathbf{M}} = K^{-1}\mathbf{Z}\mathbf{Z}^H$, is the maximum likelihood estimate of **M**.

The following theorems show the CRLB in both flat and Riemann metric on the estimate of the covariance matrix from a set of complex data vector **Z**.

Theorem 2 *[11, Theo 5]: The CRLB on the flat distance (i. e. the Frobenius distance) between any unbiased covariance matrix estimator* $\hat{\mathbf{M}}$ *of* $\mathbf{M}$ *is:*

$$\sigma_F \geq \left(\frac{\sum_i [\mathbf{M}]_{ii}^2 + 2\sum_{i<j} [\mathbf{M}]_{ii} [\mathbf{M}]_{jj}}{K} \right)^{\frac{1}{2}}, \tag{5.90}$$

where K is the number of secondary data and $\sigma_F = \sqrt{E\left\{ \left\| \mathbf{M} - \hat{\mathbf{M}} \right\|_F^2 \right\}}$.

Theorem 3 *[11, Theo 4]: The CRLB on the Riemann distance given in* (5.27) *between* $\mathbf{M}$ *and any unbiased covariance matrix estimator* $\hat{\mathbf{M}}$ *of* $\mathbf{M}$ *is:*

$$\sigma_I \geq \frac{N}{\sqrt{K}}, \tag{5.91}$$

where N is the dimension of the data vector, K is the number of secondary data and $\sigma_I = \sqrt{E\{d(\mathbf{M}, \hat{\mathbf{M}})^2\}}$ *is the root mean squared error in Riemann distance.*

5.8.5 Comparison among the covariance matrix estimators performance

The following figures show the comparison among the five estimation algorithms. The comparison is performed in terms of error mean value and RMSE evaluated in both flat and Riemann distances. The numerical values used in the simulations are:

- Dimension of the data vector $N = 16$;
- Numbers of independent Monte Carlo runs is equal to 100.
- The clutter model is an autoregressive model of order 1. The autocorrelation function of such clutter is:

$$r[m] = \sigma_c^2 \rho^{|m|} \tag{5.92}$$

where the clutter power σ_c^2 is assumed to be equal to 1 and $\rho = 0.98e^{-j2\pi 100/PRF}$. The considered PRF is 1000 Hz.

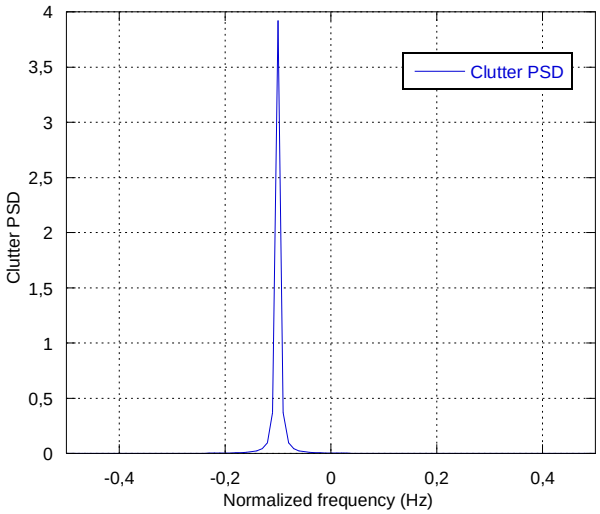

Figure 5.7 Clutter Power Spectral Density.

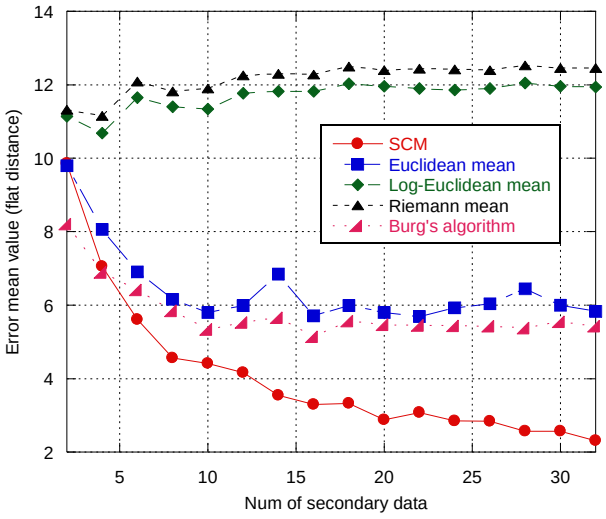

Figure 5.8 Error mean value of the five estimators in flat distance.

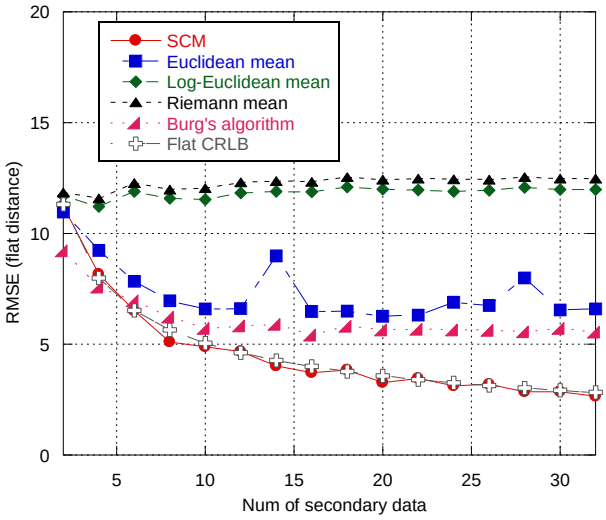

Figure5.9 RMSE and flat $\sqrt{\text{CRLB}}$ of the five estimators in flat distance.

Figures 5.8 and 5.9 show the error mean value and the RMSE of all covariance matrix estimators evaluated with the flat (Euclidean) metric. As we can see, the SCM is (asymptotically) unbiased and efficient estimator with respect to the flat metric. This is an expected result, in fact the SCM is a Maximum Likelihood estimator. The other estimators (Euclidean mean, Log-Euclidean mean, KF mean and the Burg's algorithm based) are biased and not efficient.

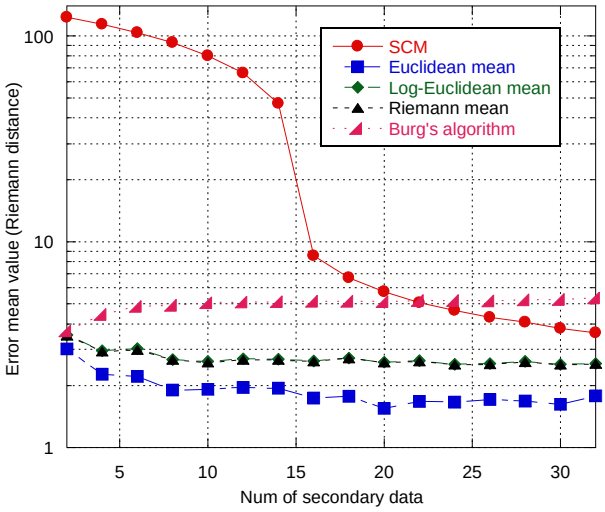

Figure 5.10 Error mean value of the five estimators in Riemann distance.

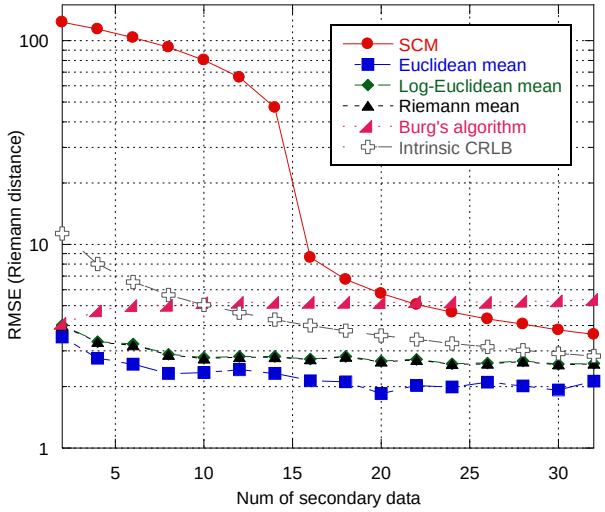

Figure 5.11 RMSE and Intrinsic $\sqrt{\text{CRLB}}$ of the five estimators in Riemann distance.

Figures 5.10 and 5.11 show the error mean value and the RMSE of all covariance matrix estimators evaluated in Riemann metric. In this case, as discussed in [11], the SCM is neither unbiased nor efficient estimator with respect to the Riemann metric. The fundamental difference between the flat distance and the Riemann distance becomes clear

by comparing figs. 5.8 with 5.10 and fig. 5.9 with 5.11. Since the covariance matrix to be estimated has dimension equal to 16×16, if we use a number of sample less than 16 in the SCM estimation, the resulting matrix will be singular. In fact, if we consider the error mean value in Riemann distance (fig. 5.10), for a number of secondary data less than 16, it is very high (theoretically it must be equal to infinity). On the other hands, if we evaluate the error mean value in flat distance (fig. 5.8), there is no discontinuity with respect to the number of secondary data: the error decreases monotonically and in a continuous way. However, the RMSE curves in fig 5.11, except for the one of the SCM estimator, present a strange behaviour: they are under the root of the Intrinsic CRLB.

5.9 Detection performance

In this Section, we compare the performance in terms of ROC curves for four covariance matrix estimation algorithms described in Section 5.8: the classical SCM, the Euclidean matrix mean, the Log-Euclidean matrix mean and the KF mean. As decision test, we use the Kelly's GLRT [13]. The simulated clutter is modelled as an autoregressive process of order 1 as described in Section 5.8.5.

The hypotheses testing model is:

$$\begin{cases} H_0: & \mathbf{z}^k = \mathbf{c}^k + \mathbf{n}^k \\ H_1: & \mathbf{z}^k = \mathbf{s} + \mathbf{c}^k + \mathbf{n}^k \end{cases}, \quad k = 1,\ldots,K, \tag{5.93}$$

where:

- $\mathbf{c}^k$ is the clutter process. To model the complex clutter, we use an autoregressive model of order 1. The autocorrelation function is $r[m] = \sigma_c^2 \rho^{|m|}$ with $\sigma_c^2 = 0.8$ and $\rho = 0.98e^{-j2\pi 100/PRF}$. The considered PRF is 1000 Hz,
- $\mathbf{n}^k$ is the thermal noise, modelled as a white Gaussian discrete random process, independent from $\mathbf{c}^k$ with variance (for each component) of $\sigma_n^2 = 0.2$,
- Size of the data vector: $N = 16$,

- Numbers of independent Monte Carlo runs is equal to 10^3.

Since the clutter and the thermal noise processes are independent, the total noise power is equal to 1. To make the model in eq. (5.93) more realistic, we have to take into account the effects of another disturbance: the radio frequency interference (RFI). The RFIs, also called outliers, can be modelled as:

$$\mathbf{o}^k = \beta^k \mathbf{p}\left(f_{d,o}^k \right), \tag{5.94}$$

where β^k is a Gaussian random variable, with zero mean and variance equal to σ_β^2, $\mathbf{p}(f_{d,o}^k)$ is the steering vector defined for a normalized Doppler frequency $f_{d,o}^k$ generated as a random variable uniformly distributed in the interval [-0.5;0.5]. In our simulation, we added 4 outliers with a power five times bigger than the clutter power. The specific range cells in which the outliers are located, are chosen randomly. For a fixed signal-to-noise ratio (in our simulation, we set SNR = 10dB), we evaluate the ROC curves for 2*N* (figs. 5.12 and 5.13) and N (figs. 5.14 and 5.15) secondary data, where *N* is the dimension of the snapshot for each range cell. The simulations show that:

1. The ROC curves relative to the matrix mean-based estimators (Euclidean mean, Log-Euclidean mean and KF mean) are always higher than the ROC curves relative to the SCM. This means that, given a value of PFA, using a matrix mean based covariance matrix estimation algorithm, it is possible to reach a higher value of PD.
2. The matrix mean-based estimators are robust with respect to the number of secondary data used to estimate the clutter covariance matrix. Figs 5.14 and 5.15 show that the gap is very high when the number of secondary data is lower than the one prescribed by the "RMB condition" discussed in [14].
3. The ROC curves relative to the Log-Euclidean mean and to the KF mean are almost identical.
4. The Log-Euclidean mean and the KF mean are more robust with respect to the presence of outliers than the Euclidean mean. This is an expected result, because both the Log-Euclidean mean and the KF mean can be interpreted as a sort of geometric mean for a matrix space.

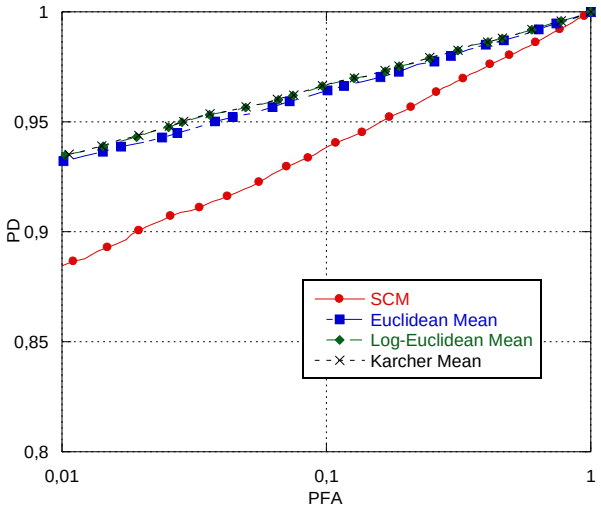

Figure 5.12 Comparison among different covariance matrix estimation algorithms without outliers with 2K secondary data with SNR=10dB.

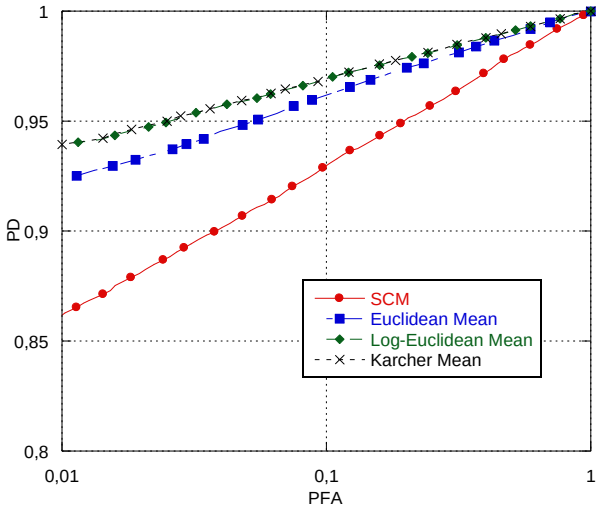

Figure 5.13 Comparison among different covariance matrix estimation algorithms in presence of outliers with 2K secondary data with SNR=10dB.

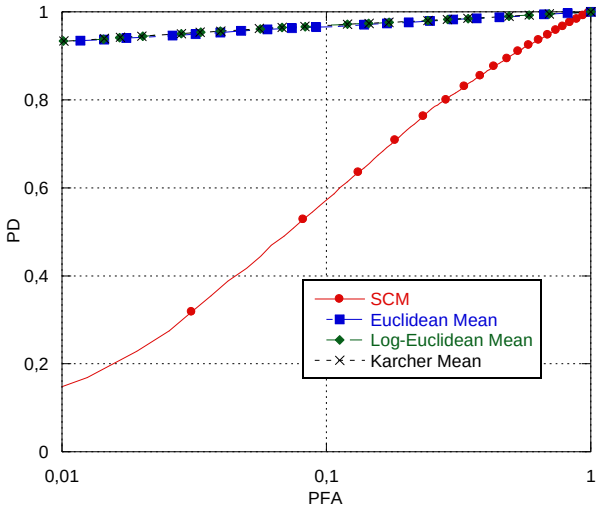

Figure 5.14 Comparison among different covariance matrix estimation algorithms without outliers with K secondary data with SNR=10dB.

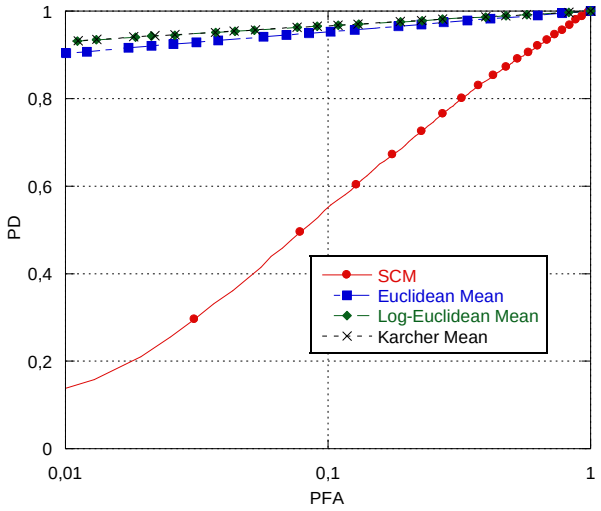

Figure 5.15 Comparison among different covariance matrix estimation algorithms in presence of outliers with 2K secondary data with SNR=10dB.

5.10 Summary

In this Chapter, a new approach to the radar target detection and clutter suppression problem is described. This new approach is based on a non-Euclidean description of the geometry of the of covariance matrices. First, we proved that the KF mean is a Maximum Likelihood estimator for a set of i. i. d. Gaussian distributed random matrices. Then, five different covariance matrix estimators are described and compared with both the Flat and Intrinsic Cramér-Rao Lower Bounds. Finally, the performance in terms of ROC curves for such covariance matrix estimators are evaluated using the Kelly's GLRT as decision rule. However, the present work is only a preliminary study on the geometrical approach to target detection and a lot of work has to be done. In particular, it will be very important to understand the behaviour of the RMSE curves with respect the Intrinsic CRLB.

References

[1] F. Barbaresco, "New foundation of radar Doppler and array processing based on German mathematical works: geometry of metric spaces with negative curvature and Von-Mangoldt-Cartan-Hadamard manifolds," *IRS 2009*, Hamburg, Germany, 09-11 September, 2009, pp. 487-491.

[2] J. Lapuyade-Lahorgue, F. Barbaresco, "Radar detection using Siegel distance between autoregressive processes, application to HF and X-band radar," *IEEE Radar Conference, 2008,* Rome, Italy, pp. 1-6, 26-30 May 2008.

[3] F. Barbaresco, "New foundation of radar Doppler signal processing based on advanced differential geometry of symmetric spaces : Doppler matrix CFAR & radar application", *International Radar Conference 2009*, Bordeaux, France, 12-16 October 2009.

[4] X. Pennac, P. Fillard, and N. Ayache, "A Riemann framework for tensor computing," *International J. Computer Vision*, vol. 66, pp. 41-66, 2006.

[5] M. Moakher, "A differential geometry approach to the geometric mean of symmetric positive definite matrices, "*SIAM J. Matrix Anal. Appl.*, vol. 26, pp. 735-747, 2005.

[6] H. Karcher, "Riemaniann center of mass and mollifier smoothing," *Commun. Pure Appl. Math.*, vol. 30, pp. 509-541, 1977.

[7] J. D. Lawson and Y. Lim, “The geometric mean, matrices, metrics and more”, *Amer. Math. Monthly*, vol. 108, pp. 797-812, 2001.

[8] V. Arsigny, P. Fillard, X. Pennac, and N. Ayache, "Geometric Means in a Novel Vector Space Structure on Symmetric Positive-Definite Matrices, " *SIAM J. Matrix Anal. Appl.*, vol. 29, pp. 328-347, 2007.

[9] V. Arsigny, X. Pennac, and N. Ayache, "Bi-Invariant Means in Lie Group. Application to Left-Invariant Polyaffine Transformations," Research Report n. 5885 of the Institut National de Recherche en Informatique at en Automatique (INRIA), April 2006. Available at: http://hal.inria.fr/docs/00/07/13/83/PDF/RR-5885.pdf.

[10] M. Moakher, "On the Averaging of Symmetric Positive-Definite Tensors", *Journal of Elasticity*, vol. 82, no. 3, pp. 273-296, March 2006.

[11] S. T. Smith, "Covariance, subspace, and intrinsic Cramér-Rao bounds," *Signal Processing, IEEE Transactions on* , vol.53, no.5, pp. 1610- 1630, May 2005.

[12] L.E. Brennan, L.S. Reed, "Theory of Adaptive Radar," *Aerospace and Electronic Systems, IEEE Transactions on* , vol.9, no.2, pp.237-252, March 1973.

[13] E. J. Kelly, "An Adaptive Detection Algorithm," *Aerospace and Electronic Systems, IEEE Transactions on* , vol. 22, no.2, pp.115-127, March 1986.

[14] I. S. Reed, J. D. Mallett, L. E. Brennan , "Rapid Convergence Rate in Adaptive Arrays," *Aerospace and Electronic Systems, IEEE Transactions on* , vol. 10, no.6, pp.853-863, Nov. 1974.

[15] S. Fiori, T. Tanaka , "An Algorithm to Compute Averages on Matrix Lie Groups," *Signal Processing, IEEE Transactions on* , vol.57, no.12, pp.4734-4743, Dec. 2009.

[16] A. Farina, *Antenna-Based Signal Processing Techniques for Radar Systems*, Artech House Publishers, January 1992.

[17] E. D’Addio, A. Farina, F. A. Studer, "The maximum entropy method and its application to clutter cancellation," *Rivista Tecnica Selenia*, vol. 8, no. 3, pp. 15-24, 1983.

[18] S. M. Kay and S. L. Marple, "Spectrum analysis. A modern prospective", *Proc. of the IEEE*, vol. 69, no. 11, pp. 1380-1419, November 1981.

[19] H. Akaike, "Fitting autoregressive models for prediction", *Ann. Inst. Stat. Math.*, vol. 21, 1963.

[20] J. P. Le Cadre, "Parametric methods for spatial signal processing in the presence of unknown colored noise fields," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 37, no. 7, pp. 965-983, July 1989.

[21] S. Haykin and A. Steinhardt, *Adaptive Radar Detection and Estimation*, Wiley-Interscience, June 1992.

[22] X. Pennac, "Probabilities and statistics on Riemannian manifolds: a geometric approach", Research Report n. 5093 of the Institut National de Recherche en Informatique at en Automatique (INRIA), April 2006.

Available at: http://hal.inria.fr/docs/00/07/14/90/PDF/RR-5093.pdf.

[23] S. de Wade and P. M. T. Broersen, "Order selection for vector autoregressive models," *Signal Processing, IEEE Transactions on*, vol. 51, pp. 427-433, Feb. 2003.

[24] P. M. T. Broersen, "Finite sample criteria for autoregressive order selection," *Signal Processing, IEEE Transactions on*, vol. 48, pp. 3550-3558, Dec. 2000.

[25] P. M. T. Broersen, "Automatic spectral analysis with time series models," *IEEE Trans. Instrum. Meas.*, vol. 51, pp. 211-216, April 2002.

Concluding remarks

In this dissertation, a notional multi-sensor system acting in a maritime border control scenario for Homeland Security (HS) has been considered and two important aspects investigated. In the first part of the work, we focused on a particular sensor in the system, i.e. the airborne radar. The tracking performance of this sensor has been investigated in the presence of the atmospheric turbulence and the losses in tracking accuracy has been evaluated. The simulations have shown that the atmospheric turbulences cause a degradation of the tracking performance for the estimate of the target velocity vector, while the losses in the estimate of the target position vector are almost negligible. A modification of the classical Kalman Filter (KF) equations is then provided in order to mitigate the effects of propagation errors due to tropospheric refraction. Simulation results have shown the effectiveness of the proposed algorithm and its ability to mitigate the effects of the tropospheric errors on the estimated target state vector. In addition, it seems to be robust with respect to the random variations of the tropospheric refractive index. In the second part of the work, the fundamental problem of sensor registration, or grid-locking problem, has been investigated. First, the relative grid-locking problem has been accurately analyzed and two different estimation algorithms have been provided, a linear least squares (LS) algorithm and an Expectation-Maximization (EM) algorithm. Moreover, a performance bound, the Hybrid Cramér-Rao lower bound (HCRLB) has been evaluated in order to assess the efficiency of the proposed algorithm. Both a single target scenario and a multi-target scenario are analyzed. The numerical results have shown that the EM algorithm generally outperforms the linear LS algorithm, even if it is not always tight to the HCRLB. Also the identifiability problem for the sensor registration has been considered and some new results derived. Then, a generalization to the absolute grid-locking problem of both the algorithms has been provided. However, such generalization is not so straightforward and some additional work is needed in order to definitely solve the absolute grid-locking problem. Finally, a theoretical fundamental problem for a huge quantity of practical applications has been taken into account in this dissertation, i.e. the problem of adaptively estimating the disturbance covariance matrix. Recently, a new geometrical concept has been applied to this particular estimation problem, the Riemann geometry. An overview on the state of the art of the application of the Riemann geometry for the covariance matrix estimation has been described and compared with both the Flat and the Intrinsic Cramér-

Rao Lower Bounds. Finally, the performance in terms of Receiver Operating Characteristics (ROC) curves for such covariance matrix estimators are evaluated using the Kelly's GLRT as decision rule. However, there is a lot of work left to do on this topic and many aspects (for example, the behaviour of the Mean Square Error of the covariance matrix estimators with respect to the Flat and the Intrinsic CRLB) is still not clear and need to be further investigated.