Fall 12-20-2013

# Estimation, Decision and Applications to Target Tracking

Yu Liu
*university of new orleans*, lyu2@uno.edu

# Estimation, Decision and Applications to Target Tracking

A Dissertation

Submitted to the Graduate Faculty of the
University of New Orleans
in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy
in
Engineering and Applied Science

by

Yu Liu

B.S. Xian Jiaotong University, 2006
M.S. Xian Jiaotong University, 2008

December 2013

# Acknowledgements

I would like to express my sincerest appreciation to a number of people without whom this dissertation would not have been possible.

To my parents, my wife and my daughter, who continuously support and encourage me throughout my study and research.

To my advisor, Dr. X. Rong Li, who is always a source of knowledge and inspiration for me.

To Dr. Huimin Chen, Dr. Vesselin Jilkov and Dr. Linxiong Li, who also provide me invaluable and insightful thinking and guidance during my research.

To my committee members, Dr. Tumulesh Solanky and Dr. Dongming Wei, for their great effort and help on my dissertation.

To my colleagues and friends, Dr. Zhansheng Duan, Dr. Xiaomeng Bian, Dr. Meiqin Liu, Dr. Ji Zhang, Dr. Jian Lan, Dr. Xianghui Yuan, Dr. Deqiang Han, Ms. Jing Hou, Mr. Jiande Wu, Mr. Gang Liu and Mr. Rastin Rastgoufard, without whom, my life during the PHD study in UNO would become much harder.

Without these people, I can hardly make this journey. I dedicate this dissertation to them.

Further, my research was supported in part by NASA/LEQSF(2013-15)-Phase3-06 through grant NNX13AD29A, ONR-DEPSCoR through grant N00014-09-1-1169 and ARO through grant W911NF-08-1-0409.

# Contents

# Abstract

This dissertation mainly consists of three parts. The first part proposes generalized linear minimum mean-square error (GLMMSE) estimation for nonlinear point estimation. The second part proposes a recursive joint decision and estimation (RJDE) algorithm for joint decision and estimation (JDE). The third part analyzes the performance of sequential probability ratio test (SPRT) when the log-likelihood ratios (LLR) are independent but not identically distributed.

The linear minimum mean-square error (LMMSE) estimation plays an important role in nonlinear estimation. It searches for the best estimator in the set of all estimators that are linear in the measurement. A GLMMSE estimation framework is proposed in this dissertation. It employs a vector-valued measurement transform function (MTF) and finds the best estimator among all estimators that are linear in MTF. Several design guidelines for the MTF based on a numerical example were provided.

A RJDE algorithm based on a generalized Bayes risk is proposed in this dissertation for dynamic JDE problems. It is computationally efficient for dynamic problems where data are made available sequentially. Further, since existing performance measures for estimation or decision are effective to evaluate JDE algorithms, a joint performance measure is proposed for JDE algorithms for dynamic problems. The RJDE algorithm is demonstrated by applications to joint tracking and classification as well as joint tracking and detection in target tracking.

The characteristics and performance of SPRT are characterized by two important functions—operating characteristic (OC) and average sample number (ASN). These two functions have been studied extensively under the assumption of independent and identically distributed (i.i.d.) LLR, which is too stringent for many applications. This dissertation relaxes the requirement of identical distribution. Two inductive equations governing the OC and ASN are developed. Unfortunately, they have non-unique solutions in the general case. They do have unique solutions in two special cases: (a) the LLR sequence converges in distributions and (b) the LLR sequence has periodic distributions. Further, the analysis can be readily extended to evaluate the performance of the truncated SPRT and the cumulative sum test.

**Keywords:** nonlinear estimation, linear minimum mean-square error estimation, joint decision and estimation, target tracking, sequential probability ratio test, performance evaluation

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Statistical Inference in Target Tracking

Estimation and decision are two key components in target tracking. For example, inferring target state (e.g., position and velocity) is clearly an estimation problem, and determining the number of targets, target attributes, maneuvering onset and termination, etc., are decision processes.

Although the optimal (point) estimator, i.e., Kalman filter [110,111], for the linear Gaussian system were develop more than five decades ago, nonlinear estimation is still under active research. Nonlinear estimation (filtering) is to infer the interested quantities of a nonlinear stochastic process based on observations. Basically, the difficulty comes from (and increases with the level of) nonlinearity, which is an intrinsic nature of many problems, including a large number of tracking applications. Nonlinear estimation can be classified into two categories:

(a) Point estimation: Only the moments (usually the first and second moments) of the quantity to be estimated are of interest.

(b) Density estimation: The whole probability density or distribution function of the interested quantity is needed. This is obviously more demanding, both technically and computationally, than point estimation.

Since most target tracking applications are point estimation problems, methods for density estimation, e.g., particle filtering and the probability hypothesis density (PHD) filter are not considered in this dissertation. Interested readers may see the surveys of density-estimation based nonlinear filtering techniques [13, 53, 54, 81, 151, 154–156, 225] for details.

Sequential tests are widely used in target tracking, e.g., maneuver detection [168,211] and counter-measurement detection [87]. The performance of different sequential tests has been studied extensively for independent and identical observations. However, this assumption is hardly satisfied in target tracking due to the non-stationary nature of the problem. In practice, the log-likelihood ratios based on innovations may be independent or weakly coupled, however, they are definitely not identically distributed in general. In this dissertation, the performance analysis of sequential tests with independent but non-stationary observations is studied.

Further, as explained later, estimation and decision are often tightly correlated in some tracking applications (e.g., joint tracking and detection, and joint tracking and classification), making them tough and open problems. This is the so-called joint decision and estimation (JDE) problem. To be clear, we interpret estimation as inferring a continuous-valued (random or non-random) quantity, while decision is to make a choice from a discrete candidate set. Inference of the target state and determination of the target attribute or the number of targets often affect each other. The inter-dependence between estimation and decision often incurs additional difficulty for solving JDE problems. For example, without knowing the class or the number of the targets, it is hard to estimate the target state. On the other hand, good decision relies heavily on accurate estimation of the target state. Conventional solutions ignore this inter-dependence either completely (e.g., separate estimation and deci-

sion) or partially (e.g., decision-then-estimation or estimation-then-decision), making their performance suffer.

## 1.2 Research Motivations

The linear minimum mean-square error (LMMSE) estimation plays a key role in nonlinear estimation [144]. Many widely used nonlinear filters are based on LMMSE estimation. It searches for the best estimator in the set of all estimators that are linear in the measurement. But it may not perform well for a highly nonlinear problem. A generalized LMMSE (GLMMSE) estimation framework is proposed in this dissertation. It employs a vector-valued measurement transform function (MTF) and finds the best estimator among all estimators that are linear in the MTF, rather than in the measurement itself. The MTF introduces more flexibility to GLMMSE. It can choose a larger or more appropriate candidate set of estimators (than the set in LMMSE) for searching, and hence the best one in the set would lead to performance superior to that of LMMSE estimation. Several design guidelines for the MTF are provided to facilitate the design process. Further, similar to LMMSE estimation, moments involved in GLMMSE estimation are difficult to evaluate exactly in general. Fortunately, many numerical approximations for LMMSE estimation are also applicable to GLMMSE estimation. An approximation of GLMMSE estimation based on the Gaussian-Hermite quadrature is provided. Our GLMMSE estimation is demonstrated by applications to radar tracking and multi-target tracking.

For the JDE problem, a JDE algorithm based on a generalized Bayes risk [143, 158] was proposed. It is a batch algorithm and thus computationally inefficient or infeasible for many dynamic JDE problems where data are made available sequentially, e.g., in target tracking. Therefore, following the same JDE framework, a recursive version of the JDE algorithm (RJDE) is proposed, which fits dynamic JDE problems more naturally and inherits JDEs theoretical superiorities. Further, a joint performance measure in the measurement space is

developed for evaluation of dynamic JDE algorithms. RJDE is demonstrated by numerical examples in target tracking.

Performance analysis of sequential tests is also studied in this dissertation. It has been extensively studied if the log-likelihood ratios (LLR) are independent and identically distributed (i.i.d.). We generalize the analysis to the case with an independent but non-stationary LLR sequence. This situation is frequently encountered in application. For example, if SPRT is implemented based on the innovations of the Kalman filter [110], e.g., in target maneuver detection, the LLRs are approximately independent or weakly coupled, but their distributions at different time instants are clearly different in general. Or some parameters in the distribution of LLR may be periodically varying, rendering the LLRs not identically distributed. Once the LLR sequence is not identically distributed, the performance analysis becomes much more complicated than the i.i.d. case due to the loss of stationarity of the LLR sequence.

## 1.3 Thesis Outline

This thesis, consisting of five chapters, is organized as follows.

Chapter 1 introduces the background and motivation.

Chapter 2 briefly surveys the related work, including nonlinear estimation methods, JDE algorithms and multi-target tracking methods.

Chapter 3 proposes the GLMMSE estimation for nonlinear estimation.

Chapter 4 presents an RJDE algorithm for dynamic JDE problems.

Chapter 5 proposes methods for performance analysis of sequential tests.

Chapter 6 draws conclusions and discusses future work.

# Chapter 2

# Related Work

Existing work on nonlinear point estimation, JDE algorithms and performance analysis of sequential tests are briefly surveyed and discussed in this chapter.

## 2.1 Nonlinear Filtering

Existing nonlinear filters are briefly surveyed in this section, which is largely based on [144, 151]. As mentioned before, density-estimation based algorithms are not considered in this section. In general, a stochastic system may be modeled in one of the following three forms:

(a) Continuous-time: Both the dynamic and measurement models are in continuous time; that is, the system is modeled by differential equations.

(b) Discrete-time: Both the dynamic and measurement models are in discrete time; that is, the system is modeled by difference equations.

(c) Mixed-time: One of the dynamic and measurement models is in continuous time and the other in discrete time. For target tracking, due to the continuity of the target trajectory and discreteness of the measurements, the mixed time model is more appropriate since it fits the truth better than the other representations.

Continuous-time nonlinear filtering are largely of only theoretical value. A survey of nonlinear filtering techniques for mixed-time system can be found in [48]. Due to the widespread use of digital computers, however, most tracking filters have been developed for discrete-time systems, particularly in a recursive form. Also, discrete-time nonlinear filtering is significantly easier than filtering in continuous time or mixed time. Therefore, we only consider discrete-time systems with additive, mutually independent white process noise and white measurement noise:

$$x_{k+1} = f_k(x_k) + w_k \tag{2.1}$$

$$z_k = h_k(x_k) + v_k \tag{2.2}$$

where $x_k,\ z_k$ are the state to be estimated and the measurement, respectively, at time $k$; $w_k \sim (\bar{w}_k, Q_k)$ (mean and covariance) and $v_k \sim (\bar{v}_k, R_k)$.

### 2.1.1 Optimal Bayesian Estimation

It is well known that the conditional mean $\hat{x}_k = E[x_k|Z^k]$ ($Z^k$ is all measurements through time $k$) is the optimal estimator that minimizes mean-square errors (MSE), namely, the minimum mean-square error (MMSE) estimator. The equation for $\hat{x}_k$ for the discrete-time system (2.1)–(2.2) with Gaussian white process and measurement noises was given in [197]. In general, evaluating $\hat{x}_k$ requires the entire distribution of $x_k$, which cannot be compressed into a finite-dimensional sufficient statistic. Further, the corresponding MSE of $\hat{x}_k$ is not available. However, [197] provides a theoretical basis for approximation techniques.

### 2.1.2 Linear Minimum-Mean Square Error Estimation

Instead of searching for the best of all estimators, the best of all linear estimators (i.e., $\hat{x}_k = a + BZ^k$) may be a good choice, resulting in the linear minimum mean-square error

(LMMSE) estimator

$$\hat{x}^{\mathrm{LMMSE}} \triangleq \arg \min_{\hat{x}=a+BZ^k} \mathrm{MSE}(\hat{x})$$

It often makes a good compromise between the simplicity and performance. It is unbiased (i.e., $E[\hat{x}^{\mathrm{LMMSE}}] = E[x]$) and the estimation error $\tilde{x}^{\mathrm{LMMSE}}$ is orthogonal to the space spanned by the measurements (i.e, $E[\tilde{x}^{\mathrm{LMMSE}} z'] = 0$). The well-known Kalman filter (KF) is a special case of the recursive LMMSE estimator when both (2.1) and (2.2) are linear. Note that not every batch estimator can be written in an equivalent recursive form (see [142] for details). Recursive LMMSE estimation was successfully applied to radar tracking [58, 252], where the target dynamics is linear and measurement model is nonlinear.

Since the expectations and covariances involved in the MMSE and LMMSE filters are difficult to evaluate analytically, various techniques have been applied to approximate the optimal filter and more often the LMMSE filter. Most of these techniques can be broadly classified into the following three categories [151]:

(a) Deterministic function approximation: Nonlinear functions, which are most often the integrands of expectations encountered in nonlinear filtering, are approximated deterministically by functional approximation, such as the Taylor series expansion (TSE) or interpolation.

(b) Moment approximation: The integral values, particularly the first- and second-moments, are approximated directly by, e.g., deterministic or Monte Carlo sampled values. Here, the unscented transform (UT) is a representative, leading to the unscented filter.

(c) Stochastic model approximation: The original nonlinear stochastic system is approximated by a simpler (often linear) model optimally in terms of some statistical measure, resulting in the so-called stochastic linearization or statistical linearization.

Some widely used filters in these categories are presented in the following.

### 2.1.3 Extended Kalman Filters

#### First-Order Extended Kalman Filter

The first-order extended Kalman filter (EKF) was developed for nonlinear filtering in engineering applications [178, 231] right after the KF came to existence. EKF is now the most widely used nonlinear filter due to its simplicity and generality. The first-order EKF is obtained by approximating the $f_k(x_k)$ and $h_k(x_k)$ in (2.1)–(2.2) with the first-order TSE at the latest estimates and applying the standard KF to the resulting linear system. It is hence a deterministic function approximation.

#### Second-Order EKF

The performance of the first-order EKF may be improved by employing higher-order TSE approximations (but it might not work in some cases [29, 52, 123]). The second-order EKF follows this path and it differs from the first-order EKF only in the state and measurement predictions as well as their error covariances, obtained based on second-order TSE. It has exactly the same filter update as the first-order EKF. See [22] for algorithm details, [179,248] for performance comparisons, and [179,193] for some simplifications. EKF of an even higher order is almost never used in practice.

Following a similar idea, the truncated second-order filter [24, 95] and the Gaussian second-order filter [15, 177] were proposed. Instead of approximating the nonlinear functions $f_k(x_k)$ and $h_k(x_k)$ in (2.1)–(2.2), the truncated second-order filter approximates the nonlinear functions involved in the conditional mean by a second-order TSE. The Gaussian second-order filter works in the same way as the truncated second-order filter except that the fourth moment is accounted and computed based on a Gaussian assumption. See [84,96,234] and the reference therein for more details. Some corrections and modifications were made for these filters in [83, 151].

#### Iterated EKF

TSE would approximate the nonlinear function better and improve the performance of EKF if the expansion point $\hat{x}$ is close to the true $x$. Following this idea, the iterated EKF was proposed [96], which re-linearizes the measurement model by the updated state. This makes sense since the state update should be more accurate than the prediction. This re-linearization and state update can be carried out iteratively to further improve the accuracy. The convergence of the iteration is studied in [28].

#### Perturbed Kalman Filters

The perturbed Kalman filter (PKF) works similarly as the first-order EKF. It linearizes the system at some "nominal" or reference state trajectory (instead of the latest estimate). Like the first-order EKF, the PKF has the performance relying significantly on the accuracy of the expansion point, which can be predetermined or generated online.

The major drawbacks of the TSE based filters are:

(a) They hinge on the characteristics of the nonlinear function at a single point (i.e., the expansion point). If the true state is not in a small neighborhood of this point, their performance suffers.

(b) Evaluating the function derivatives is undesirable or infeasible for some cases.

### 2.1.4 Unscented Filters

The unscented filter (UF) was first proposed in [100, 104, 107] relying on the unscented transform (UT), where a set of $\sigma$-points are generated to approximate the first- and second-moments of a nonlinear transform to the accuracy of at least second order. The approximate moments can be plugged into the LMMSE filter, leading to the UF. Due to the large freedom in generating the sigma points in UT, many designs of the sigma points are available. A set of $2n_x + 1$ sigma points was proposed in [100,104,107] and widely used, where $n_x = \dim(x)$.

Using $2n_x^2+1$ points was proposed in [103] to match some higher moments for Gaussian and some other distributions. Also, a minimum number $n_x+1$ of sigma points were designed in [105] for better computational efficiency. Robust designs were exploited in [102,251]. Two more scaling parameters were introduced in [101,106] to mitigate the diffusion of the sigma points as the state dimension increases.

### 2.1.5 Divided Difference Filters

Instead of putting all its eggs in one basket as the TSE based approximation does, interpolation based algorithms approximate a nonlinear function using multiple points and avoid derivatives by replacing them with divided differences. Following this path, the first-order (DD1) and second-order (DD2) divided difference filters were developed in [196] based on a multivariate extension of Stirling's interpolation formula [201]. The nonlinear function in (2.1)–(2.2) can be approximated by this interpolation. Therefore, the first and second moments can be computed for LMMSE estimation, leading to the DD1 or DD2 filter.

Note that the DD2 and UF use the same procedure but different sample points. Determining design parameters was discussed in [195, 196]. Contrary to the TSE-based approximation, the interpolation-based approximation depends on multiple points in a region and does not require evaluation of function derivatives. Performance evaluation of DD1 was included in [129], compared with several KF based filters. See [3,253] for some applications.

### 2.1.6 Central Difference Filters

The central difference filter (CDF) [218,219] follows a similar idea of the first-order EKF but with the Jacobian matrix replaced by the first-order central difference to avoid evaluation of derivatives. As shown in [219], CDF has a slightly better approximation of the covariance matrix for symmetric distributions than the first-order TSE. The DD1 is actually a refined version of CDF due to the flexibility of some design parameter. Another version of CDF

based on the second-order central difference was proposed in [94] and it is basically equivalent to the DD2 filter.

### 2.1.7 Gaussian Filters

The Gaussian filter or Gaussian-Hermite filter was proposed in [7, 94]. It relies on the Gaussian-Hermite Quadrature (GHQ) and successive approximation of the probability densities needed by moment-matched Gaussian densities. All the moments involved in LMMSE are approximated by the GHQ. Instead of a single Gaussian, the density can be successively approximated by a Gaussian mixture to improve the accuracy. This leads to the Gaussian sum filter or mixed Gaussian filter [94].

GHQ suffers from a "curse of dimensionality." Many methods are available to mitigate, to some extent, this effect [42, 238], such as lattice rules [230], sparse grids [70, 232], Monte Carlo [164] and quasi-Monte Carlo methods [194]. In [6], the integral in the Gaussian filter was transformed from the Cartesian coordinate system to a spherical-radial integration form [184] by change of variables. A third-order spherical-radial cubature rule was applied to compute the integral, leading to the so-called cubature Kalman filter (CKF) [4, 8, 79, 200]. In [5, 6], a numerically robust implementation of the Gaussian filter was proposed, which computes the means and square-root of the covariance matrices at the cost of increased computational complexity.

### 2.1.8 Stochastic Model Approximation

The deterministic function approximation approach approximates the function in a small region. They usually incur large errors for large deviations and have no optimality in general. Since the system is stochastic and the state $x$ is random, the nonlinear function $g(x)$ involved may be better approximated by a linear system optimally in some stochastic sense, leading to the stochastic linearization [69, 235]. It accounts for large errors stochastically and should be superior for the specific distribution of $x$.

The major difficulty with the optimal stochastic linearization is to evaluate the expectations involved. Replacing these expectations by their sample versions results in the statistical linearization [17, 62, 63, 89, 130]. Techniques covered before, such as the UT, central difference, and Gaussian-Hermite quadrature are all applicable to this expectation evaluation. In [128, 129], the linear model was obtained by linear regressions, leading to the linear regression Kalman filter (LRKF).

### 2.1.9 Two-Step Filters

The two-step filter proposed by [77, 78] was derived based on the two-step least-squares (LS) estimation. For a nonlinear LS problem,

$$\hat{x}^{\mathrm{LS}} = \arg\min_x [z - h(x)]' R_k^{-1} [z - h(x)]$$

it may be solved optimally in two steps:

- Find a transform $y = g(x)$ such that $h(x) = Hy$. Then the optimal LS estimate $y^{\mathrm{LS}}$ and its covariance matrix $\Sigma_y^{-1}$ of $y$ can be determined by linear LS solution.
- The optimal estimate $\hat{x}^{\mathrm{LS}}$ is the solution of a nonlinear LS problem:

$$\hat{x}^{\mathrm{LS}} = \arg\min_x [\hat{y}^{\mathrm{LS}} - g(x)]' \Sigma_y^{-1} [\hat{y}^{\mathrm{LS}} - g(x)] \tag{2.3}$$

The key to the two-step filter is to find the transform $y_k = g_k(x_k)$ and $h_k(x_k) = H_k y_k$ for system (2.1)–(2.2). Applications of the two-step filter can be found in [74, 98].

## 2.2 Joint Decision and Estimation Algorithms

The mutual dependence between decision and estimation in a JDE problem makes it difficult to solve. [180] first proposed an integrated framework to solve the "simultaneous signal

detection and estimation" problem. A solution for the multiple-hypothesis case was given by [66]. As pointed out in [242], their solution is estimation-oriented. Both estimation-oriented and decision-oriented methods were provided in [109] for a specific example. More recently, an integrated methods for simultaneous signal detection and estimation under false alarm constraints was proposed in [27]. However, their "estimation" actually means classification, which belongs to decision in our terminology.

In general, the existing methods for solving JDE problems can be classified into four categories [143, 158, 167], which are explained next.

### 2.2.1 Decision-then-Estimation

It tries to make the best decision from the data first, and estimation is obtained based on this decision as if it were certainly correct. This is the most natural way of thinking and the majority of the JDE problems are solved in this way. However, although its major drawbacks are obvious—the possible decision errors are completely ignored by the estimation process and the decision is made regardless of the estimation accuracy it would lead to—an effective remedy is hard to come by within this two-stage framework. Most algorithms for multi-target tracking (MTT) with an unknown number of targets follow this method—determining the number of targets first, and then estimating the target state based on the determined number of targets.

### 2.2.2 Estimation-then-Decision

Some JDE algorithms, e.g., the generalized likelihood ratio test (GLRT) and the marginalized likelihood ratio test (MLRT), follow this strategy. It estimates the state (or the unknown parameters) first and decision is made based on this estimation. It has some inherent flaws and will not work well if the estimation depends heavily on the decision or the estimation is not secondary in the problem.

### 2.2.3 JDE Based on Density Estimation

The cornerstone of this type of methods is the estimation of the mixed density-probability of estimation and decision

$$\{f(\{x, D\}_1|Z), f(\{x, D\}_2|Z), \cdots, f(\{x, D\}_i|Z)\}$$

which can be inferred by a density estimation algorithm (e.g., particle filter or numerical integration). Here, $Z$ is the available data, $D_i$ the $i$th decision and $x$ the corresponding estimate. For a JDE problem, in general this hybrid quantity has the form $\{x, D\}_i \triangleq \{x(D_i), D_i(x)\}$, which signifies explicitly the mutual dependence between $x$ and $D$. This method was applied to target tracking and classification in [39, 85], and in [90] to multitarget tracking with an unknown number of targets. Although this posterior mixed density-probability may be a complete Bayes solution for JDE problems, it is not considered in this dissertation due to the reasons given in [167].

### 2.2.4 JDE Based on Generalized Bayes Risk

The foundation of this method is a novel Bayes risk [143, 158, 250], which is a generalization of those for decision and estimation respectively. It has the potential of arriving at the globally optimal solution. The generalized Bayes risk, which explicitly considers the inter-dependence between decision and estimation, is theoretically superior to the conventional two-stage methods (i.e., decision-then-estimation and estimation-then-decision) and the method of separate decision and estimation. In this framework, decision cost and estimation cost are converted to a unified measure by additional weight coefficients and hence the final results depend on both decision and estimation performances. The power of this JDE method was elaborated in [143,158] by several challenging JDE applications. However, the currently available JDE method is a batch process, which does not fit dynamic problems well since in many cases, e.g., multi-target tracking, the data are coming sequentially. A

batch algorithm has to start from scratch when new data arrive. Hence, a recursive JDE algorithm is desired for dynamic problem.

## 2.3 Performance Analysis of Sequential Tests

The operating characteristic (OC) and average sample number (ASN) of the sequential probability ratio test (SPRT) and the average run length (ARL) of the cumulative sum test (CUSUM) reveal the performance of these sequential tests. Methods for evaluating these functions are briefly summarized as follows

### 2.3.1 OC and ASN for SPRT

Two important functions—OC and ASN—characterize the behavior of SPRT. The existing methods proposed to evaluate these two functions are almost all based on the assumption of an i.i.d. log-likelihood ratio (LLR) sequence with constant bounds. In this case, it has been known [25, 44, 198] for a long time that these two functions satisfy the Fredholm integral equations of the second kind (FIESK) [18,203]. Analytical solutions are not available except for some special cases [241]. In general, one has to resort to some numerical techniques for the FIESK. Approximations were made in [244] by omitting the overshoot when the test statistic crosses one of the bounds. An iterative method was proposed in [199] to numerically solve the FIESK, and the convergence property of this method has been examined in [114]. Besides, extensive studies have been done to compute the OC and ASN for the truncated SPRT with some specific (e.g., Poisson and Binomial) processes [10, 11, 61, 67]. A method was proposed in [72] to convert the FIESK to a system of linear algebraic equations (SLAE) by approximating the integrals with a Gaussian quadrature. Although the SLAE method was proposed under the Gaussian assumption, actually it is generally applicable provided the Gaussian quadrature works well. Further, the fact that many probability densities can be well approximated by Gaussian mixtures with only a few components also broadens its

applicability. A problem with a discontinuous core of the FIESK was considered in [181], and the solution becomes more complicated. Other methods, such as finite element analysis [14, 16, 92], can also be employed to find the numerical solutions of the FIESK. Further, the bounds for the OC and ASN were calculated in [25, 73, 114, 244] (see the references therein). However, all these studies of OC and ASN are based on the i.i.d. assumption. The stationarity of an i.i.d. LLR sequence simplifies the analysis of OC and ASN greatly.

### 2.3.2 OC and ASN for Truncated SPRT

Unlike SPRT, the OC and ASN of the truncated SPRT (TSPRT) do not satisfy the FIESK no matter the LLR are i.i.d. or not. Methods have been proposed to compute the exact OC and ASN for TSPRT recursively and explicitly ("explicitly" means that they are not in the form of a solution to an equation), for example, by the so-called direct method proposed in [10–12]. Contrary to our method, this direct method actually does not compute the OC and ASN "directly." First, it computes recursively (from the initial time on) the probabilities of rejection, acceptance, and continuation at each time. Then, the OC and ASN are calculated based on the results of the first step. Convolutions over a finite range must be evaluated in order to yield analytical solutions. In most cases, the convolution becomes increasingly harder as the algorithm proceeds and numerical solutions are needed.

### 2.3.3 ARL for CUSUM

CUSUM's behavior is characterized by its ARL function, which is similar to ASN for SPRT. It can be computed based on the OC and ASN of the corresponding SPRT [25], which can be viewed as the building block of CUSUM for the i.i.d. LLR case. ARL also satisfies the FIESK [25, 44, 198] for this case. An ARL for exponentially distributed observations was provided in [241] by solving the FIESK equation. All the aforementioned numerical methods for SPRT are applicable to solve the ARL numerically. The upper and lower limits for the ARL were provided in [25, 115].

Multiple methods following another path have been proposed to approximate the OC and ASN for SPRT, and ARL for CUSUM. The idea is to approximate the discrete-time sequence by a continuous-time diffusion process. Then, the OC, ASN and ARL can be obtained by solving the so-called first-passage problem for the approximated diffusion processes [2,47,224] with different types of bounds (e.g., two absorbing bounds, or one absorbing bound and one reflecting bound). The probability of the first-passage time was computed and the OC and ASN can be inferred from it.

## 2.4 Brief Survey on Target Tracking Algorithms

### 2.4.1 Tracking without Measurement-Origin Uncertainty

If the correspondence between the measurements and targets are exactly known, it is a single target tracking problem without measurement origin uncertainty. Target maneuver, which may incur target motion-model uncertainty, and nonlinear filtering are two major problems in this case. A series of comprehensive surveys was provided for different aspects of single maneuvering target tracking: target dynamic model [150, 153], measurement model [148], maneuver detection [149], model uncertainty [152] and nonlinear filtering [151, 154–156].

### 2.4.2 Tracking in Clutter

The data association problem arises in multitarget tracking (MTT) or tracking in clutter due to measurement-origin uncertainty. State estimation under the measurement-origin uncertainty was pioneered by [192, 229]. However, what is involved in MTT is actually a joint decision and estimation problem [143,158]. Although it is not necessary to resolve data association explicitly before inferring the (multiple) target state, the most widely used strategy here is decide-then-estimate (i.e., data association followed by filtering).

### Validation Gate

For tracking in clutter, most often a gate for each track is formed to exclude extremely unlikely measurement-track association. A gate is a region in the measurement space that has a very high probability (called gating probability $P_G$) of capturing the true measurement. In most tracking systems, the gate takes simple shapes, such as rectangles, ellipses, and trapezoids, although more sophisticated gates have also been developed [21, 31, 33, 152, 221, 222, 228, 249].

### Nearest Neighbor Filter

Nearest neighbor (NN) association is one of the most widely used data association techniques due to its simplicity. The NN filter (NNF) uses the measurement closest to the predicted target state, as if it were the true measurement, to update the track by a conventional filter. NNF's performance was first examined in [221,226] and more thoroughly studied in [147,209]. NNF can be extended to multiple target tracking, leading to the global nearest neighbor (GNN) algorithm [31, 33, 119], which is equivalent to solving an assignment problem. The association probability of [64], generalized statistical distance of [31], and score gain of [33] are examples of the proposed assignment cost. Suboptimal algorithms, such as Hungarian algorithm [122], Munkres algorithm [186], JVC algorithm [99], auction algorithm [30, 38], have been proposed. Their performance was evaluated in [56, 108, 135]. See [35, 36, 45, 46, 175,187] for more information and more details for the assignment problem and its solutions.

### Strongest Neighbor Filter

If available, the signal intensity of a measurement can be used for data association. Like NNF, the strongest neighbor filter (SNF) uses the strongest measurement (in terms of echo intensity) to update the state estimate. SNF is widely used, especially for sonar tracking [132, 133]. Its performance was examined in [141, 146]. A probabilistic SNF was proposed in [162, 163, 233] with a significant performance improvement. SNF can also be extended

to multiple target tracking, resulting in the global strongest neighbor (GSN) algorithm. Nothing prevents us from using both the kinematic information and signal attribute of the measurement for data association if they are available. This combination certainly has the potential to beat both NNF and SNF.

### Probabilistic Data Association

The probabilistic data association filter (PDAF) was proposed in [23] for single target tracking in clutter. It makes soft decision on association and updates the track using the expected association. PDAF has been extended to MTT, resulting in the joint probabilistic data association filter (JPDAF) [21, 65]. PDAF is incapable of track initiation, confirmation, or termination. The "target observability" based PDAF (OPDAF) [40, 41] was the first piece of work to overcome this limitation. The IMM-PDAF [19, 21, 34, 88, 117] and the integrated PDAF (IPDAF) [188, 191] are more widely used. The IMM algorithm and IPDAF were combined in [82] to deal with track decision with a maneuvering target. IPDAF was also extended to multiple target tracking, resulting in the Joint IPDAF [189, 190]. A perceivability-based PDA tracker was also proposed in [138–140, 157].

### Multiple-Hypothesis Tracking

The multiple-hypothesis tracker (MHT) was pioneered by [208]. Its seed for single target tracking in clutter was planted in [227]. It is more powerful than previous methods and has a built-in mechanism for track management, but at the cost of being much more complicated. MHT does multi-scan association. It forms and propagates hypotheses, and decision on these hypotheses is postponed until enough data are collected. In general, the number of hypotheses increases exponentially as time goes. The combinatorial explosion must be overcome to make MHT practical. It involves two key steps: a) evaluation of each hypothesis; b) hypothesis management. See [32, 118, 185, 216] for more details and extensions of MHT. A good coverage of MHT can be found in [31, 33].

### Probabilistic Multiple-Hypothesis Tracking

The probabilistic MHT (PMHT) was first proposed in [236, 237]. It relaxes the assumption that a target cannot generate more than one measurements at every sensor scan, which has been widely used in other algorithms, and relies on the EM algorithm [51] to make a soft decision on data association. Although quite realistic and reasonable, the assumption renders the association of each measurement in every scan dependent and leads to a combinatorial optimization problem [247]. Without this assumption, the association of different measurements is relatively independent and hence simplifies the problem considerably. Applications and more details of PMHT can be found in [59, 68, 214, 247]. Extensions have been made to deal with maneuvering targets [171, 213], multiple sensors [49, 121, 183, 207], and nonlinear models [71]. An evaluation was provided in [91, 212]. [247] pointed out several design issues and problems of PMHT, along with proposed solutions.

### Data Association by Assignment

Data association can also be formulated as a multi-dimensional assignment problem [204], where the optimal association minimizes the assignment cost between measurements (in multi-scan) and tracks [33]. The optimal solution in general is infeasible for most practical applications. Other than the methods used in the GNN approach mentioned before, Lagrangian relaxation [50, 205, 206] is a promising and efficient suboptimal algorithm. See also [20] (chapter 2) for a good coverage of this topic.

### Symmetric Measurement Equation Method

As mentioned before, the data association problem need not to be solve explicitly for target tracking. Other than some density estimation based algorithms (e.g., PHD filter [173, 174]), the symmetric measurement equations (SME) method [26,112,113,136,137,215] addresses the data association problem implicitly by converting it to a highly nonlinear filtering problem, which is independent of the association. This conversion is done by a symmetric function $\varphi$

which must be invariant (symmetric) under the permutation of its arguments [215]. Possible symmetric functions $\varphi$, e.g., sum of power and sum of product, were proposed in [112,113,215] and their pros and cons were compared based on simulations [113, 137, 215].

# Chapter 3

# Generalized Linear Minimum Mean-Square Error Estimation

## 3.1 Introduction

In a Bayesian framework, estimation is to infer a random quantity $x$ based on prior information and measurement $z$. It is well known that the posterior mean $E[x|z]$ is the optimal estimator that minimizes mean-square error (MSE). This minimum mean-square error (MMSE) estimation in general requires knowledge of the entire distribution of $x$, which is not achievable for most applications other than the linear Gaussian case. However, it provides a theoretical basis for analysis and approximation techniques.

Instead of looking for the best estimator among all estimators, the best one in the set of all linear (in the measurement $z$) estimators may be a good choice, resulting in the linear minimum mean-square error (LMMSE) estimation. It makes a good compromise between technical simplicity and performance for problems with moderate nonlinearity. The well-known Kalman filter (KF) [110, 111] is a special case of the recursive LMMSE estimator, which is optimal (i.e., MMSE estimator) for a linear Gaussian system. LMMSE estimation plays a major role in nonlinear point estimation [151] and many popular nonlinear estimators,

for example, the first-order extended Kalman filter (EKF) [22, 96, 178, 179, 231, 248], the unscented filter (UF) [100–107, 251], the divided difference filter [196] and the Gaussian filter [94], are either approximation of or based on LMMSE estimation. Further, LMMSE estimation was successfully applied directly to radar tracking [58, 252] with linear target dynamics model and nonlinear measurement model. It was also applied to state estimation for Markovian jump linear systems [43, 76], which, however, should be better handled by multiple-model estimation [152].

Intuitively, LMMSE estimation should work well for problems with low degree of nonlinearity. However, if $x$ and $z$ are related highly nonlinearly, a linear estimator, even the best one, may not be adequate to provide acceptable accuracy. Instead of searching for the best in the set of all linear (in $z$) estimators, as LMMSE estimation does, we can look for the best estimator in a larger or different set, and hence improve estimation performance. We propose generalized LMMSE (GLMMSE) estimation by employing this idea. The candidate set in GLMMSE estimation can be determined by a vector-valued measurement transform function (MTF) $g(z)$, and we may search for the best estimator within the set $\mathcal{L}(g)$ of all estimators that are linear in $g(z)$, rather than in $z$. Clearly, this includes LMMSE estimation as a special case with $g$ being an identity function. Theoretically speaking, GLMMSE estimation with a proper MTF $g(z)$ performs at least as well as LMMSE estimation if the moments involved can be evaluated exactly. However, similar to LMMSE estimation, analytical evaluation of the moments needed is difficult to achieve in general. Fortunately, many approximation techniques, e.g., unscented transform and numerical integration, developed for LMMSE estimation are also applicable to GLMMSE estimation.

The idea of applying LMMSE estimation to converted measurements exists in the target tracking literature. In [58, 252], the radar measurements in polar coordinates were converted to Cartesian coordinates, resulting in the pseudo-linear measurements, and then LMMSE estimation was applied to the converted measurement. The converted-measurement Kalman filters proposed in [134, 182, 239] also followed this idea. However, they are all special cases

of our proposed GLMMSE. First, they all converted the measurements to be pseudo-linear in the target (partial) state, which may be difficult to achieve for some cases (e.g., the range rate). Besides, the pseudo-linear conversion may not always be beneficial. Further, most often the converted measurements have the same dimension as the original measurements, which is not a necessity in our GLMMSE estimation framework. Actually, we may argument the original measurements by the converted measurements in MTF $g(z)$ to enhance performance, provided they are not linearly dependent.

Clearly, "what $g(z)$ to use?" is the key to GLMMSE estimation. Design of $g(z)$ is evidently problem dependent and up to the preference of users. It is difficult to come up with rigorous rules for constructing MTF in general. However, design guidelines are provided and discussed based on a numerical example in this chapter. A general guideline can be made that $x$ should be less nonlinear in the converted data $y = g(z)$ than in $z$. A measure of nonlinearity [145] can be employed to quantify the degree of nonlinearity and thus provides criterion for designing function $g$. Also, an MTF can be selected from a family of functions by solving an optimization problem, i.e., minimizing the estimated mean-square errors computed in the estimator. Cautions are needed to consider the problems of the numerical instability and algorithm's sensitivity to the prior assumption.

Our method is illustrated by two nonlinear problems in target tracking: a) single target tracking with nonlinear dynamics model and measurement model; b) multi-target tracking (MTT) with a linear dynamics model and a nonlinear measurement model. Note that MTT is a nonlinear estimation problem due to the uncertainty of measurement origin, i.e., data association problem, no matter if the target dynamics model and measurement model are linear or not.

This chapter is organized as follows. LMMSE estimation is briefly reviewed in Sec. 3.2. GLMMSE estimation is proposed in Sec. 3.3 and the design guidelines for MTF are discussed in Sec. 3.4. Computation of GLMMSE estimator based on Gaussian density approximation and Gaussian-Hermite quadrature is given in Sec. 3.5. Our method is demonstrated by two

target tracking problems in Sec. 3.6 and 3.7, respectively. Its performance is compared with the conventional LMMSE estimation based on the results of Monte Carlo simulation. A summary is made in Sec. 3.8.

## 3.2 Review of LMMSE Estimation

Consider a parameter estimation problem, where the quantity to be estimated $x$ and measurement $z$ are related by a nonlinear function $\phi$

$$z = \phi(x) + v \tag{3.1}$$

with zero-mean additive noise $v$. The LMMSE estimation finds the best estimator in the set $\mathcal{L}(z)$ of all linear (more rigorously, affine) estimators (i.e., $\hat{x} = a + Bz$), that is,

$$\hat{x}^{\text{LMMSE}} \triangleq \arg \min_{\hat{x}=a+Bz} \text{MSE}(\hat{x}) \tag{3.2}$$

The coefficients $a$ and $B$ are determined such that the MSE is minimized. It turns out LMMSE estimation can be computed based on the first and second moments of $(x, z)$, that is,

$$\hat{x}^{\text{LMMSE}} = \bar{x} + C_{xz} C_z^{-1} (z - \bar{z}) \tag{3.3}$$

$$P = C_x - C_{xz} C_z^{-1} C_{xz}' \tag{3.4}$$

where

$$\bar{x} = E[x],\ \bar{z} = E[z]$$

are the prior means of $x$ and $z$, respectively. $C_{(\cdot)}$ is the covariance matrix of $(\cdot)$. $P$ is the (estimated) MSE matrix.

The estimator above is a batch algorithm, which is computationally inefficient for a dynamic problem. Recursive LMMSE estimation is desirable for filtering systems (2.1) and (2.2), which are repeated in the following for convenience

$$x_{k+1} = f_k(x_k) + w_k \tag{3.5}$$

$$z_k = h_k(x_k) + v_k \tag{3.6}$$

where, without loss of generality, we assume $w_k \sim (\mathbf{0}, Q_k)$ (mean and covariance) and $v_k \sim (\mathbf{0}, R_k)$. The corresponding recursive LMMSE estimation for the discrete-time system (3.5)–(3.6) is given by

$$\hat{x}_k^{\text{LMMSE}} = \hat{x}_{k|k-1} + K_k \tilde{z}_k$$

$$P_k = P_{k|k-1} - K_k C_{\tilde{z}_k} K_k'$$

where

$$K_k = C_{\tilde{x}_{k|k-1}\tilde{z}_k} C_{\tilde{z}_k}^{-1}, \quad Z^k = [z_1' \; z_2' \; \cdots \; z_k']'$$

$$\hat{z}_{k|k-1} = E[z_k | Z^{k-1}], \quad \hat{x}_{k|k-1} = E[x_k | Z^{k-1}]$$

$$\tilde{z}_k = z_k - \hat{z}_{k|k-1}, \quad \tilde{x}_{k|k-1} = x_k - \hat{x}_{k|k-1}$$

and $P_{k|k-1}$ and $P_k$ are the predicted and updated MSE matrices at time $k$.

## 3.3 Generalized LMMSE Estimation

For a highly nonlinear system, the performance of a linear estimator, even the best one, may not be good enough. We can enhance the performance of LMMSE estimation by enlarging or appropriately selecting the candidate set of estimators, rather than stick to the set of linear (in $z$) estimators. That is, we introduce a vector-valued measurement transform function

(MTF) $y = g(z)$, and find the best one within all estimators that are linear in the MTF, leading to the GLMMSE estimation:

$$\hat{x}^{\text{GLMMSE}} \triangleq \arg\min_{\hat{x}} \text{MSE}(\hat{x})$$

where

$$\hat{x} = a + By,\ y = g(z)$$

Clearly, it includes LMMSE estimation as a special case with $g$ being the identity function. The candidate set is determined by $g$, which is evidently problem dependent and up to the user's preference. Taking a scalar case as an example, $g$ may be chosen to include monomials of $z$ up to the third-degree:

$$g(z) = [z \quad z^2 \quad z^3]'$$

Then $\hat{x}$ is actually a third-degree polynomials and $\hat{x}^{\text{GLMMSE}}$ is the best one within the set of all polynomial functions of $z$ to the third-degree, which clearly includes the set of linear estimators as a subset. However, although increasing to a higher degree polynomial is theoretically promising, it should be avoided in general due to its numerical instability. Theoretically speaking, GLMMSE estimation with a proper MTF $g(z)$ should perform at least as well as LMMSE estimation if all the moments involved can be evaluated exactly. However, inaccurate approximation of these moments may result in worse performance.

For a parameter estimation problem (3.1), a similar derivation of LMMSE estimation yields

$$\hat{x}^{\text{GLMMSE}} = \bar{x} + C_{xy}C_y^{-1}(y - \bar{y}) \tag{3.7}$$

$$P = C_x - C_{xy}C_y^{-1}C_{xy}' \tag{3.8}$$

This is actually exactly the same formula of LMMSE estimation except that the first and second moments of $(x, y)$, rather than $(x, z)$, are used. Actually, it is simply applying the LMMSE estimation to the transformed measurement model. Hence, to avoid potential ambiguity, in this chapter, LMMSE estimation denotes specifically the linear estimation with original measurement model, and we refer all LMMSE estimation with transformed measurement model as GLMMSE estimation.

## 3.4 Design Guidelines

"What $g(z)$ to choose?" is clearly a central question for GLMMSE estimation. Unfortunately, rigorous rules to construct $g(z)$ that guarantees enhanced performance for the general case are difficult to come up and need further effort. In this section, some guidelines and observations based on a numerical example are provided to facilitate the design process for practical problems.

*Example:* Consider a scalar parameter estimation problem

$$z = \phi(x) + v = x^3 + v \tag{3.9}$$

where $v$ is zero-mean Gaussian white noise with variance $R$, and it is independent of $x$, which has a prior Gaussian density $\mathcal{N}(0, C_x)$. We define the signal-to-noise ratio (SNR) as

$$\text{SNR} \triangleq E[z_t' R^{-1} z_t]$$

which is the ratio between the power of the noise-free signal

$$z_t \triangleq \phi(x) = x^3$$

and the power of the noise. In this example, we have

$$\text{SNR} = \frac{15C_x^3}{R}$$

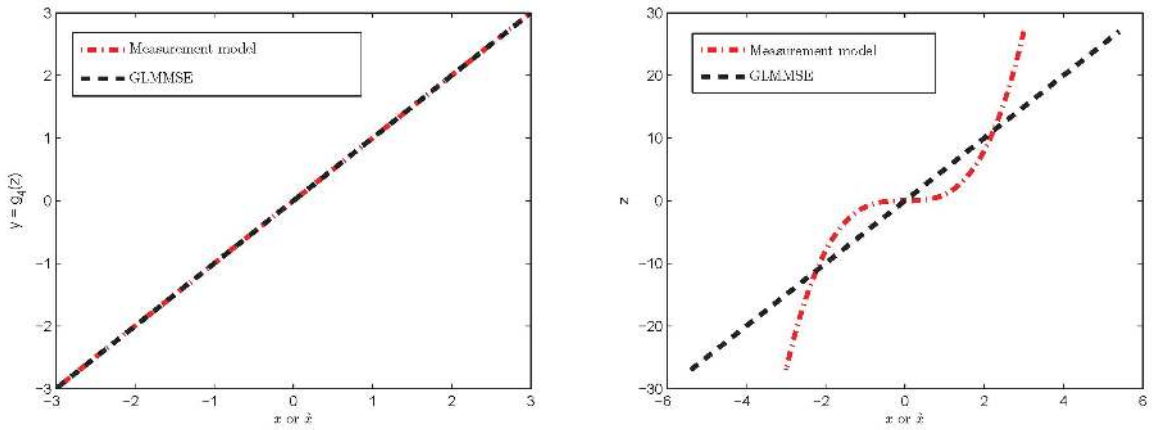
There are some options for applying GLMMSE estimation to this problem. GL($g_2$) (i.e., GLMMSE estimator with MTF $g_2$) and GL($g_4$) (see Table 3.2) are two widely used methods. Although $x$ appears linear in $g_4(z) = z^{1/3}$, the noise is no longer additive and is state dependent. Further, other MTF, e.g. $g(z) = z|z|^{r-1}$, may be also considered for this case, where $r$ is a real number. The root mean square errors (RMSE) of GL($g_i$), $i = 1, \cdots, 6$, with different prior variances $C_x$, are given in Table 3.2. Note that the GL($g_2$) is actually the LMMSE estimator. The RMSE of the MMSE estimation in the last row are estimated by density estimation and serve as performance lower bound for this example. The numbers in boldface are the lowest RMSE among $g_i,\ i = 1, \cdots, 6$.

When the SNR is small (i.e., from $C_x^1$ to $C_x^3$), the LMMSE estimation performs the best. In these cases, the degree of the nonlinearity of the measurement model does not have a significant impact on the performance due to the low SNR. The additive and state-independent noise in $g_2$ benefits the estimation performance. Further, the low SNR leads to insignificant performance differences among these estimators.

For the cases with high SNR (i.e., $C_x^8$ and $C_x^9$), GL($g_4$) outperforms the others. This can be intuitively understood that the degree of nonlinearity of the measurement model determines the performance in this case since a linear (in $g(z)$) estimator is used and the impact of the noise term is small. This becomes clearer when the noise is diminishing, which implies SNR$\to +\infty$, GL($g_4$) approaches the perfect estimate, that is,

$$\lim_{R\to 0} \hat{x}^{\text{GLMMSE}}(g_4) = \frac{E[xz^{1/3}]}{E[z^{1/3}z^{1/3}]} z^{1/3} = \frac{E[x^2]}{E[x^2]} x = x$$

However, none of the other GL($g_i$) in Table 3.2 achieve this, since $x$ is nonlinearly related to $g_i$, $i \neq 4$, and a linear estimator (in $g_i$) is used. For example, even with a perfect




(a) GL($g_4$) (b) GL($g_2$)

Figure 3.1: When $R \to 0$, estimation error is solely determined by the degree of nonlinearity of the measurement model. Although this is an extreme case, it reveals the situation with high SNR. However, for cases with low SNR, the resulting non-additive and state-dependent noise may make the estimator suffer.

measurement, GL($g_2$) does not equal to $x$ in general (see Fig. 3.1), that is,

$$\lim_{R\to 0} \hat{x}^{\text{GLMMSE}}(g_2) = \frac{E[xz]}{E[zz]} z = \frac{E[x^4]}{E[x^6]} z = \frac{x^3}{5C_x} \neq x$$
$$\lim_{R\to 0} \text{MSE} = C_x - C_{xz} C_z^{-1} C_{xz}' = \frac{2C_x}{5}$$

Hence, the estimation error in this case is purely due to the degree of nonlinearity of the (transformed) measurement model. So, $g(z)$ can be constructed by reducing the degree of nonlinearity of the measurement model for the high SNR case.

A measure of nonlinearity for estimation was proposed in [145], which can serve as a criterion for design. A normalized measure of nonlinearity for Eq. (3.1) was defined as

$$\mathcal{M} = \min_{L\in\mathcal{L}(x)} \sqrt{\frac{E[||L(x) - \phi(x)||_2^2]}{\text{tr}(C_\phi)}} \tag{3.10}$$

Table 3.1: measure of nonlinearity of $g_i$, with $C_x = 10,\ R = 1$.

| $g$ | $g_1$ | $g_2$ | $g_3$ | $g_4$ | $g_5$ | $g_6$ |
|---|---|---|---|---|---|---|
| Normalized MoN | 98.34% | 63.47% | 32.43% | 0 | 21.10% | 31.42% |

where $C_\phi$ is the covariance matrix of $\phi(x)$, and $\mathcal{L}(x)$ is the set of all linear functions

$$L(x) = a + Bx \tag{3.11}$$

Note that we are interested in the nonlinearity between $x$ and $z$, and hence additive noises are not considered in the measure. Intuitively, $\mathcal{M}$ quantifies the portion (or percentage) of the nonlinear part that cannot be accounted for by a linear function in $\phi$. As shown in [145], it has a standard range of $[0, 1]$. Clearly, $\{\mathcal{M} = 0\}$ implies that $\phi$ is linear almost everywhere, and $\{\mathcal{M} = 1\}$ implies that $\{\hat{L} = 0\}$, roughly meaning that $\phi$ contains no "linear component" at all. The computation of $\mathcal{M}$ requires knowledge of probability density function (PDF) $p(x)$ of $x$. If it is difficult to evaluate analytically due to the nonlinearity or the dynamics of the system, a sample representation of $p(x)$ can be obtained by various techniques, e.g., sequential Monte Carlo method, based on the prior distribution. For the measure of nonlinearity of the transformed model, simply replace the nonlinear function $\phi(x)$ by $g(\phi(x))$. Table 3.1 shows the measure of nonlinearity of the transformed measurement models $g_i,\ i = 1, \cdots, 6$. It appears that the conversion leading to the pseudo-linear measurements may be the first choice for the high SNR case. However, this may not be achievable in practice, or doing so may incur numerical problems, as explained later.

Further, it does not have to reduce the degree of nonlinearity to zero to benefit the performance, as demonstrated by the superior performance of GL($g_3$) in Table 3.2 for a certain range of SNR. In this case, $g_3(z)$ outperforms other MTF due to a good combination of the reduction of the degree of nonlinearity and the statistical property of the noise term in the converted measurement model. Hence, design of $g(z)$ depends on the SNR of the problem.

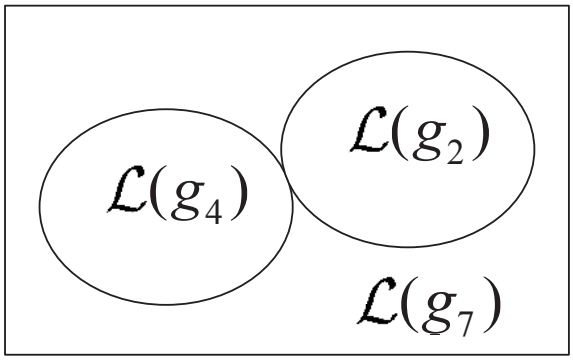



Figure 3.2: Candidate sets of estimators for searching.

Most often, conventional MTF applied in target tracking have the same dimension as the original measurements. They often correspond to coordinate transforms. For example, the converted-measurement Kalman filter converts the radar measurements in polar coordinates to Cartesian coordinates. Existing methods used either the original or the converted measurements for tracking. However, GLMMSE estimation can beat both methods by including both measurements in $g(z)$, which has a higher dimension than the original data. For example, GL($g_7$) beats both GL($g_2$) and GL($g_4$), as shown in Table 3.2. Actually, it is equivalent to augment the original measurement with the transformed measurement. Let $\mathcal{L}(g)$ denote the set of all estimators that are linear in $g$. GL($g_7$) finds the best estimator within $\mathcal{L}(g_7)$, which clearly includes the candidate sets $\mathcal{L}(z)$ in LMMSE estimation and $\mathcal{L}(g_4)$ in GL($g_4$) as subsets, and hence results in superior performance. See Fig. 3.2 for illustration. Other augmentations are optional. For example, $g(z) = [g_3 \;\; g_4]'$ or $g(z) = [g_2 \;\; g_3 \;\; g_4]'$ may be preferred to $g(z) = [g_2 \;\; g_4]'$ for the cases of $C_x^4$ to $C_x^7$ in Table 3.2. Theoretically speaking the more converted measurements are augmented (provided they are not linearly dependent), the better it should perform. This may not be true in practice. If the dimension of the augmented measurement (i.e., the dimension of $g(z)$) is much higher than that of $x$, computing the inversion of the second moment of $g(z)$ may become more difficult and numerically unstable.

The computed MSE $P$ (Eq. (3.8)) can also serve as an optimality criterion for comparing $g(z)$. An optimal $g(z)$ in a set $\mathcal{G}$ of candidate functions may be obtained by minimizing trace($P$), that is,

$$g(z) = \arg\min_{g \in \mathcal{G}} \text{trace}(P) \tag{3.12}$$

This requires solving a nonlinear (maybe constrained) optimization problem, which, for parameter estimation, can be done off-line (i.e., without measurements). Note that $P$ is the filter computed MSE, which would be accurate if the assumed prior information is precise and the numerical approximation involved is accurate. However, the actual filter performance depends on the credibility of these factors [161]. Consider the MTF

$$g(z) = \text{sign}(z)|z|^r, \ r \in [0.1, \ 1] \tag{3.13}$$

as an example. We want to find the optimal $r$ for Eq. (3.13) based on Eq. (3.12). Clearly, the optimal $r$ depends on the SNR of the problem. Several levels of SNR are considered in Table 3.3 by setting $C_x = 1$ and varying $R$. The optimal $r$ under each SNR and the corresponding RMSE are given in Table 3.3, where the RMSE of the MMSE estimation provides a performance lower bound. For the dynamic case, the relation between the optimal $r$ and SNR may be also calculated off-line, i.e, a function $r$(SNR) can be computed. At each filter step, the SNR can be estimated and corresponding $r$(SNR) can be used for MTF. Otherwise, at the cost of additional online computational demands, $r$ can be calculated online, as explained in the next section.

The estimator's sensitivity to the assumptions and stability to the numerical approximations should be seriously considered for designing $g(z)$. For example, measurement conversions that result in a high power of $z$, e.g., $g_1(z)$ in Table 3.2, should be avoided in general. GLMMSE estimation with such an MTF requires computing high-order moments of $z$ or $x$, and they are in general sensitive to the assumed prior information (e.g., distributions

of $x$ and $v$, which are hard to obtain precisely in practice) and the accuracy of numerical approximation. Further, they often cause numerical problems, e.g., ill-conditioned matrices, in the computation.

## 3.5 Numerical Approximation for GLMMSE Estimation

Similar to LMMSE estimation, the first and second moments of $(x,\ y)$ are required in the GLMMSE estimator (3.7)–(3.8) and they are difficult to evaluate analytically in general. Fortunately, the approximation techniques for LMMSE estimation can also be applied to GLMMSE estimation. We present our algorithm based on moment-matched Gaussian-density approximations and Gaussian-Hermite quadratures (GHQ), as does in the Gaussian filter [94].

For a scalar-valued $x$, the GHQ approximates an integral by a weighted sum:

$$\int_{-\infty}^{+\infty} f(x)e^{-x^2}dx \approx \sum_{i=1}^{n} \omega_i f(x_i)$$

where $n$ is the number of sample points $x_i$, which are the roots of the Hermite polynomial

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n e^{-x^2}}{dx^n}$$

and the corresponding weights are

$$\omega_i = \frac{2^{n-1} n! \sqrt{\pi}}{[nH_n x_i]^2}$$

The generalization of GHQ to the vector case with a general Gaussian density is

$$\begin{aligned}
&\int \cdots \int f(x)\mathcal{N}(x;\bar{x},C)dx_1 \cdots d_{x_{n_x}} \\
\overset{\text{shorthand}}{=} &\int f(x)\mathcal{N}(x;\bar{x},C)dx \\
\overset{x=C^{\frac{1}{2}}r+\bar{x}}{=} &\int f(C^{\frac{1}{2}}r+\bar{x})\mathcal{N}(r;0,I)dr \\
= &\int \frac{1}{\sqrt{(2\pi)^{n_x}}} f(C^{\frac{1}{2}}r+\bar{x})e^{-\frac{1}{2}rr'}dr \\
\overset{r=\sqrt{2}s}{=} &\int \frac{1}{\sqrt{\pi^{n_x}}} f((2C)^{\frac{1}{2}}s+\bar{x})e^{-ss'}ds \\
= &\int \cdots \int a(s)e^{-s_1^2} \cdots e^{-s_{n_x}^2} ds_1 \cdots d_{s_{n_x}}
\end{aligned}$$

where $a(s) = \frac{f((2C)^{\frac{1}{2}}s+\bar{x})}{\sqrt{\pi^{n_x}}}$ and $n_x$ is the dimension of $x$. Therefore, the GHQ can be applied to each dimension of $s$ separately.

For parameter estimation (3.1), let $\eta = [x' \ v']'$, and approximate the distribution of $\eta$ by a Gaussian density

$$\eta \sim \mathcal{N}(\eta;\bar{\eta},P_\eta)$$

where

$$\bar{\eta} = \begin{bmatrix} \bar{x} \\ \mathbf{0} \end{bmatrix}, \quad P_\eta = \begin{bmatrix} C_x & \mathbf{0} \\ \mathbf{0} & R \end{bmatrix}$$

Table 3.2: RMSE of GLMMSE estimation with different $g(z)$, $R = 1$.

| Function $g(z)$ | $C_x^{(1)} = 0.1$ | $C_x^{(2)} = 0.3$ | $C_x^{(3)} = 0.5$ | $C_x^{(4)} = 0.75$ | $C_x^{(5)} = 1$ | $C_x^{(6)} = 1.5$ | $C_x^{(7)} = 2$ | $C_x^{(8)} = 5$ | $C_x^{(9)} = 10$ |
|---|---|---|---|---|---|---|---|---|---|
| SNR | 0.015 | 0.405 | 1.875 | 6.33 | 15 | 50.625 | 120 | 1875 | 15000 |
| $g_1(z) = z^3$ | 0.3152 | 0.5369 | 0.6968 | 0.8543 | 0.9868 | 1.2087 | 1.3957 | 2.2069 | 3.1210 |
| $g_2(z) = z$ | **0.3148** | **0.4981** | **0.5517** | 0.6021 | 0.6614 | 0.7858 | 0.9000 | 1.4148 | 2.0001 |
| $g_3(z) = z^{\frac{3}{5}}$ | 0.3150 | 0.5083 | 0.5731 | **0.5884** | **0.5847** | **0.5822** | **0.5946** | 0.7660 | 1.0415 |
| $g_4(z) = z^{\frac{1}{3}}$ | 0.3152 | 0.5174 | 0.6058 | 0.6558 | 0.6715 | 0.6711 | 0.6553 | **0.5607** | **0.4836** |
| $g_5(z) = z^{\frac{1}{5}}$ | 0.3153 | 0.5225 | 0.6223 | 0.6990 | 0.7403 | 0.7887 | 0.8136 | 0.8804 | 0.9828 |
| $g_6(z) = z^{\frac{1}{7}}$ | 0.3154 | 0.5261 | 0.6306 | 0.7178 | 0.7704 | 0.8430 | 0.8903 | 1.0599 | 1.2791 |
| $g_7(z) = [g_2 \;\; g_4]'$ | 0.3148 | 0.4944 | 0.5516 | 0.5879 | 0.6083 | 0.6186 | 0.6145 | 0.5478 | 0.4818 |
| RMSE of MMSE | 0.3137 | 0.4645 | 0.5511 | 0.5692 | 0.5598 | 0.5535 | 0.5205 | 0.4474 | 0.3645 |

Table 3.3: Optimal $r$, with $C_x = 1$.

| $R$ | 0.001 | 0.01 | 0.1 | 0.25 | 0.5 | 0.75 | 1 | 2.5 | 5 | 7.5 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| SNR | 15000 | 1500 | 150 | 60 | 30 | 20 | 15 | 6 | 3 | 2 | 1.5 |
| Optimal $r$ | 0.3670 | 0.4064 | 0.4895 | 0.5426 | 0.5947 | 0.6302 | 0.6575 | 0.7673 | 0.8750 | 0.9508 | 1.0000 |
| RMSE | 0.1449 | 0.2391 | 0.3849 | 0.4577 | 0.5181 | 0.5550 | 0.5816 | 0.6695 | 0.7360 | 0.7739 | 0.8000 |
| RMSE of MMSE | 0.1207 | 0.2082 | 0.3481 | 0.4302 | 0.5002 | 0.5305 | 0.5598 | 0.6570 | 0.7342 | 0.7725 | 0.7992 |

and $\bar{x}$ and $C_x$ are the assumed prior mean and covariance matrix of $x$, respectively. Then, the moments involved in Eqs. (3.7) and (3.8) are computed by

$$\begin{aligned}
\bar{y} &= \int y(\eta)\mathcal{N}\left(\eta; \bar{\eta}, P_\eta\right) d\eta = \int g(\phi(x) + v)\mathcal{N}\left(\eta; \bar{\eta}, P_\eta\right) d\eta \\
C_y &= \int (y(\eta) - \bar{y})(y(\eta) - \bar{y})'\mathcal{N}\left(\eta; \bar{\eta}, P_\eta\right) d\eta \\
C_{xy} &= \int (x - \bar{x})(y(\eta) - \bar{y})'\mathcal{N}\left(\eta; \bar{\eta}, P_\eta\right) d\eta
\end{aligned}$$

where $\phi(x)$ is given in Eq. (3.1), and all the integrals are evaluated by GHQ.

The recursive GLMMSE estimation for the dynamic system (3.5)–(3.6) can be obtained similarly

$$\begin{aligned}
\hat{x}_k^{\text{GLMMSE}} &= \hat{x}_{k|k-1} + K_k \tilde{y}_k \\
P_k &= P_{k|k-1} - K_k C_{\tilde{y}_k} K_k'
\end{aligned}$$

where

$$\begin{aligned}
K_k &= C_{\tilde{x}_{k|k-1}\tilde{y}_k} C_{\tilde{y}_k}^{-1} \\
Y^k &= [y_1' \; y_2' \; \cdots \; y_k']' \\
y_k &= g(z_k) \\
\hat{y}_{k|k-1} &= E[y_k | Y^{k-1}] \\
\tilde{y}_k &= y_k - \hat{y}_{k|k-1}
\end{aligned}$$

Following the same idea of the Gaussian filter [7, 94], which successively approximates the probability densities needed by moment-matched Gaussian densities and evaluates integrals by GHQ, we have the recursive GLMMSE estimation. One cycle of it is given below:

1. State prediction

Approximate the posterior density of $x_{k-1}$ by a moment-matched Gaussian density:

$$\text{PDF}(x_{k-1}|Y^{k-1}) \approx \mathcal{N}(x_{k-1}; \hat{x}_{k-1}, P_{k-1})$$

where $\text{PDF}(\cdot)$ stands for probability density function of $(\cdot)$. Then

$$\begin{aligned}
\hat{x}_{k|k-1} &\approx \int f_{k-1}(x)\mathcal{N}(x; \hat{x}_{k-1}, P_{k-1})dx \\
P_{k|k-1} &\approx \int (f_{k-1}(x) - \hat{x}_{k|k-1})(f_{k-1}(x) - \hat{x}_{k|k-1})'\mathcal{N}(x; \hat{x}_{k-1}, P_{k-1})dx \\
&\quad + Q_{k-1}
\end{aligned}$$

2. Measurement prediction

   Let $\eta_k = [x_k' \; v_k']'$. Approximate the probability density by a moment-matched Gaussian density:

$$\text{PDF}(\eta_k|Z^{k-1}) \approx \mathcal{N}(\eta_k; \bar{\eta}_k, P_k^\eta)$$

   where

$$\bar{\eta}_k = \begin{bmatrix} \hat{x}_{k|k-1} \\ \mathbf{0} \end{bmatrix}, \quad P_k^\eta = \begin{bmatrix} P_{k|k-1} & \mathbf{0} \\ \mathbf{0} & R_k \end{bmatrix}$$

   Let $y_k(\eta_k) = g(h_k(x_k) + v_k)$. Then

$$\hat{y}_{k|k-1} \approx \int_{-\infty}^{+\infty} y_k(\eta_k)\mathcal{N}(\eta_k; \bar{\eta}_k, P_k^\eta)d\eta_k$$

3. Filter gain

$$K_k = C_{xy}S_k^{-1} \tag{3.14}$$

where

$$S_k \approx \int (y_k(\eta_k) - \hat{y}_{k|k-1})(y_k(\eta_k) - \hat{y}_{k|k-1})' \mathcal{N}(\eta_k; \bar{\eta}_k, P_k^{\eta}) d\eta_k$$
$$C_{xy} \approx \int (x - \hat{x}_{k|k-1})(y_k(\eta_k) - \hat{y}_{k|k-1})' \mathcal{N}(\eta_k; \bar{\eta}_k, P_k^{\eta}) d\eta_k$$

All the integrals above are evaluated by the GHQ.

4. Update

$$\hat{x} = \hat{x}_{k|k-1} + K_k[g(z_k) - \hat{y}_{k|k-1}]$$
$$P_k = P_{k|k-1} - K_k S_k K_k'$$

As mentioned in the previous section, trace($P_k$) may serve as an optimization criterion for selecting an MTF $g(z_k)$. For example, if Eq. (3.13) is chosen as the MTF, then $P_k$ is a function of $r$. An optimal or suboptimal $r$ for each time step $k$ may be found adaptively by a numerical optimization procedure, which may iterate steps (2-4) a few times. Although searching for the optimal solution is usually computationally demanding, we stress that it does not have to find the best $r$ to benefit the performance.

Instead of a single Gaussian density, the PDF can be approximated by a Gaussian mixture to promote approximation accuracy, similarly as the Gaussian sum filter or mixed Gaussian filter does [94]. GHQ is chosen due to its superior performance to the unscented transform in our simulation. However, other approximation techniques may be preferred for a specific problem. It is well known that GHQ suffers from the "curse of dimensionality" and is computationally inefficient for high dimensional $x$ or $\eta$. Many methods are available to mitigate, to some extent, this effect (see [4, 6, 8, 42, 70, 79, 164, 184, 194, 200, 230, 232, 238]).

## 3.6 Application to Radar Tracking

We first illustrate our method by a 2 two-dimensional tracking problem with radar measurements. The performance of LMMSE estimation, GLMMSE estimation with pseudo-linear measurement and GLMMSE estimation with augmented measurement is compared. As mentioned before, the first two methods are widely used in tracking. As demonstrated by the simulation results, we can further improve the performance by using both the original measurement and the converted measurement.

### 3.6.1 Simulation Scenario

Assume a target flies in a (nearly) constant-turn (CT) motion with an unknown turn rate $\omega_k$. Let the target state be

$$x_k = [\mathrm{x}\ \dot{\mathrm{x}}\ \mathrm{y}\ \dot{\mathrm{y}}\ \omega]_k'$$

where subscript $k$ denotes the time index, $(\mathrm{x}, \mathrm{y})$ are Cartesian coordinates of target position, and $(\dot{\mathrm{x}}, \dot{\mathrm{y}})$ are the velocities along x and y directions, respectively. Then the target dynamics can be described by [150]

$$x_k = \begin{bmatrix} \mathrm{x} + \frac{\sin\omega T}{\omega}\dot{\mathrm{x}} - \frac{1-\cos\omega T}{\omega}\dot{\mathrm{y}} \\ (\cos\omega T)\dot{\mathrm{x}} - (\sin\omega T)\dot{\mathrm{y}} \\ \mathrm{y} + \frac{1-\cos\omega T}{\omega}\dot{\mathrm{x}} + \frac{\sin\omega T}{\omega}\dot{\mathrm{y}} \\ (\sin\omega T)\dot{\mathrm{x}} + (\cos\omega T)\dot{\mathrm{y}} \\ \omega \end{bmatrix}_{k-1} + w_k \tag{3.15}$$

where $T$ is the sampling interval. Due to the unknown turn rate $\omega_k$, the target dynamics (3.15) model is nonlinear and $\omega_k$ is estimated online together with the target position and velocity. Assume the range $r_k$ and the bearing $\theta_k$ of the target are available from each radar

scan at time $t = kT$:

$$z_k = \begin{bmatrix} r \\ \theta \end{bmatrix}_k = \begin{bmatrix} \sqrt{\mathrm{x}^2 + \mathrm{y}^2} \\ \arctan \frac{\mathrm{y}}{\mathrm{x}} \end{bmatrix}_k + v_k \tag{3.16}$$

The additive noises $w_k$ and $v_k$ in Eqs. (3.15)–(3.16) are assumed zero-mean white Gaussian noises with covariance $Q_k$ and $R_k$, respectively. We consider two MTF for GLMMSE estimation

$$g_1(z_k) = \begin{bmatrix} r\cos\theta \\ r\sin\theta \end{bmatrix}_k$$

and

$$g_2(z_k) = \begin{bmatrix} z \\ r\cos\theta \\ r\sin\theta \end{bmatrix}_k$$

Clearly, $g_1$ results in pseudo-linear measurements and $g_2$ arguments the pseudo-linear measurements into the original measurements.

### 3.6.2 Ground Truth

The true initial state of the target is set to be

$$x_0 = [50000m\ 200m/s\ -50000m\ 200m/s\ 0.01rad/s]'$$

and it follows the CT model (3.15). The initial state and MSE matrix for each filter are chosen to be

$$\hat{x}_0 \sim N(x_0, P_0)$$
$$P_0 = \text{diag}[10^8 \ 10^2 \ 10^8 \ 10^2 \ 0.001]'$$

The sampling interval is $T = 10s$ and totally $k = 50$ steps are simulated.

### 3.6.3 Performance Measure

The following performance measures are computed based on 1000 Monte Carlo (MC) runs. Each measure reveals different aspects of an estimator's performance.

1. Root mean-square error (RMSE):

$$\text{RMSE}_k = \sqrt{\frac{1}{I}\sum_{i=1}^{I}(\tilde{x}_k^{[i]})'\tilde{x}_k^{[i]}}$$

   where $\tilde{x}_k^{[i]} = x_k^{[i]} - \hat{x}_k^{[i]}$ is the estimation error at time $k$ on the $i$th MC run, and $I$ is the total number of MC runs. This measure evaluates an estimator's (average) accuracy.

2. Filter credibility:

   Other than estimation accuracy, we also evaluate the credibility of each estimator—how close the estimator's self-assessments are to the true performance—by evaluating the accuracy of the estimated MSE. The non-credibility index (NCI)

$$\gamma_k = \frac{10}{I}\sum_{i}^{I} |\log_{10}(\rho_k^i)|$$

   and inclination index (I.I.)

$$\xi_k = \frac{10}{I}\sum_{i}^{I} \log_{10}(\rho_k^i)$$

were proposed for such a purpose in [159, 161], where

$$\rho_k^{[i]} = \frac{(\tilde{x}_k^{[i]})'(\hat{P}_k^{[i]})^{-1}\tilde{x}_k^{[i]}}{(\tilde{x}_k^{[i]})'(\hat{P})_k^{-1}\tilde{x}_k^{[i]}}$$
$$\hat{P}_k = \frac{1}{I}\sum_{i=1}^{I}\tilde{x}_k^{[i]}(\tilde{x}_k^{[i]})'$$

and $[i]$ is the index of the MC run, $\hat{P}_k^{[i]}$ is the filter computed MSE matrix on the $i$th run, $\hat{P}_k$ can be understood as the (approximated) real MSE based on simulated estimation errors, and $\rho_k$ is the ratio of squared estimation errors normalized by computed and real MSE matrices, respectively. Hence, an estimator is more credible if the NCI is smaller, meaning that the computed MSE is closer to the true MSE. The inclination (pessimistic or optimistic) of an estimator is indicated by I.I.: if I.I. is significantly larger than zero and $\gamma_k \approx \xi_k$, it implies that the estimator tends to be optimistic, meaning that the computed MSE is smaller than the truth; if I.I. is significantly smaller than zero and $\gamma_k \approx -\xi_k$, then the estimator tends to be pessimistic since the computed MSE is larger than the truth in this case.

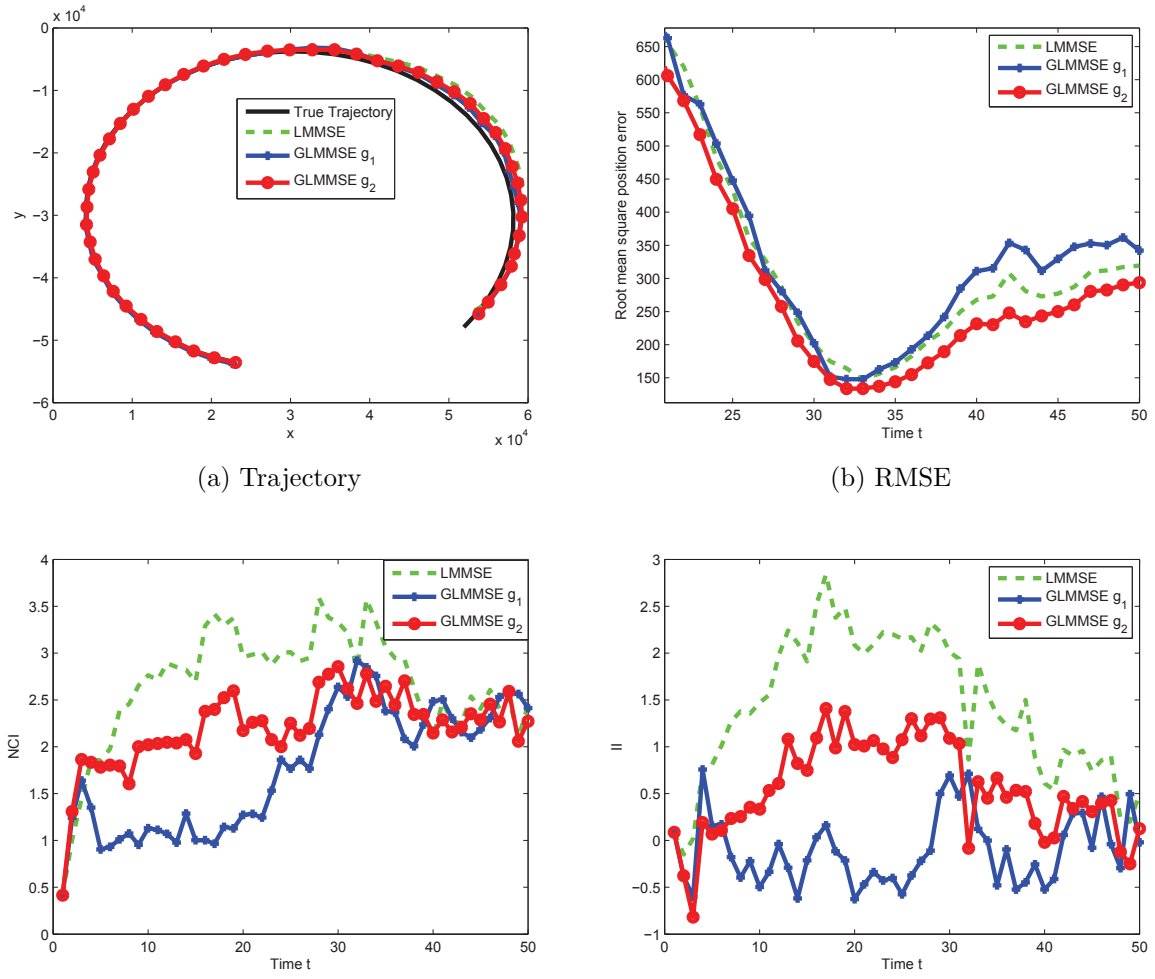
### 3.6.4 Simulation Results

Two cases of different levels of measurement noises are considered. Three estimators—LMMSE estimator, GL($g_1$) and GL($g_2$)—were simulated and the results of these two cases are given in Figs. 3.3 and 3.4, respectively.

Case 1: We consider a moderate level of measurement noise,

$$R_k = \begin{bmatrix} 1 & 0 \\ 0 & 0.001 \end{bmatrix}$$

The true and estimated trajectories are given in Fig. 3.3a. The RMSE of the filters are given in Fig. 3.3b. GL($g_2$) outperforms the other two filters in terms of estimation accuracy,




(a) Trajectory (b) RMSE

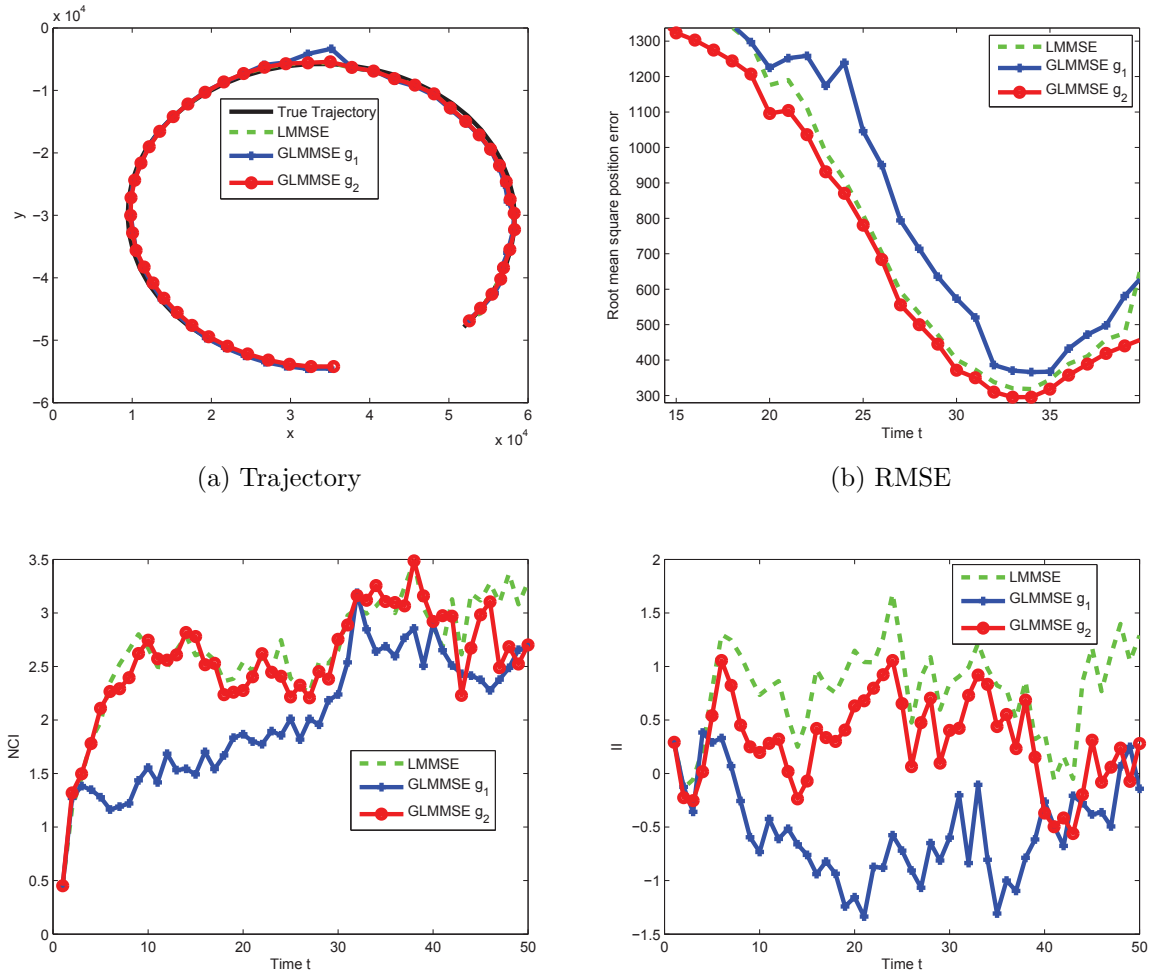
(c) NCI (d) I.I.

Figure 3.3: Radar tracking (Case 1).

and the LMMSE filter is slightly better than GL($g_1$). The NCI and I.I. are shown in Figs. 3.3c and 3.3d, respectively. As indicated by NCI, the GL($g_1$) is significantly less non-credible than the other two filters in the beginning stage (from start to about $k = 25$) and then their performance difference becomes insignificant. Based on the I.I., LMMSE filter is somewhat optimistic, GL($g_2$) tends to be slightly optimistic and GL($g_1$) shows no persistent inclination.

Case 2: The covariance of the measurement noise is increased to

$$R_k = \begin{bmatrix} 10 & 0 \\ 0 & 0.005 \end{bmatrix}$$




(a) Trajectory

(b) RMSE

(c) NCI

(d) I.I.

Figure 3.4: Radar tracking (Case 2).

The trajectories and estimation errors are given in Figs. 3.4a and 3.4b, respectively. The estimation errors of all three estimators become larger than Case 1, as expected. Although GL($g_2$) still outperforms the other filters, the performance gap between GL($g_2$) and LMMSE filter becomes insignificant. Similar observations (as in Case 1) on NCI, shown in Fig. 3.4c, can be made here. However, unlike Case 1, GL($g_1$) tends to be slightly pessimistic in this case, as shown by I.I. in Fig. 3.4d.

## 3.7 Application to Multitarget Tracking

In this section, we demonstrate our method by an example of multitarget tracking (MTT). In general,a major difficulty in MTT arises from the uncertainty of measurement origin, that is, the data association problem. MTT algorithms abound in the tracking literature and they all have pros and cons. We proceed based on the symmetric measurement equations (SME) method [26, 112, 113, 136, 137, 215], which avoids the data association problem by converting it to a highly nonlinear filtering problem. Since we focus on the performance of nonlinear filtering, a simplified MTT scenario is considered here. Assume the number of targets in the surveillance region is known, and the sensor has a perfect detection rate and no false alarm. That is, each target generates one and only one measurement at each sensor scan. However, the SME method can be generalized to deal with clutter and miss detection. Let

$$X_k = [(x_k^1)',\ \cdots,\ (x_k^n)']'$$

be the multitarget state vector obtained by stacking together the state vectors of individual targets $x_k^i$, and $\hat{X}_k$ be its estimate. The state vector of each individual target is defined as

$$x_k^i = [\mathrm{x}^i\ \dot{\mathrm{x}}^i\ \mathrm{y}^i\ \dot{\mathrm{y}}^i]_k',\ i = 1, \cdots, n$$

in the 2D Cartesian coordinate system, where $k$ is the time step and $n$ is the total number of targets, which is assumed known and constant. Assume all the targets move in the (nearly) constant velocity (CV) model [150]

$$X_k = \mathbf{F}_k X_{k-1} + W_k \tag{3.17}$$

where

$$\mathbf{F}_k = \text{diag}(F, \cdots, F)$$

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

$$W_k = [(w_k^1)', \ \cdots, \ (w_k^n)']'$$

and $T$ is the sampling interval, $w_k^i$ is the process noise of target $i$. Based on the assumptions of perfect sensor detection and no false measurement, each target will generates one and only one measurement at each sensor scan. Let

$$Z_k = \{z_k^1, \ \cdots, \ z_k^n\}$$

be the measurement set at time $k$, which contains $n$ measurements from the $n$ targets. We assume the range and azimuth are measured

$$z_k^j \triangleq \begin{bmatrix} \mathrm{z}_\mathrm{x}^j \\ \mathrm{z}_\mathrm{y}^j \end{bmatrix}_k = H_k(x_k^i) + v_k^i \tag{3.18}$$

if $z_k^j$ is from target $x_k^i$, where

$$H_k(x_k^i) = \begin{bmatrix} \sqrt{(\mathrm{x}^i)^2 + (\mathrm{y}^i)^2} \\ \arctan(\mathrm{y}^i/\mathrm{x}^i) \end{bmatrix}_k$$

and $v_k^i$ is the measurement noise. As mentioned before, even if the target dynamics model (3.17) and measurement model (3.18) were all linear, MTT is a non-linear estimation problem due to the unknown correspondence between the measurements and targets. The SME

method avoids data association by system conversion, results in association-independent measurement model. This conversion is done by a symmetric function $\varphi$. Then a nonlinear filter is needed to tackle the converted nonlinear system. Note that nonlinear filters based on density estimation, e.g., particle filter, may not be computationally feasible because of the high-dimensionality of this system.

The symmetric function $\varphi$ in SME must be a vector-valued function of measurement set $Z_k$, that is, the function value must be invariant (symmetric) under the permutation of its arguments:

$$\varphi(z_k^1, \cdots, z_k^n) = \varphi(z_k^{\pi_1}, \cdots, z_k^{\pi_n})$$

where $\{\pi_1, \cdots, \pi_n\}$ could be any permutation of the index $\{1, \cdots, n\}$. Further, some regularity conditions [215] on $\varphi$ should be satisfied. Some symmetric functions, e.g., sum of power and sum of product, were proposed in [112, 113, 215] and their pros and cons were compared based on simulations [113, 137, 215].

### 3.7.1 Simulation Scenario

We consider $n = 3$ targets in a CV motion in the surveillance region. Their initial states are

$$x_0^1 = [5000m \ -10m/s \ 1000m \ 10m/s]'$$
$$x_0^2 = [4400m \ 10m/s \ 1000m \ 10m/s]'$$
$$x_0^3 = [4700m \ -2m/s \ 1000m \ 10m/s]'$$

The sensor sampling interval is $T = 1s$ and totally $k = 50$ steps were simulated. Two symmetric functions $\varphi$ can be used:

Sum of power:

$$\varphi_e(Z_k) = \begin{bmatrix} \mathrm{z}_{\mathrm{x}}^1 + \mathrm{z}_{\mathrm{x}}^2 + \mathrm{z}_{\mathrm{x}}^3 \\ \mathrm{z}_{\mathrm{y}}^1 + \mathrm{z}_{\mathrm{y}}^2 + \mathrm{z}_{\mathrm{y}}^3 \\ (\mathrm{z}_{\mathrm{x}}^1)^2 + (\mathrm{z}_{\mathrm{x}}^2)^2 + (\mathrm{z}_{\mathrm{x}}^3)^2 \\ (\mathrm{z}_{\mathrm{y}}^1)^2 + (\mathrm{z}_{\mathrm{y}}^2)^2 + (\mathrm{z}_{\mathrm{y}}^3)^2 \\ (\mathrm{z}_{\mathrm{x}}^1)^3 + (\mathrm{z}_{\mathrm{x}}^2)^3 + (\mathrm{z}_{\mathrm{x}}^3)^3 \\ (\mathrm{z}_{\mathrm{y}}^1)^3 + (\mathrm{z}_{\mathrm{y}}^2)^3 + (\mathrm{z}_{\mathrm{y}}^3)^3 \end{bmatrix}_k \tag{3.19}$$

Sum of product:

$$\varphi_\pi(Z_k) = \begin{bmatrix} \mathrm{z}_{\mathrm{x}}^1 + \mathrm{z}_{\mathrm{x}}^2 + \mathrm{z}_{\mathrm{x}}^3 \\ \mathrm{z}_{\mathrm{y}}^1 + \mathrm{z}_{\mathrm{y}}^2 + \mathrm{z}_{\mathrm{y}}^3 \\ \mathrm{z}_{\mathrm{x}}^1\mathrm{z}_{\mathrm{x}}^2 + \mathrm{z}_{\mathrm{x}}^1\mathrm{z}_{\mathrm{x}}^3 + \mathrm{z}_{\mathrm{x}}^2\mathrm{z}_{\mathrm{x}}^3 \\ \mathrm{z}_{\mathrm{y}}^1\mathrm{z}_{\mathrm{y}}^2 + \mathrm{z}_{\mathrm{y}}^1\mathrm{z}_{\mathrm{y}}^3 + \mathrm{z}_{\mathrm{y}}^2\mathrm{z}_{\mathrm{y}}^3 \\ \mathrm{z}_{\mathrm{x}}^1\mathrm{z}_{\mathrm{x}}^2\mathrm{z}_{\mathrm{x}}^3 \\ \mathrm{z}_{\mathrm{y}}^1\mathrm{z}_{\mathrm{y}}^2\mathrm{z}_{\mathrm{y}}^3 \end{bmatrix}_k \tag{3.20}$$

Hence the converted measurement

$$Z_k^s = \varphi(Z_k) \triangleq \begin{bmatrix} \varphi^1 \\ \varphi^2 \\ \varphi^3 \\ \varphi^4 \\ \varphi^5 \\ \varphi^6 \end{bmatrix}_k = \varphi(\mathbf{H}_k(X_k) + V_k) \tag{3.21}$$

is invariant with respect to the association between the measurements and targets, where $\varphi^i$ is the $i$th element in vector $Z_k^s$ and

$$\mathbf{H}_k(X_k) = [H_k(x_k^1)\ H_k(x_k^2)\ H_k(x_k^3)]'$$
$$V_k = [(v_k^1)',\ (v_k^2)',\ (v_k^3)']'$$

So, after the symmetric conversion, MTT can be addressed without the data association. However, Eq. (3.21) becomes highly nonlinear. We choose $\varphi = \varphi_\pi$ since it outperforms $\varphi_e$ in our scenario, agreeing with the observations in [113].

Directly applying LMMSE estimation to system (3.17) and (3.21) tends to suffer from numerical problems. The condition number of $S_k$ in Eq. (3.14) is larger than $10^{20}$ and hence results in vulnerable performance. Further, converting $Z_k^s$ to make it pseudo-linear in $X_k$ is difficult, if not impossible. However, with the flexibility introduced by MTF in GLMMSE estimation, many options are at hand to reduce the degree of nonlinearity. We proceed by choosing the following MTF:

$$g_1(Z_k^s) = \begin{bmatrix} \varphi^1 \\ \varphi^2 \\ \mathrm{sign}(\varphi^3)\sqrt{|\varphi^3|} \\ \mathrm{sign}(\varphi^4)\sqrt{|\varphi^4|} \\ \sqrt[3]{\varphi^5} \\ \sqrt[3]{\varphi^6} \end{bmatrix}_k, \quad g_2(Z_k^s) = \begin{bmatrix} \varphi^1 \\ \varphi^2 \\ \sqrt[3]{\varphi^3} \\ \sqrt[3]{\varphi^4} \\ \sqrt[5]{\varphi^5} \\ \sqrt[5]{\varphi^6} \end{bmatrix}_k$$

and

$$g_{\text{adapt}}(Z_k^s) = \begin{bmatrix} \varphi^1 \\ \varphi^2 \\ \text{sign}(\varphi^3)|\varphi^3|^{\frac{1}{r}} \\ \text{sign}(\varphi^4)|\varphi^4|^{\frac{1}{r}} \\ \text{sign}(\varphi^5)|\varphi^5|^{\frac{2}{3r}} \\ \text{sign}(\varphi^6)|\varphi^6|^{\frac{2}{3r}} \end{bmatrix}_k , \; r_l \leq r_k \leq r_u \tag{3.22}$$

Note that $g_{\text{adapt}}$ is a function parameterized by $r_k$, which is computed adaptively by minimizing the trace($P_k$) online. Further, $r_k$ is confined within the interval $[r_l \quad r_u]$ to avoid numerical problems. If we set $r_k = 2$, then $g_{\text{adapt}} = g_1$.

We demonstrate the performance of GL($g_1$), GL($g_2$) and GL($g_{\text{adapt}}$) by comparing them with the performance lower bound provided by the unscented filter (UF) with exact association. All the filters were initialized by

$$\hat{X}_0 \sim N(X_0, P_0), \quad P_0 = \text{diag}(p_0, p_0, p_0)$$

where

$$X_0 = [(x_0^1)', (x_0^2)', (x_0^3)']', \quad p_0 = \text{diag}(1000\ 0.01\ 1000\ 0.01)$$

### 3.7.2 Performance Measure

All the filters are evaluated by the optimal subpattern assignment (OSPA) metric [220] proposed for evaluating MTT algorithms. Since the number of targets is known in our scenario, OSPA can be simplified to

$$D(X_k, \hat{X}_k) = \min_{\pi \in \Pi_n} \sum_{i=1}^{n} d_o(x_k^i, \hat{x}_k^{\pi(i)})$$

where $\Pi_n$ is the set of all possible permutations of the index $\{1, \cdots, n\}$, and we choose the distance function

$$d_o(x_k^i, \hat{x}_k^j) \triangleq ||x_k^i - \hat{x}_k^j||_2$$

due to its popularity. The measure is evaluated based on 100 MC simulation runs.

### 3.7.3 Simulation Results

Two cases of different levels of measurement noise were simulated and the results are given in Figs 3.5 and 3.6, respectively.

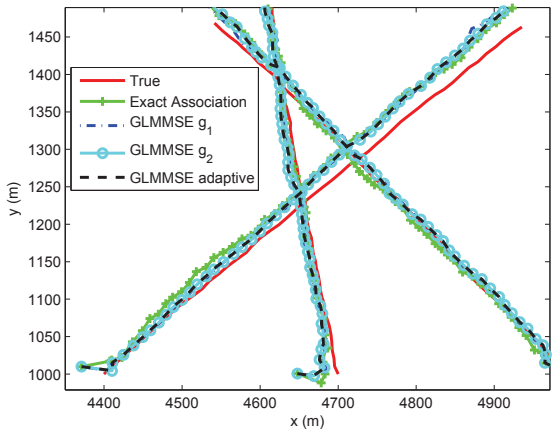
Case 1: The covariance of the measurement noise is set to

$$R_k = \begin{bmatrix} 100 & 0 \\ 0 & 0.0001 \end{bmatrix}$$

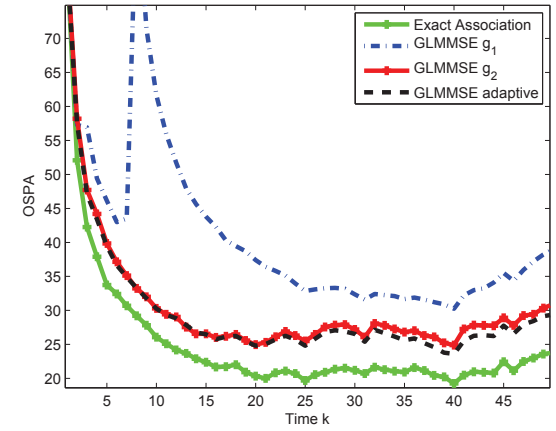
$r_l = 2.5$ and $r_u = 3.5$ are chosen for $g_{\text{adapt}}$. One realization of the true and estimated trajectories of the three targets are given in Fig. 3.5a. The OSPA of each filters are given in Fig. 3.5b. The UF with exact association provides the performance lower bound. GL($g_2$) and GL($g_{\text{adapt}}$) perform similarly and they outperform GL($g_1$) greatly. The average $r_k$ for GL($g_{\text{adapt}}$) is shown in Fig. 3.5c. The ratio of computational requirements of GL($g_2$) and GL($g_{\text{adapt}}$) is 1/5.5. In this case, the performance gain to GL($g_2$) achieved by the optimization procedure in GL($g_{\text{adapt}}$) is insignificant, meaning that GL($g_2$) may be preferred due to its good combination of accuracy and computation complexity.

Case 2: The covariance of measurement noise is increased to

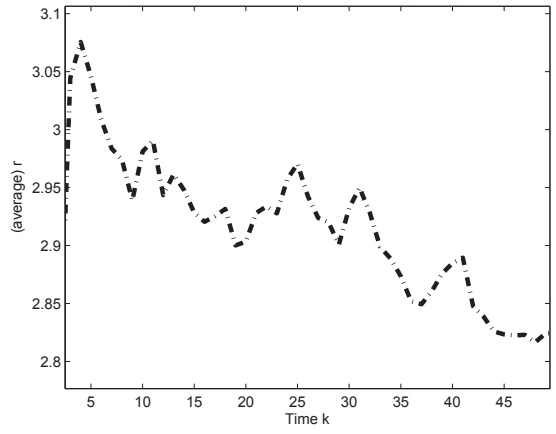
$$R_k = \begin{bmatrix} 1000 & 0 \\ 0 & 0.001 \end{bmatrix}$$




(a) Trajectory

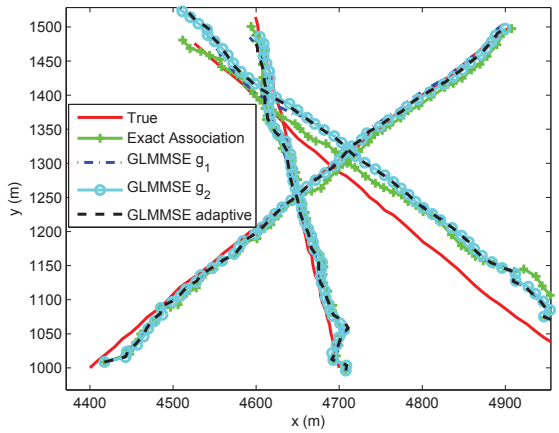



(b) OSPA

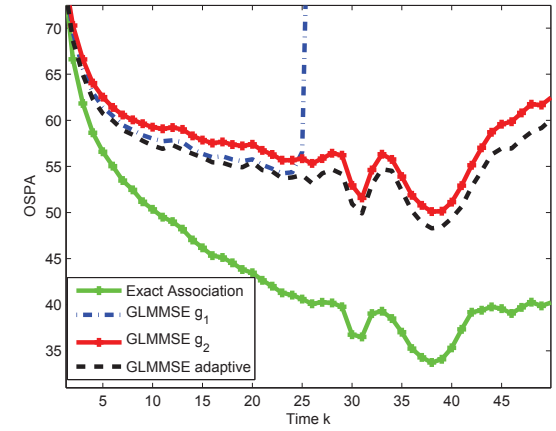



(c) Average $r_k$ for GL($g_{\text{adapt}}$) in Eq. (3.21)

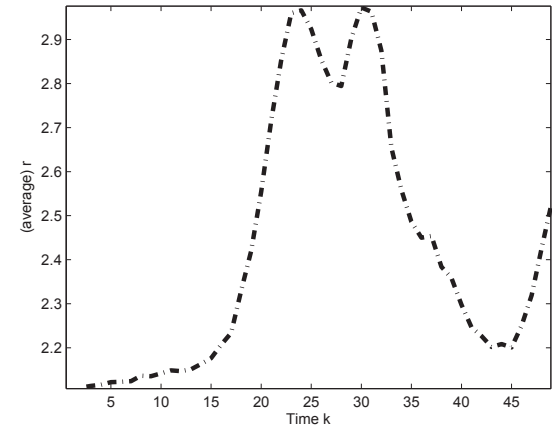
Figure 3.5: MTT (Case 1).




(a) Trajectory




(b) OSPA




(c) Average $r_k$ for GL($g_{\text{adapt}}$) in Eq. (3.21)

Figure 3.6: MTT (Case 2).

$r_l = 2.1$ and $r_u = 3$ are chosen for $g_{\text{adapt}}$. The true and estimated trajectories are given in Fig. 3.6a, and the estimation errors are shown in Fig. 3.6b. Due to the increase in measurement noise, all the filters have much larger estimation errors than in Case 1, and GL($g_1$) even diverges. The performance gaps among the UF with exact association, GL($g_2$) and GL($g_{\text{adapt}}$) become much larger than in Case 1. The average $r_k$ for GL($g_{\text{adapt}}$) is shown in Fig. 3.6c.

## 3.8 Summary

By introducing a vector-valued measurement transform function (MTF), the GLMMSE estimation finds the best estimator among all estimators that are linear in MTF, rather than in the measurement itself. With a properly designed MTF, GLMMSE estimation should have superior performance to LMMSE estimation due to the benefit of a more appropriate or enlarged candidate set of estimators (than the set of all linear estimators in LMMSE estimation) for searching. Although the rules for constructing an MTF is difficult to obtain for the general case, several design guidelines based on a numerical example have been provided to facilitate the design process. GLMMSE estimation and LMMSE estimation have similar formulas. Hence, similar to LMMSE estimation, moments involved in GLMMSE estimation are difficult to evaluate exactly in general. Fortunately, many numerical approximations for LMMSE estimation are also applicable to GLMMSE estimation. Approximation of GLMMSE estimation based on Gaussian density approximation and Gaussian-Hermite quadrature has been presented. Our GLMMSE estimation is demonstrated by applications to two nonlinear problems in target tracking.

# Chapter 4

# Recursive Joint Decision and Estimation

## 4.1 Introduction

Solutions to many practical problems involve both decision and estimation. Difficulties often arise when decision and estimation are tightly coupled. This is the so-called joint decision and estimation (JDE) problem. To be clear, we interpret estimation as inferring a continuous-valued (random or non-random) quantity, while decision is to make a choice from a discrete candidate set. JDE problems are not uncommon, such as target tracking and classification and multitarget tracking with an unknown number of targets. Inference of the target state is an estimation problem; determination of the target attribute or the number of targets is a decision process; and they clearly affect each other. The inter-dependence between estimation and decision often incurs additional difficulty for solving JDE problems. For example, without knowing the class or the number of targets, it is hard to estimate the target state. On the other hand, good decision relies heavily on accurate estimation of the target state. Conventional solutions ignore this inter-dependence either completely (e.g.,

separate estimation and decision) or partially (e.g., decision-then-estimation or estimation-then-decision), making their performance suffers.

This chapter follows the spirit of the batch JDE algorithm [143, 158] and develops a recursive implementation [167]. In many applications (e.g., radar and sonar), measurements are obtained sequentially. Thus a recursive JDE (RJDE) algorithm would be attractive and would fit the problem more naturally. RJDE uses and updates the existing results based only on the new data when they are available. Therefore, RJDE is computationally more efficient than a batch algorithm. It is important to point out that, in general, this RJDE algorithm is only an approximation of the batch JDE method and may have degraded performance. This is similar to the case where the optimal linear estimator does not always have a recursive form [142]. As explained later, if the decision partition of the data space and the expected estimation cost can be computed recursively, RJDE would have the same performance as batch JDE. However, this is difficult in practice. Approximations have been made in RJDE to obtain a recursive algorithm, and thus may result in degraded performance. Our RJDE is applied to a joint target tracking and classification problem and a multitarget detection and tracking problem. Its performance is demonstrated by the results of Monte Carlo simulation.

This chapter is organized as follows. The batch JDE method is reviewed in Sec. 4.2. Our RJDE algorithm is developed in Sec. 4.3. A joint performance measure for JDE algorithms is given in Sec. 4.4. Applications of RJDE to (a) joint target tracking and classification and (b) multitarget detection and tracking are presented in Sec. 4.5 and Sec. 4.6, respectively. A summary is made in Sec. 4.7.

## 4.2 Joint Decision and Estimation Algorithm

The Bayes decision risk is defined as

$$\bar{R}_D \triangleq \sum_i \sum_j c_{ij} P\{D_i, H_j\}$$

where $c_{ij}$ is the cost of deciding on $D_i$ while the true hypothesis is $H_j$. $P\{D_i, H_j\}$ is the joint probability of decision and hypothesis. The optimal decider minimizes this Bayes risk. For a JDE problem, this risk is generalized as [143]

$$\bar{R} \triangleq \sum_i \sum_j (\alpha_{ij} c_{ij} + \beta_{ij} E[C(x, \hat{x})|D_i, H_j]) P\{D_i, H_j\} \tag{4.1}$$

where $x$ is the quantity to be estimated and $\hat{x}$ the estimate, $C(x, \hat{x})$ is the estimation cost function. By introducing the weight coefficients $\{\alpha_{ij}, \beta_{ij}\}$ and the conditional expected estimation cost $E[C(x, \hat{x})|D_i, H_j]$, both estimation errors and decision errors contribute to $\bar{R}$, and the correlations between decision and estimation are accounted for explicitly. Although various $C(x, \hat{x})$ are possible (provided they satisfy some admissibility conditions), quite often the mean square error (mse) is adopted:

$$\begin{aligned} E[C(x, \hat{x})|D_i, H_j] &= E[(x - \hat{x})'(x - \hat{x})|D_i, H_j] \\ &= \text{mse}(\hat{x}|D_i, H_j) \end{aligned} \tag{4.2}$$

In the sequel, this square error is chosen as the estimation cost function unless stated otherwise. The weight coefficients (design parameters) $\alpha_{ij}$ and $\beta_{ij}$ are application-dependent and up to the user to choose.

The basic idea of the JDE algorithm [143,158], [250] searches the solution by minimizing $\bar{R}$ with an iteratively procedure, which is given below:

*Algorithm I*: The batch JDE algorithm.

1. Initialize the algorithm by an initial decision partition $\mathcal{D}^0 = \{\mathcal{D}_1^0, \mathcal{D}_2^0, \cdots, \mathcal{D}_m^0\}$ of the data space $\mathcal{Z}$. (Actually any set of decision region, not necessarily a partition, works in this framework [143]. For simplification, however, we only consider partition here.)

2. E-step: At each iteration $l$, for the given partition $\mathcal{D}^l = \{\mathcal{D}_1^l, \mathcal{D}_2^l, \cdots, \mathcal{D}_m^l\}$ of the data space $\mathcal{Z}$, compute the optimal state estimate $\hat{x}_l$ by minimizing $\bar{R}$. If the expected cost

function is chose to be Eq. (4.2), for $Z \in \mathcal{D}_i^l$, we have

$$\hat{x}_l = \check{x}_l^{(i)} = \sum_j \hat{x}_l^{(j)} \bar{P}^{(i)}\{H_j|Z\}$$
$$\hat{x}_l^{(j)} = E[x_l|Z, H_j]$$
$$\bar{P}^{(i)}\{H_j|Z\} = \frac{\beta_{ij} P\{H_j|Z\}}{\sum_h \beta_{ih} P\{H_h|Z\}}$$

3. D-step: Compute the conditional expected estimation cost based on $\mathcal{D}^l$

$$\begin{aligned} \varrho_{ij}^l &\triangleq E[C(x, \hat{x}_l)|D_i^l, H_j] \\ &= \int_{z \in \mathcal{D}_i^l} \int_{x \in \mathcal{X}} C(x, \hat{x}_l) dF(z, x|H_j), \quad \forall i, j \end{aligned} \tag{4.3}$$

where $\mathcal{X}$ is the state space of $x$. Given $\varrho_{ij}^l$, determine the new decision partition $\mathcal{D}^{l+1} = \{\mathcal{D}_1^{l+1}, \mathcal{D}_2^{l+1}, \cdots, \mathcal{D}_m^{l+1}\}$ by minimizing $\bar{R}$, which is a standard Bayes decision problem.

4. Repeat the E- and D-steps until the change of the generalized Bayes risk $\bar{R}$ is smaller than a prespecified threshold.

5. Output the final decision and estimation $\{\hat{D}, \hat{x}\}$.

The generalized Bayes risk, which is the foundation of the JDE method, extends the traditional Bayes risk in several aspects [143]: (a) It is a joint risk for both decision and estimation. The decision cost, the estimation error and their couplings are all considered in this framework. (b) Unlike hypothesis testing, the decision candidate set and the hypothesis set are not necessarily one-to-one in the JDE framework. (c) The estimation cost function $C(x, \hat{x})$ rather than $C(\tilde{x})$ is involved (where $\tilde{x} \triangleq x - \hat{x}$), which empowers the algorithm to cope with the case where $x$ and $\hat{x}$ are of different dimensions. This is a typical case in multitarget tracking with an unknown number of targets. Further, the coefficient $\beta_{ij}$ converts possibly incommensurable estimation risks $E\left[C(x, \hat{x}) \,|D_i, H_j\right]$ to a unified risk, rendering the

summation meaningful. (d) The weight coefficients $\{\alpha_{ij}, \beta_{ij}\}$ provide extra flexibility to fine tune the algorithm. The relative weights of the decision and estimation in a JDE problem are captured by the relative magnitudes of $\{\alpha_{ij}, \beta_{ij}\}$ [143, 158], [250].

Convergence of the batch JDE algorithm is guaranteed, since the E-step and the D-step make $\bar{R}$ at least non-increasing during iterations. Let

$$c'_{ij} = (\alpha_{ij} c_{ij} + \beta_{ij} E[C(x, \hat{x}) | D_i, H_j])$$

be the generalized cost. For a given expected estimation cost $E[C(x, \hat{x})|D_i, H_j]$, $c'_{ij}$ is fixed. Then, as in step 3, the partition $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_m\}$ is chosen to minimize

$$\bar{R} = \sum_i \sum_j c'_{ij} P\{D_i, H_j\}$$

similarly as the traditional Bayes decision procedure does. $\bar{R}$ is clearly non-increasing in this step. Next, for the given partition $\mathcal{D}$, it yields

$$\bar{R} = \sum_i \sum_j c_{ij} P\{D_i, H_j\} + \sum_i \sum_j \beta_{ij} E[C(x, \hat{x}) | D_i, H_j] P\{D_i, H_j\} \tag{4.4}$$

The first term in Eq. (4.4) is a constant and the second term can be minimized by choosing the optimal $\hat{x}$ (conditioned on the partition $\mathcal{D}$) as in step 2, which also does not increase $\bar{R}$. Thus the iteration converges, but not necessarily to the joint optimal solution in general, since the iterations may be "trapped" at some local optimum.

## 4.3 Recursive JDE

In many dynamic applications the measurements are observed sequentially. Although the JDE method of [143] is theoretically superior, because of its developed algorithm's batch na-

ture, it may be computationally inefficient. A recursive JDE (RJDE) algorithm is developed in this section. It avoids starting from scratch when new measurements become available.

The RJDE searches for a JDE solution recursively by minimizing Eq. (4.1) based on sequential data. Ideally, if the decision partition of the data space $\mathcal{Z}^k$ (the space of $Z^k$, where $Z^k$ is given in Eq. (4.8)) and $\varrho_{ij}$ of Eq. (4.3) can be calculated recursively, the RJDE and batch JDE would have the same results, which is, unfortunately, difficult in general. Hence some approximations are made to develop a recursive algorithm. As the data arrive sequentially for each time $k$, unlike the batch JDE which computes the decision partition of $\mathcal{Z}^k$, only the space $\mathcal{Z}_k$ of the current data $z_k$ is partitioned conditioning on all previous data $Z^{k-1}$. Further, the conditional expected estimation cost $\varrho_{ij}$ is approximated by $\varepsilon_{ij}^k$ of Eq. (4.7) at time $k$, which is computed based on the partition of $\mathcal{Z}_k$ with additional conditions on $Z^{k-1}$. Here, for simplicity, it is assumed that the underlying hypothesis $H_j$ does not change over time, but it can be generalized to a first-order Markov chain, as mentioned in Sec. 4.6.

*Algorithm II*: The RJDE algorithm.

1. Initialize the algorithm by the initial parameters, i.e., expected estimation cost $\varepsilon_{ij}^0$ and $P\{H_j\}$.

2. The posterior cost at time $k$ is

$$C_i^k(Z^k) = \sum_j c_{ij}^k P\{H_j|Z^k\} \tag{4.5}$$

where

$$c_{ij}^k = \alpha_{ij} c_{ij} + \beta_{ij} \varepsilon_{ij}^k \tag{4.6}$$

$$\varepsilon_{ij}^k = E[C(x_k, \hat{x}_k)|Z^{k-1}, D_i^k, H_j] \tag{4.7}$$

$$Z^k = [z_1', z_2', \cdots, z_k']' \tag{4.8}$$

3. At time $k+1$, the decision partition (conditioned on $Z^k$) of space $\mathcal{Z}_{k+1}$

$$\mathcal{D}^{k+1} = \{\mathcal{D}_1^{k+1}, \mathcal{D}_2^{k+1}, \cdots, \mathcal{D}_M^{k+1}\}$$

can be determined by

$$\mathcal{D}_i^{k+1} = \{z_{k+1} : C_i^k(z_{k+1}|Z^k) \leq C_l^k(z_{k+1}|Z^k), \forall l\}$$

where

$$C_i^k(z_{k+1}|Z^k) = \sum_j c_{ij}^k P\{H_j|z_{k+1}, Z^k\} \tag{4.9}$$

4. The state estimate $\hat{x}_{k+1}$ and conditional expected cost $\varepsilon_{ij}^{k+1}$ are computed based on $\{\mathcal{D}^{k+1}|Z^k\}$. Update $C_i^k(z_{k+1}|Z^k)$ to $C_i^{k+1}(z_{k+1}|Z^k)$ by replacing $c_{ij}^k$ in Eq. (4.9) with $c_{ij}^{k+1}$.

5. Recalculate the decision partition $\{\mathcal{D}^{k+1}|Z^k\}$ based on $C_i^{k+1}(z_{k+1}|Z^k)$ obtained in step 4.

6. Repeat steps 4 and 5 until the termination conditions (e.g., between two iterations the decision does not change and the change of the expected estimation cost is smaller than a prespecified threshold) are satisfied. Output the JDE solution $\{\hat{D}^{k+1},\ \hat{x}_{k+1}\}$.

7. let $k = k+1$ and go to step 2.

The iteration of steps 4 and 5 is guaranteed to converge, which can be proved similarly as for the batch JDE.

## 4.4 Performance Evaluations

Decision performance and estimation performance are often evaluated by the correct-decision rate and the estimation mean square error (mse) respectively. For a JDE problem, however, these two evaluation criteria are incomprehensive and often incapable of comparing different algorithms—e.g., one algorithm may have a higher correct-decision rate but a larger mse. In this case, it is not straightforward to tell which algorithm is superior. Further, these measures require precise knowledge of the ground truth, which is hardly available except in simulations. A joint performance measure (JPM) based on the statistical distance between the real measurement and the predicted measurement was proposed in [158]. Following the same spirit, we propose a JPM for dynamic problems. The mean prediction-measurement distance $\gamma_k$ is defined as

$$\gamma_k \triangleq \int\int d(z_k, \hat{z}_{k|k-1}) dF(\hat{z}_{k|k-1}|\hat{x}_{k-1}, \hat{\mathcal{D}}^{k-1}) \times dF(z_k, x_k, H_j) \tag{4.10}$$

where

$d(z_1, z_2)$—distance between $z_1$ and $z_2$, which is up to the user to choose;

$z_k$—real measurement at time $k$;

$\hat{z}_{k|k-1}$—one-step predicted measurement.

The philosophy of choosing the distance between the real measurement $z_k$ and the predicted measurement $\hat{z}_{k|k-1}$ as a performance measure lies in the following remarks:

> "A first-rate theory predicts; a second-rate theory forbids and a third-rate theory explains after the event." —Aleksander Isaakovich Kitaigordski [160]

and

> "A theory is a good theory if it satisfies two requirements: It must accurately describe a large class of observations ... and it must make definite predictions about the results of future observations." — Stephen Hawking [80]

Therefore, a better algorithm should predict the future measurement better (in terms of some distance measures), and hence, the mean prediction-measurement distance $\gamma_k$ is justified to be a performance measure. Further, $\gamma_k$ jointly evaluates the decision and estimation parts of the RJDE algorithm since the errors resulting from both parts contribute to $\gamma_k$. So, $\gamma_k$ is eligible to be a JPM.

In a discrete setting, $\gamma_k$ can be approximated by the sample average $\varepsilon_k$,

$$\gamma_k \approx \varepsilon_k \triangleq \frac{1}{I} \sum_{i=1}^{I} \varepsilon_k^{[i]}$$

where

$$\varepsilon_k^{[i]} = \frac{1}{J} \sum_{j=1}^{J} d(z_k^{[i]}, z_{k|k-1,j}^{[i]})$$

and

$\hat{z}_{k|k-1,j}^{[i]} \sim f(\hat{z}_k|\hat{x}_{k-1}^{[i]}, \hat{D}^{k-1,[i]})$—the $j$th one-step predicted measurement on the $i$th MC run;

$J$—total number of the one-step predicted measurements generated at each step;

$I$—total number of runs;

$z_k^{[i]}$—real measurement at time $k$ on the $i$th run.

Note that $\varepsilon_k$ can be computed even without knowing the ground truth, since the real data are used as a "ruler" to evaluate different algorithms. Other measures usually require the precise knowledge of the ground truth, which is hardly available except in simulations. The distance function $d(z_1, z_2)$ is problem dependent and needs to be carefully selected for different applications. Examples of $d(z_1, z_2)$ are given in the following sections, in which our RJDE algorithm is applied to JDE problems in target tracking.

## 4.5 Application to Joint Target Tracking and Classification

Now consider a simplified joint target tracking and classification problem. The target belongs to two possible classes $H_j,\ j = 1, 2$. Based on measurements, we want to simultaneously identify the target class and estimate its state. Clearly this is a JDE problem. The dynamics model and measurement model of the target are given by the following linear equation system

$$x_{k+1} = F_k^{(j)} x_k + w_k^{(j)} \tag{4.11}$$

$$z_k = H_k^{(j)} x_k + v_k^{(j)} \tag{4.12}$$

It is assumed that the target class does not change over time; the initial target state is normal distributed; the target class is Bernoulli distributed; and their distributions are all known:

$$x_0 \sim \mathcal{N}(\bar{x}_0, P_0) \tag{4.13}$$

$$P\{H_0\} = p_0,\ \ P\{H_1\} = 1 - p_0 \tag{4.14}$$

Further, suppose the standard Kalman filter assumptions [1,142] apply to this example: the process noise $w_k^{(j)}$ and measurement noise $v_k^{(j)}$ are mutually uncorrelated white Gaussian noise sequences with zero mean and covariances $Q_k^{(j)}$ and $R_k^{(j)}$ respectively. They are also uncorrelated with the initial state $x_0$.

The RJDE algorithm presented in Sec. 4.3 is given below:

1. Initialize the algorithm by Eqs. (4.13)–(4.14).

2. Assume at time $k$, $c_{ij}^k$ and $P\{H_j|Z^k\}$ are obtained from the previous loop.

3. When the new measurement $z_{k+1}$ is available, compute the state $\hat{x}_k^{(j)} \triangleq E[x_k|Z^k, H_j]$ and the corresponding MSE matrix $P_{k+1}^{(j)}$ by the Kalman filter [110, 111]

$$\hat{x}_{k+1}^{(j)} = \hat{x}_{k+1|k}^{(j)} + K_k^{(j)}(z_{k+1} - \hat{z}_{k+1|k}^{(j)}) \tag{4.15}$$

$$P_{k+1}^{(j)} = P_{k+1|k}^{(j)} - K_{k+1}^{(j)} S_{k+1}^{(j)} (K_{k+1}^{(j)})' \tag{4.16}$$

where

$$\begin{aligned}
\hat{z}_{k+1|k}^{(j)} &= H_{k+1}^{(j)} F_k^{(j)} \hat{x}_k^{(j)} \\
P_{k+1|k}^{(j)} &= F_k^{(j)} P_k^{(j)} (F_k^{(j)})' + Q_k^{(j)} \\
S_{k+1}^{(j)} &= H_{k+1}^{(j)} P_{k+1|k}^{(j)} (H_{k+1}^{(j)})' + R_{k+1}^{(j)} \\
K_{k+1}^{(j)} &= P_{k+1|k}^{(j)} (H_{k+1}^{(j)})' (S_{k+1}^{(j)})^{-1}
\end{aligned}$$

Then, the posterior probability of target class $H_j$ follows

$$\begin{aligned}
P\{H_j|Z^{k+1}\} =& \frac{f(z_{k+1}|Z^k, H_j) P\{H_j|Z^k\}}{\sum_l f(z_{k+1}|Z^k, H_l) P\{H_l|Z^k\}} \\
=& \frac{N_{k+1}^{(j)}(z_{k+1}) P\{H_j|Z^k\}}{\sum_l N_{k+1}^{(l)}(z_{k+1}) P\{H_l|Z^k\}}
\end{aligned}$$

where $N_{k+1}^{(j)}(z_{k+1}) = \mathcal{N}(z_{k+1}; \hat{z}_{k+1|k}^{(j)}, S_{k+1}^{(j)})$.

4. Calculate the intermediate cost $C_i^k(z_{k+1}|Z^k)$ by Eq. (4.9).

5. The decision partition $\mathcal{D}^{k+1} = \{\mathcal{D}_0^{k+1}, \mathcal{D}_1^{k+1}\}$ is determined by

$$(c_{01}^k - c_{11}^k) L^{k+1} \underset{D_0}{\overset{D_1}{\gtrless}} (c_{10}^k - c_{00}^k)$$

where

$$L^{k+1} \triangleq \frac{P\{H_1|Z^{k+1}\}}{P\{H_0|Z^{k+1}\}} = \frac{f(z_{k+1}|Z^k, H_1)}{f(z_{k+1}|Z^k, H_0)} L^k$$

6. Based on the decision region $\mathcal{D}^{k+1}$, if $z^{k+1} \in \mathcal{D}_i^{k+1}$, a JDE solution is computed

$$\hat{D}^{k+1} = \{D_i : z_{k+1} \in \mathcal{D}_i^{k+1}\} \tag{4.17}$$

$$\hat{x}_{k+1} = \check{x}_{k+1}^{(i)} \triangleq \sum_j \hat{x}_{k+1}^{(j)} \frac{\beta_{ij} P\{H_j|Z^{k+1}\}}{\sum_l \beta_{il} P\{H_l|Z^{k+1}\}} \tag{4.18}$$

and $\varepsilon_{ij}^{k+1}$ is updated by

$$\begin{aligned}
\varepsilon_{ij}^{k+1} &\triangleq \mathrm{mse}(\hat{x}|Z^k, D_i, H_j) \\
&= E[\mathrm{mse}(\hat{x}^{(ij)}|Z^{k+1}, D_i, H_j)|Z^k, D_i, H_j] + E[(\hat{x}^{(ij)} - \hat{x})^2|Z^k, D_i, H_j] \\
&= E[\mathrm{mse}(\hat{x}^{(j)}|Z^{k+1}, H_j)|Z^k, D_i, H_j] + E[(\hat{x}^{(j)} - \check{x}^{(i)})^2|Z^k, D_i, H_j]
\end{aligned}$$

where the index $k+1$ for $D_i$ and $x$ is dropped for simplicity. The third equality holds if $z^{k+1} \in \mathcal{D}_i^{k+1}$ since

$$\begin{aligned}
\hat{x}_{k+1}^{(ij)} &\triangleq E[x|Z^{k+1}, D_i^{k+1}, H_j] \\
&= \int x f(x|Z^{k+1}, D_i^{k+1}, H_j) dx \\
&= \int x f(x|Z^{k+1}, H_j) dx \\
&= E[x|Z^{k+1}, H_j] = \hat{x}_{k+1}^{(j)}
\end{aligned}$$

and by Eq. (4.18), it yields $\hat{x}_{k+1} = \check{x}_{k+1}^{(i)}$. These quantities are not defined if $z^{k+1} \notin \mathcal{D}_i^{k+1}$. Further, in this case, $\mathrm{mse}(\hat{x}_{k+1}^{(j)}|Z^{k+1}, H_j) = \mathrm{tr}(P_{k+1}^{(j)})$ (see Eq. (4.16) for $P_{k+1}^{(j)}$)

does not depend on measurement $Z^{k+1}$, so

$$\begin{aligned}\varepsilon_{ij}^{k+1} &= E[\mathrm{tr}(P_{k+1}^{(j)})|Z^k, D_i^{k+1}, H_j] + \tilde{\varepsilon}_{ij}^{k+1} \\ &= \mathrm{tr}(P_{k+1}^{(j)}) + \tilde{\varepsilon}_{ij}^{k+1}\end{aligned}$$

where $\tilde{\varepsilon}_{ij}^{k+1} \triangleq E[(\check{x}_{k+1}^{(i)} - \hat{x}_{k+1}^{(j)})^2|Z^k, D_i^{k+1}, H_j]$ is difficult to calculate. It can be approximated by $\tilde{\tilde{\varepsilon}}_{ij}^{k+1}$ numerically using the Monte Carlo method:

$$\begin{aligned}\tilde{\varepsilon}_{ij}^{k+1} \approx \tilde{\tilde{\varepsilon}}_{ij}^{k+1} &\triangleq E[(\check{x}_{k+1}^{(i)} - \hat{x}_{k+1}^{(j)})^2|\hat{x}_k^{(j)}, D_i^{k+1}, H_j] \\ &= \int_{z_{k+1}\in\mathcal{D}_i^{k+1}} (\check{x}_{k+1}^{(i)} - \hat{x}_{k+1}^{(j)})^2 dF(z_{k+1}|\hat{x}_k^{(j)}, H_j) \\ &\approx \frac{1}{L_i}\sum_{l_i=1}^{L_i}[\check{x}_{k+1}^{(i)}(z_{k+1,l_i}^{(ij)}) - \hat{x}_{k+1}^{(j)}(z_{k+1,l_i}^{(ij)})]^2\end{aligned}$$

where $z_{k+1,l_i}^{(ij)}(l_i = 1, 2, \cdots, L_i)$ are the simulated measurements (from the distribution $f(z_{k+1}|\hat{x}_k^{(j)}, H_j)$) that lie inside the decision region $\mathcal{D}_i^{k+1}$. And

$$\begin{aligned}\hat{x}_{k+1}^{(j)}(z_{k+1,l_i}^{(ij)}) &= E[x_{k+1}|z_{k+1,l_i}^{(ij)}, Z^k, H_j] \\ \check{x}_{k+1}^{(i)}(z_{k+1,l_i}^{(ij)}) &= \sum_j \hat{x}_{k+1}^{(j)}(z_{k+1,l_i}^{(ij)}) \frac{\beta_{ij}P\{H_j|z_{k+1,l_i}^{(ij)}, Z^k\}}{\sum_l \beta_{il}P\{H_l|z_{k+1,l_i}^{(ij)}, Z^k\}}\end{aligned}$$

can be calculated by the Kalman filter. If some $\mathcal{D}_i^{k+1}$ are empty, then $\hat{x}_{k+1}^{(j)}(z_{k+1,l_i}^{(ij)})$ and $\check{x}_{k+1}^{(i)}(z_{k+1,l_i}^{(ij)})$ can be replaced by predictions.

7. Based on the updated estimation error $\{\varepsilon_{ij}^{k+1}\}$, $c_{ij}^k$ us updated to $c_{ij}^{k+1}$ by replacing $\varepsilon_{ij}^k$ with $\varepsilon_{ij}^{k+1}$ in Eq. (4.6).

8. The decision partition $\mathcal{D}^{k+1}$ is recalculated according to $c_{ij}^{k+1}$:

$$(c_{01}^{k+1} - c_{11}^{k+1})L^{k+1} \underset{D_0}{\overset{D_1}{\gtrless}} (c_{10}^{k+1} - c_{00}^{k+1})$$

9. Go to step 6 until the termination criterion is satisfied. Output the JDE solutions $\{\hat{D}^{k+1},\ \breve{x}_{k+1}^{(i)}\}$ at time $k+1$ by Eqs. (4.17)–(4.18).

10. Let $k = k+1$ and go to step 2.

Two cases of an illustrative numerical example and their simulation results are given later. The RJDE method is compared with the traditional decision-then-estimation (DTE) and estimation-then-decision (ETD) methods in terms of the root mean square error (RMSE), the probability of correct classification ($P_c$) and the joint performance measure (JPM) $\varepsilon_k$. The performance of the ideal case (which always classifies the target correctly) is also provided as a performance bound.

In the DTE method, decision is made first at each time $k$ based on the ratio $L^k$ of the posterior probabilities of $H_1$ and $H_0$ (provided $c_{10} > c_{00}$ and $c_{01} > c_{11}$):

$$\begin{aligned} &\text{if } L^k > \frac{c_{10}-c_{00}}{c_{01}-c_{11}}, \quad \text{decide } \hat{D}^k = 2; \\ &\text{else,} \quad \text{decide } \hat{D}^k = 1. \end{aligned}$$

Then estimation is obtained by assuming this decision is always correct, $\hat{x}_k = E[x|Z^k, H_{\hat{j}_k}]$. Since the decision part is optimal in that it minimizes the Bayes decision risk, this method should performs the best in terms of the $P_c$ (if $c_{ij}$ is chosen as in Table 4.1).

In the ETD method, the best estimate of the target state is obtained by the autonomous multiple model (AMM) algorithm [152], $\hat{x}_k = \sum_{j=1}^{2} \hat{x}_k^{(j)} \mu_k^{(j)}$, where $\mu_k^{(j)}$ is the model probability. (AMM is chosen since the target class is invariant overtime. See [152] for details.) This step is optimal in the sense that it minimizes the unconditional estimation mse. Then decision is made based on the ratio of the measurement likelihoods conditioned on $\hat{x}_k$ and $H_j$.

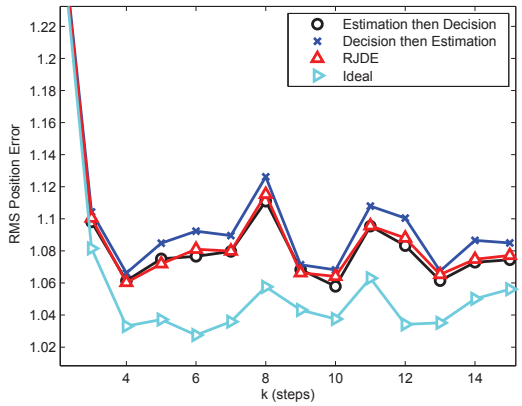
Although the first steps of these two two-stage algorithms are optimal (optimal decider and optimal estimator respectively) in their own domains (in terms of Bayes decision risk and estimation mse respectively), the JDE algorithm has the potential to simultaneously beat

them in terms of both these two criteria. This relies on specific types of available data: [143] elaborates the case and [158] gives an example and simulation results.

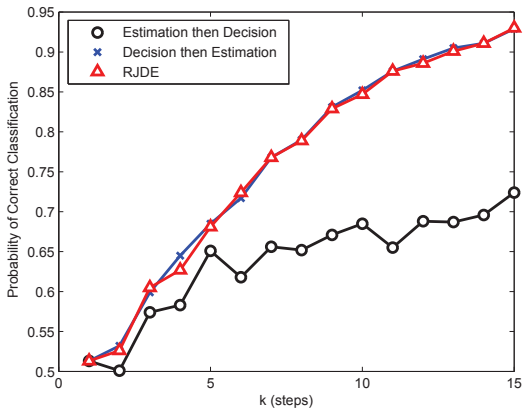
Two numerical cases with different parameter sets are given in Table 4.1 and simulation results are given in Figs. 4.1 and 4.2. We are trying to jointly track and classify the target. The ground truth was generated according to Eqs. (4.11)–(4.12). All the simulation results were obtained from 1000 MC runs. The initial target state was Gaussian distributed with mean $\bar{x}_0$ and covariance $P_0$ and the target class was Bernoulli distributed with $p_0 = 0.5$.

*Case 1*: The dynamics matrix $F_k$ depends on the target class but the measurement matrix $H_k$ does not. Since there is no model ambiguity, obviously the ideal case sets the performance lower bounds for RMSE and JPM and an upper bound ($P_c = 1$) for $P_c$. In terms of RMSE, the ETD performs the best, as expected, the RJDE is in the middle, and the DTE is the worst. Their performance differences are not significant. With respect to $P_c$, RJDE and DTE have almost the same performance and are much better than ETD. However, as mentioned before, each of these criteria only measures one aspect of algorithm's performance for the JDE problem. In terms of JPM, our RJDE method beats the two traditional methods, meaning that it can make a better tradeoff between the estimation error and decision error. DTE also outperforms ETD in terms of JPM, since DTE has much better decision accuracy than ETD does, while DTE is only slightly worse than ETD in terms of RMSE.

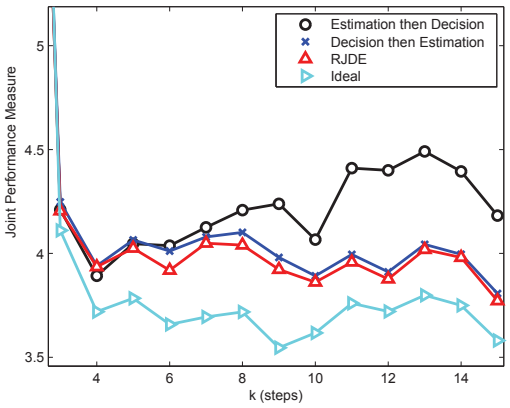
*Case 2*: $F_k$ and $H_k$ are all dependent on the target class. The measurement noise covariance $R$ is also increased to a larger value than in case 1. In terms of RMSE and $P_c$, the RJDE method still performs in the middle of the other two algorithms. But unlike case 1, the performance differences among these three algorithms are much more significant, especially in RMSE. RJDE is still the best in terms of JPM, while ETD is better than DTE in this case. The simulation results also show that the correct-decision rate and estimation mse together are incapable of telling which algorithm is better for a JDE problem, since an algorithm may win in one but lose in the other. So, the power of JPM emerges since it provides a unified measure and evaluates the overall performance of decision and estimation.




(a) RMSE

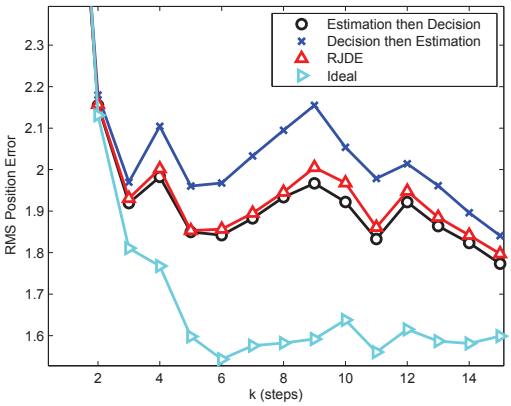



(b) $P_c$

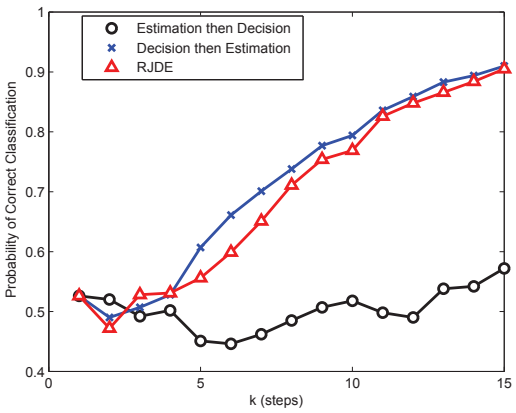



(c) JPM

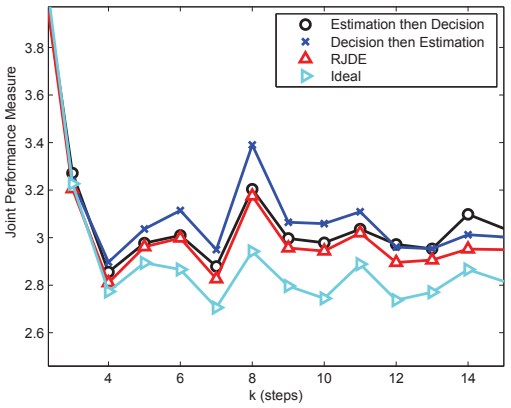
Figure 4.1: Joint target tracking and classification, Case 1.




(a) RMSE




(b) $P_c$




(c) JPM

Figure 4.2: Joint target tracking and classification, Case 2.

Table 4.1: Simulation parameters

| Parameter | $[F^{(1)}, F^{(2)}]_k$ | $[H^{(1)}, H^{(2)}]_k$ | $\alpha_{ij}$ | $\beta_{ij}$ | $c_{ij}$ | $R_k^{(j)}$ | $\bar{x}_0$ | $P_0$ |
|---|---|---|---|---|---|---|---|---|
| Case 1 | $[1,\ 1.2]$ | $[1,\ 1]$ | 1 | $\begin{bmatrix} 0.5 & 0.2 \\ 0.2 & 0.5 \end{bmatrix}$ | $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ | 2 | 1 | 10 |
| Case 2 | $[1,\ 1.2]$ | $[1,\ 0.8]$ | 1 | $\begin{bmatrix} 0.5 & 0.25 \\ 0.25 & 0.5 \end{bmatrix}$ | $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ | 5 | 1 | 10 |

## 4.6 Application to Multitarget Detection and Tracking

In this section, RJDE is applied to multitarget tracking (MTT) in clutter with an unknown number of targets. In general, this is a tough problem and is still open in the tracking community. Most existing methods follow the decision-then-estimation, that is, determine (estimate) the number of targets first and then estimate the target state based on the determined number of targets. This problem is actually a JDE problem since both the number of targets and their states are of interest and they are clearly tightly coupled. Our RJDE algorithm provides a easily implementable framework to handle this difficult scenario. For simplicity, the following assumptions are made:

1. The number of targets $m_k^t$ is unknown but constant over time $k$, and its maximum is $N$.

2. One target can only generate at most one measurement (with detection probability $P_d$), and one measurement only has one source—either from a target or clutter. This is a common assumption in MTT.

3. The number $m_f$ of false measurements is Poisson distributed $P_f\{m_f\}$. The false measurements are i.i.d. and uniformly distributed in the surveillance region, i.e., $f_f(z) = 1/V$, where $V$ is the volume of surveillance region. The true measurements are independent of false measurements.

4. All targets follow a CV model with linear position measurements. That is, for a 2-Dimensional case, a target's dynamics and measurement models are

$$x_k = Fx_{k-1} + w_k \tag{4.19}$$

$$z_k = Hx_k + v_k \tag{4.20}$$

where

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

and

$$x_k = [\mathrm{x}\ \dot{\mathrm{x}}\ \mathrm{y}\ \dot{\mathrm{y}}]' \tag{4.21}$$

is the target state vector with position and velocity in the Cartesian coordinate system, $T$ is the sampling period.

The goal of JDE is to jointly infer the number of targets and their states. Let $H_j$ denote the hypothesis that $j$ $(1 \leq j \leq N)$ targets are present in the surveillance region and $D_i^k$ the decision that $i$ $(1 \leq i \leq N)$ targets are present at time $k$. For simplicity and without loss of generality, we do not consider the case that no target is in the surveillance region. Let $Z^k = \{z_1, z_2, \cdots, z_k\}$ denote the set of all measurements up to time $k$ and $z_k = \{z_k^1, z_k^2, \cdots, z_k^{m_k^z}\}$ the set of measurements at scan $k$, where $z_k^i$ is the $i$th measurement and $m_k^z$ is the number of measurements at $k$. Denote by $\mu_k^j \triangleq P\{H_j|Z^k\}$ the posterior probability of each hypothesis and by $\hat{X}_k^j \triangleq E[X_k|Z^k, H_j]$ the posterior mean conditioned on each $H_j$. Note that conditioned on hypothesis $H_j$, $X_k$ is a stacked vector of $j$ target states,

that is, $X_k = [(x_k^1)', (x_k^2)', \cdots, (x_k^j)']'$. We choose the expected estimation cost to be

$$\begin{aligned} \varepsilon_{ij}^k &= E[C(X_k, \hat{X}_k)|Z^{k-1}, D_i^k, H_j^k] \\ &\approx \begin{cases} \frac{\tau(i)}{i}\text{mse}(\hat{X}_k|D_i^k, H_j), & \text{if } i = j \\ \eta, & \text{if } i \neq j \end{cases} \end{aligned} \tag{4.22}$$

where $\eta$ is a cost parameter, $\tau(i)$ is a non-increasing positive function of $i$ and $\tau(1) = 1$, $\hat{X}_k$ is the estimate of $X_k$ which depends on the decision $D_i^k$. That is, conditioned on decision $D_i^k$, $\hat{X}_k$ is a stacked vector of $i$ target estimates. The expected cost $\varepsilon_{ij}^k$ is defined this way since $X_k$ and $\hat{X}_k$ have different dimensions if $D_i^k$ does not match $H_j$, rendering the "estimation error" not well defined. The cost for this mismatch case can be reflected by a cost parameter $\eta$, which may be either a constant or a function of $D_i^k$ and $H_j$. For the case that $D_i^k$ matches $H_j$, the normalized mean square error (mse) is adopted. The normalization factor is a ratio between the determined number $i$ of targets and an adjustment function $\tau(i)$. If an algorithm tracking 10 targets has the same average mse (i.e., normalized only by factor $i$) as an algorithm tracking only 1 target, it makes much sense to say the first algorithm does a better job than the second. $\tau(i)$ is introduced to favor the algorithms tracking more targets.

RJDE for multitarget detection and tracking:

1. Initialized the algorithm:

   Conditioned on each hypothesis $H_j$, initialize the state $\hat{X}_0^j$ and its MSE $P_0^j$. Initialize the probability $\mu_0^j = P\{H_j\} = 1/N$.

2. At time $k$, conditioned on each hypothesis $H_j$, the conditional estimate $\hat{X}_k^j$ and its MSE $P_k^j$ are computed based on $\hat{X}_{k-1}^j$ and $P_{k-1}^j$. This is actually a multitarget tracking with a known number of targets (hypothesized by $H_j$), which can be solved by many algorithms. We adopt the joint probabilistic data association filter (JPDAF) [21, 65] for its popularity and simplicity. The fundamental idea of JPDAF is to compute the probabilities of all feasible measurement-to-target association events $\theta_k^i$ jointly. The

track update is obtained by the probabilistic average over all $\theta_k^i$. Details of JPDAF are given in Appendix A.

3. The posterior probability of each hypothesis $H_j$ is updated

$$\mu_k^j = \frac{1}{c} f(z_k|H_j, Z^{k-1})\mu_{k-1}^j$$

where

$$f(z_k|H_j, Z^{k-1}) = \sum_i f(z_k|H_j, \theta_k^i, m_k^z, Z^{k-1}) P\{\theta_k^i|H_j, m_k^z\}$$

is the likelihood of $H_j$, the summation is over all possible $\theta_k^i$, and $c$ is a normalizing constant. $f(z_k|H_j, \theta_k^i, m_k^z, Z^{k-1})$ and $P\{\theta_k^i|H_j, m_k^z\}$ are provided by JPDAF in Appendix A.

4. Based on the expected cost function (4.22), once a decision $D_i^k$ is made, the corresponding state estimate is $\hat{X}_k = \hat{X}_k^i$. So we have

$$\varepsilon_{ii}^k = \frac{\tau(i)}{i}\text{mse}(\hat{X}_k|D_i^k, H_i) = \frac{\tau(i)}{i}\text{mse}(\hat{X}_k^i|D_i^k, H_i) \approx \frac{\tau(i)}{i}\text{tr}(P_k^i)$$

and $\varepsilon_{ij}^k = \eta$ if $i \neq j$. Then we have

$$c_{ij}^k = \alpha_{ij}c_{ij} + \beta_{ij}\varepsilon_{ij}^k$$

5. Decision is made by

$$D_l^k = \{l : C_l(Z^k) \leq C_i(Z^k), \forall i\}$$

where

$$C_i(Z^k) = \sum_{j=1}^{N} c_{ij}^k \mu_k^j$$

is the posterior cost. Output the result $\{D_l^k, \hat{X}_k^l\}$ at time $k$.

6. Let $k = k + 1$ and loop to Step 2.

Note that the JPDAF may be replaced by other MTT algorithms, e.g., nearest neighbor filter (NNF) or multiple hypothesis tracker (MHT).

The above algorithm can be generalized to the case that the number of targets is time varying. Then $H_j^k$ is time dependent and it may be modeled as a first-order Markov chain. The hypothesis transition probability $P\{H_i^k|H_j^{k-1}\}$ needs to be introduced in updating $\mu_k^j$. Further $\hat{X}_k^j = E[X_k|Z^k, H_j^k]$ must be approximated since it is an average over all possible hypothesis sequences up to $k$ that end up with $H_j^k$, and the number of such sequences increases exponentially as time goes. This is similar to the computation of model conditional estimate in multiple model estimation [152], where many approximations are available. For example, we may approximate $\hat{X}_k^j$ based on the most likely sequence.

Five numerical cases are given to demonstrate the performance of our RJDE algorithm for multitarget detection and tracking. Our JDE method is compared with a decision-then-estimation (DTE) method. All simulation results are obtained by Monte Carlo (MC) simulation with 1000 runs.

### 4.6.1 Ground Truth

For each target, the state $x_t$ is defined by Eq. (4.21). Each target's position is initialized by uniformly sampling from a region $[100\ 100]' \times [100\ 100]$ in the Cartesian coordinate system. The target dynamics and measurement models are given in Eqs. (4.19) and (4.20) with process noise covariance $Q = \text{diag}[1, 0.01, 1, 0.01]$ and measurement noise covariance $R = \text{diag}[100, 100]$, respectively. The sampling period is $T = 1$ second.

For each target, the initial position estimate is sampled uniformly from the region $[500\ 500]' \times [500\ 500]'$ and the velocities are set to zero. The corresponding MSE matrix is

$$P_0 = \text{diag}[1000^2, 20^2, 1000^2, 20^2]$$

The surveillance region is $[1500\ 1500]' \times [1500\ 1500]'$ in the 2-dimensional Cartesian coordinate system. The number of false measurements at each time step $k$ is sampled from a Poisson distribution

$$P_f\{m\} = \frac{e^{-\lambda V}(-\lambda V)^m}{m!}$$

where $\lambda$ is the clutter density and $V$ is the volume of the surveillance region. All the false measurements are uniformly distributed within the region and independent of each other.

The parameters in our RJDE algorithm are chosen to be

$$\alpha_{ij} = 1,\ \beta_{ij} = 1,\ c_{ij} = 150|i-j|,\ \eta = 100$$

The decision cost $c_{ij}$ is proportional to the error of the determined number of targets. The DTE method decides the number of targets based on this $c_{ij}$ first and then estimates the state based on this decision. Also, the performance of a JPDAF knowing the true number of targets (we call it the "ideal algorithm") is also given, which sets a performance lower bound for the other algorithms.

The following performance measures are evaluated:

1. Optimal subpattern assignment (OSPA) metric [220]:

   It is a refined measure from the measures proposed in [55, 57, 86, 210] for MTT with an unknown number of targets. For the true number $m_k^t$ and the determined number $\hat{D}^k$

of targets, let

$$m = \min\{m_k^t,\ \hat{D}^k\},\ n = \max\{m_k^t,\ \hat{D}^k\}$$

Then the OSPA is defined as

$$D(X_k, \hat{X}_k) = \left[\frac{1}{n}\left(\min_{\pi\in\Pi_n}\sum_{i=1}^{m}(\min[d_o(x_k^i, \hat{x}_k^{\pi(i)}), c])^p + c^p(n-m)\right)\right]^{1/p}$$

where $x_k^i$ and $\hat{x}_k^i$ are the position vectors of the $i$th target in $X_k$ and $\hat{X}_k$ (velocity errors can be computed similarly based on the velocity entries), respectively, $c$ and $p$ ($c = 100, p = 2$ in our simulation) are design parameters, $\Pi_n$ is the set of all possible permutations of $\{1, \cdots, n\}$, and the Euclidean distance is chosen for function $d_o$.

2. Joint performance measure (JPM):

   For the true number $m_k^z$ and the predicted number $\hat{m}_k^z$ of measurements, let

$$m = \min\{m_k^z,\ \hat{m}_k^z\}$$

The predicted measurement $\hat{z}_{k|k-1}$ is generated based on $\hat{X}_{k-1}$ and Eqs. (4.19) and (4.20). Then we choose the distance function $d$ in the JPM (Eq. (4.10)) as

$$d(z_k, \hat{z}_{k|k-1}) = \left[\frac{1}{N}\left(\min_{\pi\in\Pi_n}\sum_{i=1}^{m}(z_k^i - \hat{z}_k^{\pi(i)})'(z_k^i - \hat{z}_k^{\pi(i)}) + c^2(N-m)\right)\right]^{1/2} \tag{4.23}$$

where $z_k^i$ and $\hat{z}_k^i$ are the $i$th measurement in $z_k$ and $\hat{z}_{k|k-1}$, respectively, $c$ ($c = 10^2$ in our simulation) is a design parameter which penalizes an algorithm with a small determined number of targets. $N$ is the maximum possible number of targets in the surveillance region. The OSPA and our JPM function similarly. They both count estimation errors and decision errors. However, OSPA requires precise knowledge of the ground truth, while JPM doesn't.

3. Probability of decision error ($P_e$):

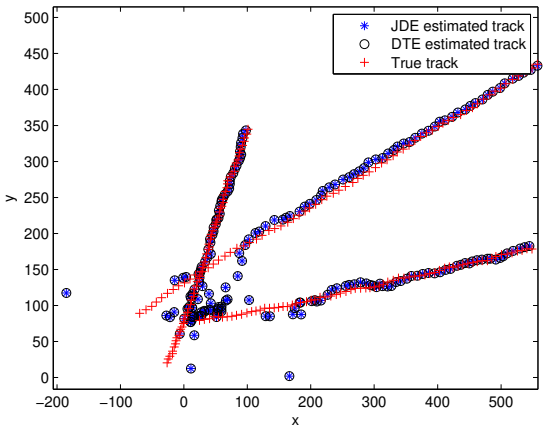
   The probability of making a wrong decision on the number of targets at each time step.

### 4.6.2 Case 1, Moderate Clutter Density

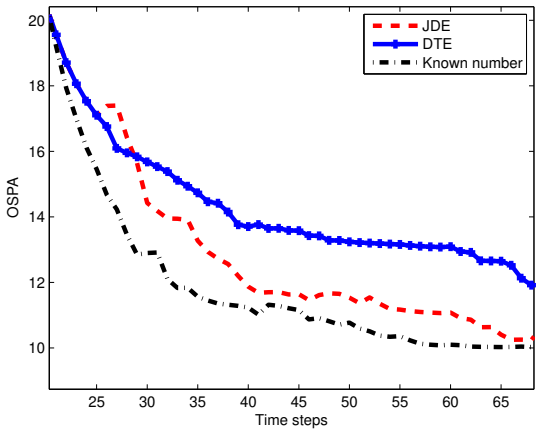
In this case, we choose the clutter density $\lambda = 10/V$, target detection probability $P_d = 0.95$, and the maximum number of targets $N = 3$. The number of targets is uniformly sampled from 1 to $N$ on each run and remains constant. The simulation results are given in Fig. 4.3. One realization of the true and estimated target trajectories are given in Fig. 4.3a. The OSPA and JPM are given in Figs. 4.3b and 4.3c, respectively. The average decision error rate is shown in Fig. 4.3d. The ideal algorithm performs the best in terms of OSPA and JPM, as expected. Clearly, JDE is superior to DTE in terms of all three performance measures. We found that JDE algorithm is less effective to determine the number of targets at beginning (in transient time) due to a large initial estimation error, while DTE method is less vulnerable to this error since the estimation error is not considered in decision. Therefore, we can take advantage of the flexibility of our JDE algorithm and use the results from DTE for the first 25 steps (and thereafter switch to the full RJDE algorithm) to mitigate the negative impact of the large initial errors. The RJDE algorithm is implemented in the same way in the following cases.

### 4.6.3 Case 2, Heavy Clutter Density

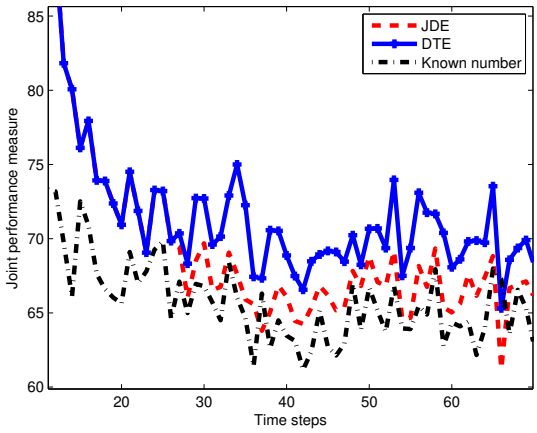
The clutter density is increased to $\lambda = 20/V$, while other parameters are the same as in Case 1. The simulation results are given in Fig. 4.4. Comparing with Case 1, the increase in clutter density has little impact on the ideal algorithm, as expected, while JDE and DTE deteriorate significantly.



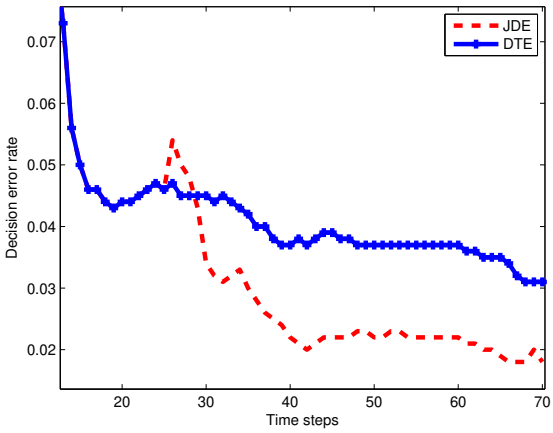

(a) Trajectory




(b) OSPA




(c) JPM




(d) Decision error rate

Figure 4.3: Simulation results of MTT, Case 1. The number of targets is randomly sampled on each run and remains constant. $\lambda = 10/V$ and $P_d = 0.95$.

### 4.6.4 Case 3, Low Detection Rate

In this case, we set clutter density $\lambda = 5/V$ and detection rate $P_d = 0.75$. A low detection rate may be encountered when we track a dim target or when the detection threshold is increased to suppress the number of false measurements. The simulation results are given in Fig. 4.5. Comparing with Case 1, the performance of all three algorithms deteriorate significantly, and their performance differences become much smaller than in Case 1, meaning that they are all sensitive to $P_d$. This makes sense since missing a true measurement indeed has a great impact on these algorithms.

### 4.6.5 Case 4, Varying Number of Targets

In the previous cases, the number of targets is constant over each run. In this case, we examine their capabilities of tracking a varying number of targets in the surveillance region. The hypothesis transition probability is chosen to be

$$P\{H_i^k | H_i^{k-1}\} = 0.98, \ i = 1, 2, 3$$
$$P\{H_i^k | H_j^{k-1}\} = 0.01, \ 1 \leq i \neq j \leq 3$$

Assume initially three targets are present in the surveillance region. Two targets disappear at time $k = 100$ and a new target appears at time $k = 200$. The results are given in Fig. 4.6. Large errors (peaks) occur at the time of target death or birth. All the algorithms take some time to adapt to the change in the number of targets, while JDE converges slightly faster than DTE.

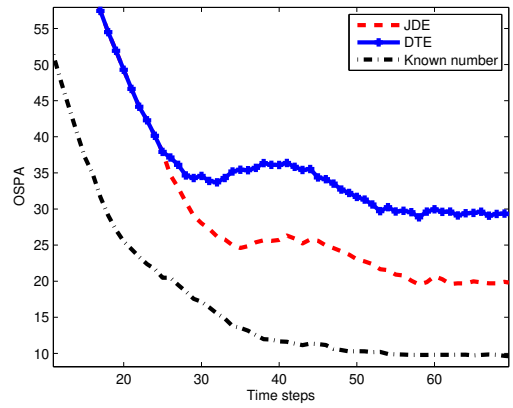
### 4.6.6 Case 5, Simultaneous Target Birth and Death

In this case, the number of targets is uniformly sampled from 1 to $N$ on each run (same as Cases 1). At time $k = 50$, a target disappears and a new target appears simultaneously. The same hypothesis transition probability as in Case 4 is used and the results are given in

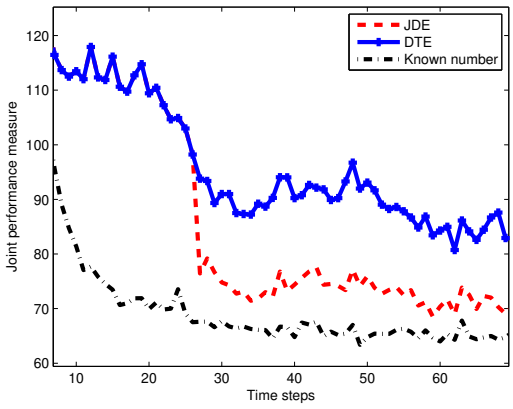
Fig. 4.7. Although the number of targets on each run remains constant in this case, the death and birth of targets incur large errors for all algorithms, as expected. Similarly as in Case 4, RJDE adapts to the change slightly faster than DTE. Note that although OSPA and JPM are computed in different spaces (state space and measurement space, respectively), they always have similar patterns in these cases. This happens due to the specific choice of the distance function (i.e., Eq. (4.23)) and the direct measurement of target position (Eq. (4.20)).

## 4.7 Summary

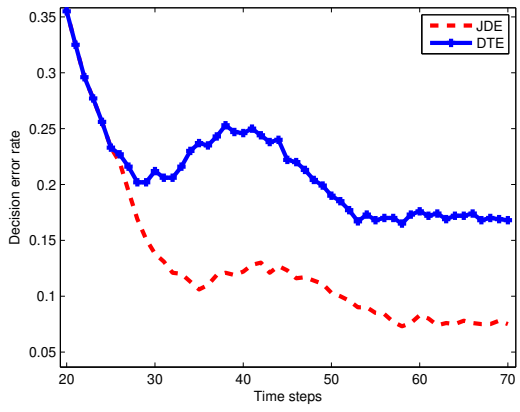
A recursive joint decision and estimation (RJDE) algorithm is proposed in this chapter. It fits dynamic JDE problems well since usually measurements are made and processed sequentially. The RJDE is an approximate recursive version of the (batch) JDE method based on a generalized Bayes risk and thus inherits its virtues and theoretical superiorities. The JDE method explicitly considers the inter-dependence between decision and estimation. Our RJDE algorithm is applied to two problems: (a) joint target tracking and classification and (b) multitarget detection and tracking. Its performance is demonstrated by the results of Monte Carlo simulation. Further, a general joint performance measure for evaluating JDE algorithms is proposed. The measure is in the measurement space since the true measurements are a good "reference point" to rank different JDE algorithms, especially when the ground truth is unknown. To our knowledge, this is the only comprehensive and systematic measure available to evaluate the performance of dynamic JDE algorithms.




(a) OSPA

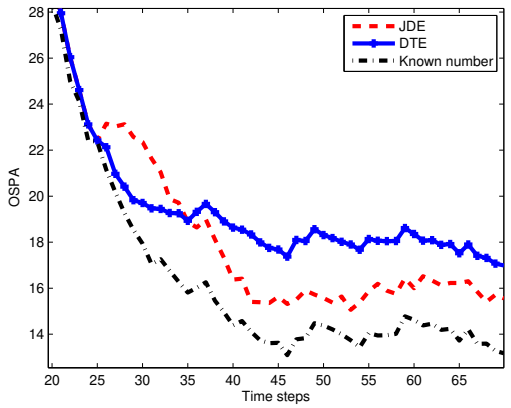



(b) JPM

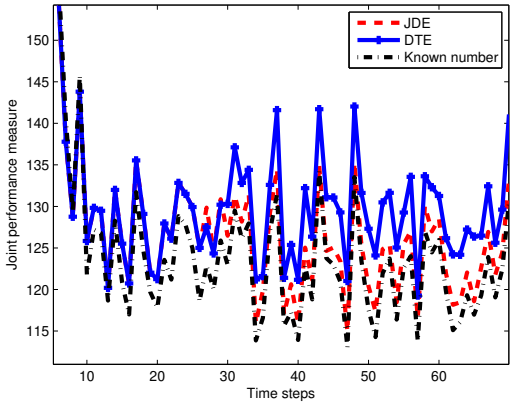



(c) Decision error rate

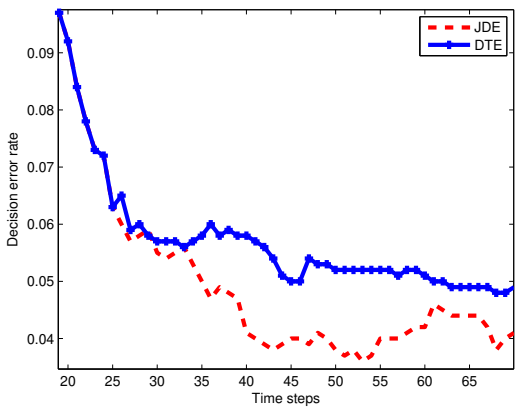
Figure 4.4: Simulation results of MTT, Case 2. This scenario differs from Case 1 only in increased $\lambda = 20/V$.




(a) OSPA

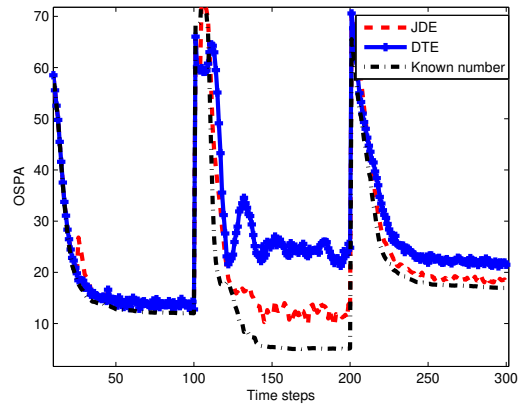



(b) JPM

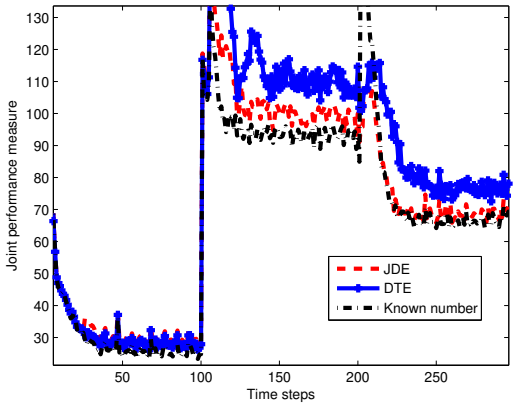



(c) Decision error rate

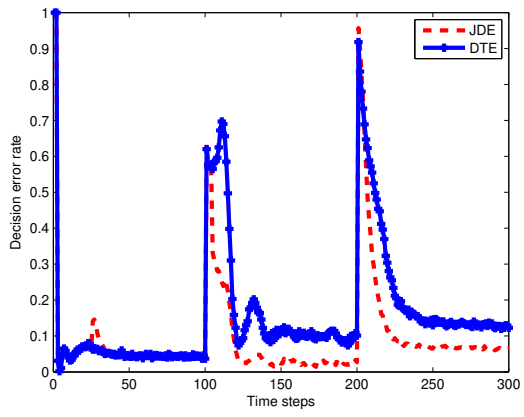
Figure 4.5: Simulation results of MTT, Case 3—tracking with low detection rate $P_d = 0.75$.




(a) OSPA

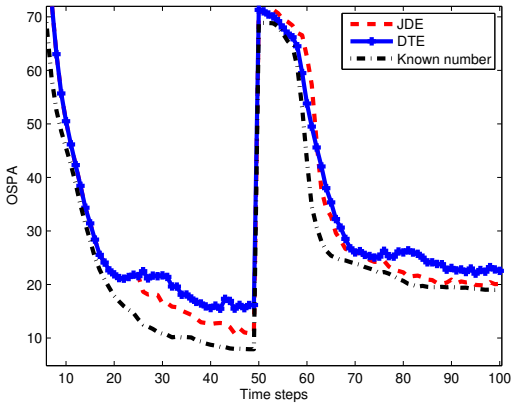



(b) JPM

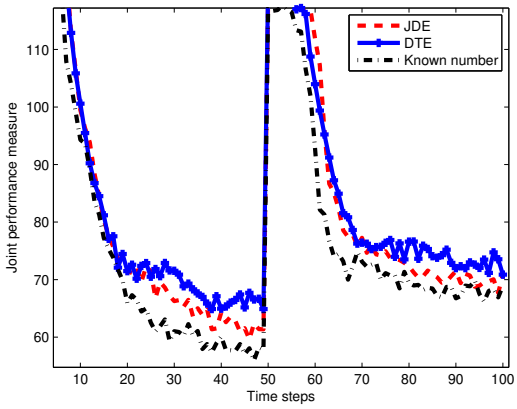



(c) Decision error rate

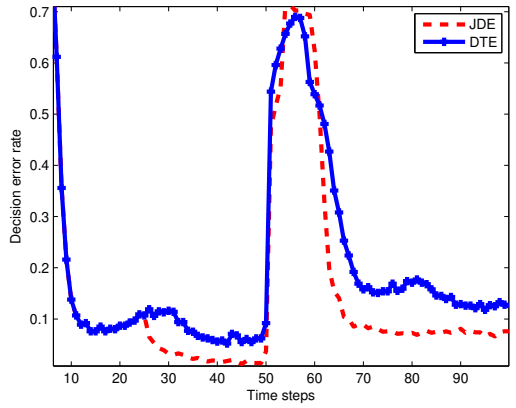
Figure 4.6: Simulation results of MTT, Case 4. Initially three targets are present in the surveillance region. Two targets disappear at time $k = 100$ and a new target appears at time $k = 200$.




(a) OSPA




(b) JPM




(c) Decision error rate

Figure 4.7: Simulation results of MTT, Case 5. The number of targets is randomly sampled on each run. One target disappears and a new target appears simultaneously at time 50.

# Chapter 5

# Performance Analysis of Sequential Probability Ratio Test

## 5.1 Introduction

This chapter studies the operating characteristic (OC) and average sample number (ASN) of the sequential probability ratio test (SPRT) and the average run length (ARL) of the cumulative sum test (CUSUM) [166, 169, 170].

The development of SPRT [243, 244] marked the birth of sequential analysis, a branch of statistics. SPRT is widely used in medicine, social science and engineering, such as clinical test, quality control, and radar signal processing. Many generalizations and modifications of SPRT (e.g., GSPRT and truncated SPRT) have been proposed [2,9,10,60,202,246] to improve the performance further for more complicated applications or to meet the requirements for more general settings than the simplest case of binary simple hypothesis testing with i.i.d. observations. Their behavior was studied in [60, 116]. Further, SPRT also forms the foundation of many sequential techniques, such as the celebrated Page's CUSUM test [198]. A comprehensive survey of sequential analysis as well as the challenges was provided by [127] and the subsequent extensive comments and discussions by the reviewers.

Theoretically speaking, SPRT requires neither the hypotheses to be simple nor the observations to be i.i.d., provided the LLRs can be computed sequentially. Further, the bounds for the test do not have to be constant over time. However, analysis of SPRT's properties and behavior becomes much more complicated without these assumptions.

The performance of SPRT has been studied extensively. Its optimality for binary simple hypothesis testing with i.i.d. observations was proved in [131,176,245]—the expected sample sizes under both hypotheses are simultaneously minimized among all the tests that do not exceed the given type I and type II error rates. This optimality is remarkable and rarely achievable elsewhere. But when it comes to the composite hypothesis problem, this "miracle" is gone and analysis of the optimality becomes much more difficult (see [116, 125, 127, 172] and the references therein). Also, the optimality properties of the SPRT without the i.i.d. assumption for observations have been studied, but only a few asymptotic results for some special cases are available (see [60, 93, 126, 165, 240]).

Two important functions—OC and ASN—characterize the behavior of SPRT. Unlike the optimality problem, the existing methods proposed to evaluate these two functions are almost all based on the assumption of an i.i.d. LLR sequence with constant bounds. In this dissertation, we consider the case of an independent but non-stationary LLR sequence $\langle s_t \rangle$, meaning that the LLRs need not be identically distributed. Two inductive integral equations governing the OC and ASN respectively for the non-stationary LLR sequence with time-varying bounds are obtained. They can be viewed as a generalization of the Fredholm integral equation of the second kind (FIESK). The governing equations are in an inductive form due to the loss of stationarity of the LLR sequence. Unfortunately, the uniqueness of solution can not be guaranteed in general, rendering numerical solution difficult. However, the theoretical value of these two governing equations can not be ignored since they form a foundation to analyze the OC and ASN and help understand the behavior of SPRT. For two frequently encountered special cases—constant bounds with (a) the LLR sequence converges in distribution or (b) it has periodic distributions—additional conditions (equations) can be

imposed, rendering a unique numerical solution, which can be easily obtained by solving a system of linear equations. As explained later, these two special cases are not uncommon in applications and their solutions are simple.

For computation of the OC and ASN for truncated SPRT (TSPRT), our methods may provide analytical solutions in general by backward induction. However, the convolutions involved in the induction process may become more and more difficult to evaluate analytically and thus impede analytical solutions. Hence, numerical methods are also provided. The fundamental ideas for these numerical methods are all similar—approximating the integral equation by a system of linear algebraic equations, which can be solved straightforwardly. Our method is also applicable to compute the average run length (ARL) of CUSUM [198], which is closely related to SPRT and they share many similarities in their performance analyses.

This chapter is organized as follows. SPRT is briefly described in Sec. 5.2. The two inductive integral equations governing the OC and ASN of SPRT with a non-stationary LLR sequence and time-varying bounds are derived in Sec. 5.3. Numerical solutions for two special cases are presented in Sec. 5.4. Application of our methods to TSPRT and CUSUM is elaborated in Sec. 5.5 and Sec. 5.6, respectively. Several illustrative examples are offered in Sec. 5.7 to demonstrate our methods and results. Summary is made in Sec. 5.8.

## 5.2 Overview of Sequential Probability Ratio Test

For a binary simple hypothesis testing problem,

$$H_0 : \ \theta = \theta_0 \quad \text{vs.} \quad H_1 : \ \theta = \theta_1$$

where $\theta$ is the parameter under test, if the observations $z_t$ are collected sequentially, then the SPRT computes the cumulative sum $S_t$ of the LLRs and the decision is made when data

are conclusive:

$$\begin{aligned} &\text{Declare } H_1 \text{ (or } H_0\text{) if } S_t \geq B_t \text{ (or } S_t \leq A_t\text{)} \\ &\text{Else, continue } S_{t+1} = S_t + s_{t+1} \end{aligned} \tag{5.1}$$

where $s_t = \ln \frac{f(z_t|H_1)}{f(z_t|H_0)}$ is the LLR at time $t$. $A_t$ and $B_t$ are the lower and upper bounds depending on the type I error probability $\alpha$ and type II error probability $\beta$. We explicitly consider time-varying bounds, indicated by the subscript $t$. It is assumed that $B_t > A_t, \forall t$. $f\left(z_t|H_i\right)$ is the likelihood of $H_i$.

If the LLR sequence $\langle s_t \rangle$ is i.i.d. (conditioned on true $\theta$) and the bounds are constant (i.e., $A_t = A$ and $B_t = B$), extensive results for the OC and ASN are available. Denote the probability density function (PDF) of $s_t$ as $f_\theta^t(x) = f_\theta(x)$, where the subscript $\theta$ denotes the ground truth of the underlying parameter, which need not be either $\theta_0$ or $\theta_1$. First, define

$$\tau_k = \min\left\{t : S_t \leq A_t \text{ or } S_t \geq B_t, t > k\right\} \tag{5.2}$$

as the stopping time of the test. The subscript $k$ denotes the start time of the test. That is, the test is initialized at time $k$ with initial value $S_k = s$, and $S_t$ (where $t > k$) are computed sequentially based on observations $z_{k+1}, z_{k+2}, \cdots$. Because of the stationarity of the i.i.d. LLR sequence, it does not matter when SPRT with constant bounds starts (i.e., its statistical properties do not change w.r.t. $k$). Without loss of generality, we assume the test starts at time $k = 0$. For this case, the OC is defined as the probability that the test statistic finally drops below $A$ as a function of the test initial value $s$ (i.e., $S_0 = s$) and the ground truth $\theta$:

$$P_\theta(s) \triangleq P_\theta\{S_{\tau_0} \leq A | S_0 = s\} = 1 - P_\theta\{S_{\tau_0} \geq B | S_0 = s\} \tag{5.3}$$

Notice that $P_\theta(\cdot)$ denotes the OC while $P_\theta\{\cdot\}$ denotes the probability of an event — the reader should be able to distinguish them from the context without ambiguity. The second equality of Eq. (5.3) holds if and only if the SPRT terminates in finite steps almost surely:

$$P_\theta\{\tau_0 < \infty\} = 1 \tag{5.4}$$

Wald [244] has proved it under very mild conditions for Wald's SPRT. One of the sufficient conditions for this case is [73]

$$P_\theta\{s_t = 0\} < 1, \forall t \tag{5.5}$$

Define $N_\theta(s) \triangleq E_\theta\left[\tau_0 \,|S_0 = s\right]$ as the average sample number (ASN) [244], which is also a function of $s$ and $\theta$. The existence of a finite ASN for Wald's SPRT was examined in [37,131]. Note that Eq. (5.4) is a necessary condition for the existence of a finite ASN.

It is known [25, 44, 198] that $P_\theta(s)$ and $N_\theta(s)$ satisfy the following Fredholm integral equations of the second kind (FIESK) [18, 203],

$$P_\theta\left(s\right) = F_\theta\left(A - s\right) + \int_A^B P_\theta\left(x\right) f_\theta\left(x - s\right) dx \tag{5.6}$$

$$N_\theta(s) = 1 + \int_A^B N_\theta\left(x\right) f_\theta\left(x - s\right) dx \tag{5.7}$$

where $F_\theta\left(\cdot\right)$ is the CDF of $s_t$. The existence and uniqueness of the solution for the general FIESK are guaranteed under mild conditions, given in [75, 97]. In general, one has to resort to numerical approximation to the solutions.

## 5.3 The OC and ASN with Independent but Non-stationary LLRs and Time-Varying Bounds

If the LLR sequence $\langle s_t \rangle$ is independent but not stationary and/or the bounds are time-varying, it does matter when the SPRT starts. So we define

$$\begin{aligned} P_\theta^k(s) &\triangleq P_\theta\{S_{\tau_k} \leq A_{\tau_k} | S_k = s\} \\ &= 1 - P_\theta\{S_{\tau_k} \geq B_{\tau_k} | S_k = s\} \end{aligned}$$

as the OC for this case with the superscript $k$ explicitly indicating the test start time. Note that the second equality is correct if SPRT terminates in finite steps almost surely:

$$P_\theta\{\tau_k < \infty\} = 1, \ \forall k \tag{5.8}$$

Similarly, Eq. (5.8) is a necessary condition for the existence of a finite ASN for this case. Unfortunately, for an independent but non-stationary LLR sequence, Eq. (5.5) is no longer sufficient. Evaluating the existence of a finite ASN requires knowledge of its distribution, which is not easy to gain in general. [120, 217] provided sufficient conditions for Eq. (5.8) and for the existence of a finite ASN respectively under very general settings, but the results are not easy to apply. In most practical problems, we believe the ASN should be finite. The case that the ASN diverges to infinity is beyond our consideration. Define

$$N_\theta^k(s) \triangleq E_\theta \left[ \tau_k - k \, | S_k = s \right]$$

as the ASN. Note that the start time $k$ is subtracted since we only consider how many future samples are needed on average.

Then $P_\theta^k(s)$ and $N_\theta^k(s)$ are governed by the following inductive integral equations,

$$P_\theta^k(s) = F_\theta^{k+1}(A_{k+1} - s) + \int_{A_{k+1}}^{B_{k+1}} P_\theta^{k+1}(x) f_\theta^{k+1}(x - s)\, dx \tag{5.9}$$

$$N_\theta^k(s) = 1 + \int_{A_{k+1}}^{B_{k+1}} N_\theta^{k+1}(x) f_\theta^{k+1}(x - s)\, dx \tag{5.10}$$

where $f_\theta^k(\cdot)$ and $F_\theta^k(\cdot)$ denote the PDF and CDF of $s_k$, respectively. The derivation follows the idea in [25, 44] and is given in Appendix B.

Unlike Eqs. (5.6)–(5.7), even the numerical solutions of Eqs. (5.9)–(5.10) are difficult to obtain (if not impossible) in general. There are two major difficulties. First, although these two equations are in an inductive form (w.r.t. time $k$), it is virtually impossible to implement the induction, be it forward or backward, since no initial value (this is actually exactly what we want to have) is available for the induction. Second, the solutions for $P_\theta^k(s)$ and $N_\theta^k(s)$ are not unique since clearly they depend on the distributions of the future LLRs from time $k+1$ on. Without specifying these distributions, Eq. (5.9) and (5.10) are under-determined and therefore solutions are not unique. It is clear that for cases with an i.i.d. LLR sequence and constant bounds, Eqs. (5.9)–(5.10) degenerate to Eqs. (5.6)–(5.7) since the test start time has no impact on OC and ASN. In the next section, we try to solve these two equations numerically for two special cases.

## 5.4 Numerical Solutions for Special Cases

In this section, numerical solutions for two special cases of SPRT are explored. If the bounds are constant and (a) the LLR sequence converges in distribution or (b) LLRs are periodically distributed, then unique solutions can be obtained by extending the SLAE method.

### 5.4.1 System of Linear Algebraic Equations (SLAE) Method

If the bounds are constant, the SLAE method can be employed, which was proposed [72] to solve Eqs. (5.6) and (5.7) numerically under the Gaussian assumption. It approximates the integral term by a Gaussian quadrature to form a system of linear equations. Numerical values of $P_\theta(s)$ and $N_\theta(s)$ on every quadrature point are calculated by solving this linear equation system. Although the SLAE method was developed under the Gaussian assumption, actually it is generally applicable provided the integral terms in Eqs. (5.6)–(5.7) can be well approximated by Gaussian quadrature. Further, the fact that many density functions can be well approximated by Gaussian mixtures of only a few components also broadens its range of applications. We modify the SLAE method and convert Eqs. (5.9) and (5.10) to systems of linear equations. Replacing integral terms with $n$-point Gaussian quadrature with weights $\omega_i$ and points $y_i$, we have the following for OC

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}_{k+1} & & \\ & \mathbf{I} & \mathbf{M}_{k+2} & \\ & & \ddots & \\ & & \mathbf{I} & \mathbf{M}_{k+m} \end{bmatrix} \begin{bmatrix} \mathbf{P}_k \\ \mathbf{P}_{k+1} \\ \vdots \\ \mathbf{P}_{k+m} \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_{k+1} \\ \mathbf{\Phi}_{k+2} \\ \vdots \\ \mathbf{\Phi}_{k+m} \end{bmatrix} \tag{5.11}$$

and for ASN

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}_{k+1} & & \\ & \mathbf{I} & \mathbf{M}_{k+2} & \\ & & \ddots & \\ & & \mathbf{I} & \mathbf{M}_{k+m} \end{bmatrix} \begin{bmatrix} \mathbf{N}_k \\ \mathbf{N}_{k+1} \\ \vdots \\ \mathbf{N}_{k+m} \end{bmatrix} = \begin{bmatrix} \mathbf{1}_n \\ \mathbf{1}_n \\ \vdots \\ \mathbf{1}_n \end{bmatrix} \tag{5.12}$$

where

$\mathbf{M}_{k+1}$ is an $n \times n$ matrix with the entry for $i$th row and $j$th column $\mathbf{M}_{k+1}^{(ij)} = -\omega_i f_\theta^{k+1}(y_i - y_j)$;

$m$ is a positive integer;

$\mathbf{1}_n = n \times 1$ vector of all 1's;

$\mathbf{I}$ is the $n \times n$ identity matrix;

$\mathbf{\Phi}_{k+1} = \left[F_\theta^{k+1}(A - y_1), \cdots, F_\theta^{k+1}(A - y_n)\right]^T$;

$\mathbf{P}_k = \left[P_\theta^k(y_1), P_\theta^k(y_2), \cdots, P_\theta^k(y_n)\right]^T$;

$\mathbf{N}_k = \left[N_\theta^k(y_1), N_\theta^k(y_2), \cdots, N_\theta^k(y_n)\right]^T$.

The derivation is given in Appendix C. However, Eqs. (5.11) and (5.12) are underdetermined since there are only $nm$ equations but $n(m+1)$ unknowns. Additional $n$ equations are needed to have unique solutions. The following two special cases are considered.

### 5.4.2 Case with Convergent LLR Sequence

First, we consider a special case that $\langle s_t \rangle$ converges to $\check{s}$ in distribution [223] and the PDF of $s_t$ are continuous. That is, $\lim_{t\to\infty} F_\theta^t(s) = \check{F}(s), \forall s$, where $\check{F}(s)$ is the CDF of $\check{s}$. In this case, $\langle s_t \rangle$ becomes i.i.d. asymptotically and thus there exists a positive integer $M$ such that

$$\left.\begin{aligned} P_\theta^k(s) \approx P_\theta^{k+1}(s) \approx \cdots \approx P_\theta^\infty(s) \\ N_\theta^k(s) \approx N_\theta^{k+1}(s) \approx \cdots \approx N_\theta^\infty(s) \end{aligned}\right\} \forall k > M$$

Hence

$$\left.\begin{aligned} \mathbf{P}_k \approx \mathbf{P}_{k+1} \approx \cdots \approx \hat{\mathbf{P}} \\ \mathbf{N}_k \approx \mathbf{N}_{k+1} \approx \cdots \approx \hat{\mathbf{N}} \end{aligned}\right\} \forall k > M \tag{5.13}$$

where $\hat{\mathbf{P}}$ and $\hat{\mathbf{N}}$ are the solutions of Eqs. (5.6)–(5.7) respectively with $f_\theta(s)$ replaced by the distribution of $\check{s}$. This can be intuitively understood: If $M$ is large enough, for $t > M$ the distributions of $s_t$ become almost idential (i.e., approximately i.i.d.), and then the impact of the test start time on the statistical properties of the SPRT can be ignored, which validates Eq. (5.13). Therefore, any existing techniques for the i.i.d. case can be applied to obtain $\hat{\mathbf{P}}$ and $\hat{\mathbf{N}}$. Then, we choose the $m$ in Eqs. (5.11)–(5.12) such that $k + m = M$. Plugging Eq.

(5.13) into Eqs. (5.11) and (5.12) respectively yields the following two $n\left(m+1\right)\times n\left(m+1\right)$ equation systems for OC

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}_{k+1} & & & \\ & \mathbf{I} & \mathbf{M}_{k+2} & & \\ & & \ddots & & \\ & & \mathbf{I} & \mathbf{M}_{k+m} & \\ & & & \mathbf{I} & \end{bmatrix} \begin{bmatrix} \mathbf{P}_k \\ \mathbf{P}_{k+1} \\ \vdots \\ \mathbf{P}_{k+m} \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_{k+1} \\ \mathbf{\Phi}_{k+2} \\ \vdots \\ \mathbf{\Phi}_{k+m} \\ \hat{\mathbf{P}} \end{bmatrix} \tag{5.14}$$

and for ASN

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}_{k+1} & & & \\ & \mathbf{I} & \mathbf{M}_{k+2} & & \\ & & \ddots & & \\ & & \mathbf{I} & \mathbf{M}_{k+m} & \\ & & & \mathbf{I} & \end{bmatrix} \begin{bmatrix} \mathbf{N}_k \\ \mathbf{N}_{k+1} \\ \vdots \\ \mathbf{N}_{k+m} \end{bmatrix} = \begin{bmatrix} \mathbf{1}_n \\ \mathbf{1}_n \\ \vdots \\ \mathbf{1}_n \\ \hat{\mathbf{N}} \end{bmatrix} \tag{5.15}$$

Note that the coefficient matrices in Eq. (5.14) and (5.15) are always invertible. The equation system can be solved by a linear-equation-system solver.

### 5.4.3 Case with LLRs of Periodic PDF

If the distributions of $s_t$ are periodic with period $T$ (a positive integer), i.e., $f_\theta^t(s) = f_\theta^{t+T}(s),\ \forall t$. Then,

$$P_\theta^k\left(s\right) = P_\theta^{k+T}\left(s\right), N_\theta^k\left(s\right) = N_\theta^{k+T}\left(s\right), \forall k$$

and hence

$$\mathbf{P}_k = \mathbf{P}_{k+T},\ \mathbf{N}_k = \mathbf{N}_{k+T}, \forall k \tag{5.16}$$

It is easily understood since the SPRT starting at time $k$ or $k+T$ are statistically equivalent, provided the initial values are the same. Inserting Eq. (5.16) into Eqs. (5.11) and (5.12) respectively yields the following two equation systems for OC

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}_{k+1} & & & \\ & \mathbf{I} & \mathbf{M}_{k+2} & & \\ & & \ddots & & \\ & & \mathbf{I} & \mathbf{M}_{k+T} \\ -\mathbf{I} & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}_k \\ \mathbf{P}_{k+1} \\ \vdots \\ \mathbf{P}_{k+T} \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_{k+1} \\ \mathbf{\Phi}_{k+2} \\ \vdots \\ \mathbf{\Phi}_{k+T} \\ \mathbf{0}_n \end{bmatrix} \tag{5.17}$$

and for ASN

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}_{k+1} & & & \\ & \mathbf{I} & \mathbf{M}_{k+2} & & \\ & & \ddots & & \\ & & \mathbf{I} & \mathbf{M}_{k+T} \\ -\mathbf{I} & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{N}_k \\ \mathbf{N}_{k+1} \\ \vdots \\ \mathbf{N}_{k+T} \end{bmatrix} = \begin{bmatrix} \mathbf{1}_n \\ \mathbf{1}_n \\ \vdots \\ \mathbf{1}_n \\ \mathbf{0}_n \end{bmatrix} \tag{5.18}$$

The solutions are readily obtained. It is clear that when $T = 1$, Eqs. (5.17)–(5.18) degenerate to the i.i.d. case.

## 5.5 Application to Truncated SPRT

In many applications, the sample size has to be limited. This leads to the truncated SPRT (TSPRT), in which the test is terminated (if the decision has not been made yet) after a maximum sample size is reached and a decision is made based on a pre-specified truncation rule $\Upsilon(\cdot)$. Compared with SPRT, this truncation will increase the actual $\alpha$ or $\beta$. One

complete cycle of TSPRT at time $t$ is:

$$\begin{aligned}
&\text{If } t < K, \quad S_t = S_{t-1} + s_t; \\
&\qquad \text{Declare } H_1 \text{ (or } H_0), \text{ If } S_t \geq B_t \text{ (or } S_t \leq A_t) \\
&\qquad \text{Else, } t = t + 1, \text{ continue.} \\
&\text{Else If } t = K, \quad S_K = \Upsilon(S_{t-1} + s_t); \\
&\qquad \text{Declare } H_1 \text{ (or } H_0), \text{ If } S_K = B_K \text{ (or } S_K = A_K)
\end{aligned}$$

where $K$ is the truncation time. The termination rule $\Upsilon(\cdot)$ can be a deterministic or random mapping of its input to $\{A_K, B_K\}$, that is, the outcome of $\Upsilon(\cdot)$ equals either $A_K$ or $B_K$. It is assumed that $B_t > A_t$ for $t = 1, 2, \cdots, K$. The stopping time $\tau_k$ for TSPRT is defined as

$$\tau_k \triangleq \min\{t : S_t \geq B_t \text{ or } S_t \leq A_t | t > k\}, \ k = 1, \cdots, K-1$$

and $\tau_K \triangleq K$. Recall that $k$ denotes the start time of the test. Since TSPRT terminates no later than $K$ (a finite time), $P\{\tau_k < +\infty\} = 1$ and $E[\tau_k] < +\infty$ is guaranteed.

Unlike the case of SPRT, the OC and ASN now are not governed by Eqs. (5.6) and (5.7) no matter whether $\langle s_t \rangle$ is i.i.d. or not. Since the SPRT is truncated at time $K$, the statistical properties of TSPRT with different start times are different even when observations are i.i.d. and the bounds are constant. This is similar to the case of SPRT with independent but non-stationary $\langle s_t \rangle$. Consequently, $P_\theta^k(s)$ and $N_\theta^k(s)$ are governed by similar inductive integral equations for OC

$$P_\theta^k(s) = F_\theta^{k+1}(A_{k+1} - s) + \int_{A_{k+1}}^{B_{k+1}} P_\theta^{k+1}(x) f_\theta^{k+1}(x - s)\, dx \tag{5.19}$$

$$P_\theta^K(s) = P_\theta\{\Upsilon(s) = A_K\} \tag{5.20}$$

and for ASN,

$$N_\theta^k(s) = 1 + \int_{A_{k+1}}^{B_{k+1}} N_\theta^{k+1}(x)\, f_\theta^{k+1}(x-s)\, dx \tag{5.21}$$

$$N_\theta^K(s) = 0; \tag{5.22}$$

where $k = 0, \cdots, K-1$. Note that unlike the SPRT (Eqs. (5.19) and (5.21)), Eqs. (5.20) and (5.22) are obtained directly from the truncation rule $\Upsilon(\cdot)$. The derivation in Appendix B is also valid for TSPRT.

### 5.5.1 Analytical Solutions

Theoretically speaking, the Eqs. (5.19)–(5.22) can be solved analytically by backward induction. Since for most applications $P_\theta^K(s)$ can be easily obtained by truncation rule $\Upsilon(\cdot)$ and $N_\theta^K(s) = 0$ is already known, the induction of Eqs. (5.19) and (5.21) can be done backwards from $k = K-1$ to 0, and exact solutions for $P_\theta^k(s)$ and $N_\theta^k(s)$ are obtained provided the convolution and the CDF involved can be evaluated analytically. An example is provided in Sec. 5.7.

### 5.5.2 Modified SLAE Method

When the convolution is hard to evaluate analytically, numerical solutions should be considered. If the bounds are constant, the modified SLAE method in Sec. 5.4 is directly

applicable. By incorporating Eqs. (5.20) and (5.22) into Eqs. (5.11) and (5.12), we have

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}_1 & & & & \\ & \mathbf{I} & \mathbf{M}_2 & & & \\ & & & \ddots & & \\ & & & & \mathbf{I} & \mathbf{M}_K \\ & & & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \vdots \\ \mathbf{P}_K \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_1 \\ \mathbf{\Phi}_2 \\ \vdots \\ \mathbf{\Phi}_K \\ \hat{\mathbf{P}}_K \end{bmatrix} \tag{5.23}$$

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}_1 & & & & \\ & \mathbf{I} & \mathbf{M}_2 & & & \\ & & & \ddots & & \\ & & & & \mathbf{I} & \mathbf{M}_K \\ & & & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{N}_0 \\ \mathbf{N}_1 \\ \vdots \\ \mathbf{N}_K \end{bmatrix} = \begin{bmatrix} \mathbf{1}_n \\ \mathbf{1}_n \\ \vdots \\ \mathbf{1}_n \\ \mathbf{0}_n \end{bmatrix} \tag{5.24}$$

for OC and ASN respectively, where

$$\hat{\mathbf{P}}_K \triangleq [P_\theta\{\Upsilon(y_1) = A_K\}, \cdots, P_\theta\{\Upsilon(y_n) = A_K\}]^T \tag{5.25}$$

Note that the matrices in Eqs. (5.23) and (5.24) are always invertible. Solutions are readily obtained by a linear-equation-system solver.

### 5.5.3 Finite Element Solutions

If the bounds $A_t$ or $B_t$ is time-varying, the modified SLAE method is difficult to apply since at each $k$ the Gaussian quadrature points $y_i$ are different, and consequently the linear equation systems (5.23) and (5.24) can not be obtained. In this case, a finite element analysis (FEA) can be adopted. Unlike the modified SLAE method, it approximates $P_\theta^k(s)$ and $N_\theta^k(s)$ by piecewise shape functions and evaluates the integral based on this approximation. We use

linear shape functions for simplicity. Let

$$A_m = \min A_i, \quad B_m = \max B_i, \quad i = 1, 2, \cdots, K$$

Divide the interval $[A_m,\ B_m]$ into $n_e$ segments with nodal points: $A_m = x_0 < x_1 < \cdots < x_{n_e} = B_m$. In general, a larger $n_e$ leads to a better approximation at the cost of more computation. Note that these segments do not necessarily have an equal length. But the maximum length should be small in order to be accurate. It is beneficial to choose the nodal points $x_e$ such that

$$A_i, B_i \in \{x_0, \cdots, x_{n_e}\}, \forall i \tag{5.26}$$

That is, the upper and lower bounds are both nodal points. As explained in Appendix D, this makes the integration easier. Denote the nodal values of $P_\theta^k(s)$ and $N_\theta^k(s)$ on $x_e$ as

$$U_e^k = P_\theta^k(x_e), \quad V_e^k = N_\theta^k(x_e), \quad e = 0, \cdots, n_e$$

Let $l_e = x_e - x_{e-1}$ be the length of the $e$th element. See Fig. 5.1 for illustration of these indices.

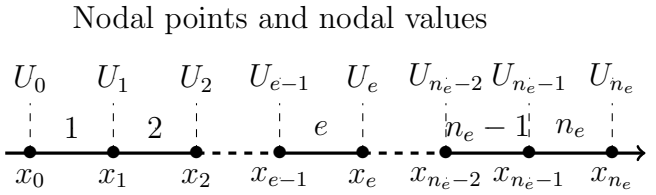



Figure 5.1: Finite element approximation. The interval $[A_m, B_m]$ is divided into $n_e$ segments with nodal points $x_e$ and nodal values $U_e = P_\theta(x_e)$. $P_\theta(s)$ is approximated by a piecewise linear function $P_\theta^{(e)}(s)$. On each subinterval $[x_{e-1}, x_e]$, $P_\theta^{(e)}(s)$ is a linear function of two terminal values $U_{e-1}$ and $U_e$, which need to be determined.

Then, $P_\theta^k(s)$ on interval $[x_{e-1}, x_e]$ can be approximated by a linear function

$$P_\theta^{k,(e)}(s) = \phi^{(e)}(s)\, U^{k,(e)}$$

where $U^{k,(e)} \triangleq [U_{e-1}^k, U_e^k]^T = [P_\theta^k(x_{e-1}), P_\theta^k(x_e)]^T$, and the shape function $\phi^{(e)}(x) = [\phi_1^{(e)}(x), \phi_2^{(e)}(x)]$ is defined on interval $[x_{e-1}, x_e]$:

$$\phi_1^{(e)}(x) \triangleq \begin{cases} 1 - \xi(x), & \xi(x) \in [0,1] \\ 0, & \text{otherwise} \end{cases} \tag{5.27}$$

$$\phi_2^{(e)}(x) \triangleq \begin{cases} \xi(x), & \xi(x) \in [0,1] \\ 0, & \text{otherwise} \end{cases} \tag{5.28}$$

$$\xi(x) = (x - x_{e-1})/l_e, \qquad x \in [x_{e-1}, x_e] \tag{5.29}$$

Graphically, $P_\theta^{k,(e)}(s)$ is simply a linear segment on interval $[x_{e-1}, x_e]$ with two terminal values specified by $U^{k,(e)}$, to be determined. Hence, $P_\theta^k(s)$ on interval $[A_m, B_m]$ can be approximated by a piecewise linear function:

$$P_\theta^k(s) \approx \sum_{e=1}^{n_e} P_\theta^{k,(e)}(s) \tag{5.30}$$

By replacing $P_\theta^k(s)$ in Eq. (5.19) with Eq. (5.30), calculation, and matrix manipulation, the following linear equation system can be derived for OC

$$\begin{bmatrix} \mathbf{I} & -\mathbf{Q}_1 & & & \\ & \mathbf{I} & -\mathbf{Q}_2 & & \\ & & \ddots & & \\ & & \mathbf{I} & -\mathbf{Q}_K & \\ & & & \mathbf{I} & \end{bmatrix} \begin{bmatrix} \mathbf{U}_0 \\ \mathbf{U}_1 \\ \vdots \\ \mathbf{U}_K \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_1 \\ \boldsymbol{F}_2 \\ \vdots \\ \boldsymbol{F}_K \\ \hat{\mathbf{U}}_K \end{bmatrix} \tag{5.31}$$

and for ASN

$$\begin{bmatrix} \mathbf{I} & -\mathbf{Q}_1 & & \\ & \mathbf{I} & -\mathbf{Q}_2 & \\ & & \ddots & \\ & & \mathbf{I} & -\mathbf{Q}_K \\ & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{V}_0 \\ \mathbf{V}_1 \\ \vdots \\ \mathbf{V}_K \end{bmatrix} = \begin{bmatrix} \mathbf{1}_{n_e+1} \\ \mathbf{1}_{n_e+1} \\ \vdots \\ \mathbf{1}_{n_e+1} \\ \mathbf{0}_{n_e+1} \end{bmatrix} \tag{5.32}$$

where

$\mathbf{Q}_k$ is given in Appendix D;

$\mathbf{U}_k = [U_0^k, U_1^k, \cdots, U_{n_e}^k]^T$;

$\mathbf{V}_k = [V_0^k, V_1^k, \cdots, V_{n_e}^k]^T$;

$\boldsymbol{F}_k = [F_\theta^k (A_k - x_0), \cdots, F_\theta^k (A_k - x_{n_e})]^T$;

$\hat{\mathbf{U}}_K$ is obtained by the truncation rule (5.20).

Although the idea is simple and the method is straightforward, the derivation is somewhat tedious, as given in Appendix D. The solutions for $\mathbf{U}_k$ and $\mathbf{V}_k$ can be obtained by a linear-equation-system solver. Note that the matrices in Eqs. (5.31) and (5.32) are always invertible. A numerical example is given in Sec. 5.7.

Compared with the modified SLAE method, the FEA can deal with the more general case. For example, the bounds can be time-varying, and the positions of the nodal points $x_i$ can be chosen by the user. This is desirable if the user has particular interests of OC and ASN on some special points. However, if the bounds are time invariant, the SLAE method is preferable due to its simplicity and efficiency, because $n$-point Gaussian quadrature approximates the integrand by a $(2n-1)$-degree polynomials, while the FEA (with a linear shape function) approximates the integrand by a piecewise linear function. Much more points are needed to achieve the same accuracy by FEA. Of course, the efficiency of FEA can be improved by choosing a higher order shape function, for example, a quadratic function or a Hermite cubic shape function [124]. But the derivation will become more complicated.

## 5.6 Application to CUSUM Test

CUSUM is a sequential test for the change detection problem. Consider a parameter-change detection problem. Assume a change from $H_0 : \theta = \theta_0$ to $H_1 : \theta = \theta_1$ is to be detected. CUSUM computes the cumulative sum $S_t$ and makes a decision when it crosses a pre-specified bound $B_t$, that is,

$$\begin{aligned} &\text{Declare the change if } S_t \geq B_t \\ &\text{Else, continue } S_{t+1} = \max\{A_t, S_t + s_{t+1}\} \end{aligned} \tag{5.33}$$

where $B_t$ is the test bound which depends on the false alarm level, and $A_t$ is the bound to re-initialize the test (usually $A_t = 0$ in practice). CUSUM may be viewed as a repeated SPRT; that is, when $S_t$ drops below the lower bound $A_t$, the SPRT is re-started until a change is detected (i.e., $S_t$ crosses $B_t$). Hence, their performance analyses are also tightly connected.

The performance of CUSUM is characterized by the average run length (ARL) function $L_\theta(s)$. If the bounds are constant (i.e., $B_t = B$ and $A_t = 0$) and $\langle s_t \rangle$ is i.i.d., $L_\theta(s)$ can be computed based on the OC and ASN of the corresponding SPRT, see [25, 170]. Also, it is known that $L_\theta(s)$ follows the following FIESK [198]

$$L_\theta(s) = 1 + L_\theta(0)F_\theta(-s) + \int_0^B L_\theta(x) f_\theta(x - s)dx \tag{5.34}$$

As mentioned in Sec. 5.1, many numerical methods have been proposed to solve this equation (see [25, 72, 73] and references therein).

When the LLR sequence is independent but not identically distributed or the bounds are time-varying, the statistical properties of CUSUM change with its start time because of the non-stationarity of the LLR sequence, similar to ASN for SPRT. Then, the computation of ARL based on OC and ASN of the corresponding SPRT, as in the i.i.d. case, is no longer

valid and hard to be extended to the non-stationary case. Besides, like OC and ASN, ARL does not satisfy an FIESK (Eq. (5.34)) any more in this case. Hence numerically solving Eq. (5.34) is not valid, either.

However, as mentioned before, ASN and ARL share many similarities because of the strong connection between CUSUM and SPRT. An inductive integral equation governing ARL can be derived, which can be solved numerically for the same special cases in Sec. 5.4. The stopping time of CUSUM is defined as

$$\tau_k = \min\{t : S_t \geq B_t, t > k\} \tag{5.35}$$

and the ARL $L_\theta^k(s)$ in this case is defined as

$$L_\theta^k(s) \triangleq E_\theta\left[\tau_k - k \,|S_k = s\right] \tag{5.36}$$

which explicitly takes the start time $k$ into consideration. Note that in Eq. (5.36), the start time $k$ is subtracted from $\tau_k$ since we only care how large the future sample size is needed on average. Then the governing equation for $L_\theta^k(s)$ is derived as

$$L_\theta^k(s) = 1 + \int_{A_{k+1}}^{B_{k+1}} L_\theta^{k+1}(x) f_\theta^{k+1}(x-s)\,dx + L_\theta^{k+1}(A_{k+1}) F_\theta^{k+1}(A_{k+1}-s) \tag{5.37}$$

The derivation is given in Appendix E and is based on the assumption that the ARL is finite, which should be the case for most practical applications.

When the bounds are constant (i.e., $A_t = A$ and $B_t = B$), the modified SLAE method in Sec. 5.4 can be applied in a similar manner and yields [170]

$$\begin{bmatrix} \mathbf{I} & \hat{\mathbf{M}}_{k+1} & & & & \\ & \mathbf{I} & \hat{\mathbf{M}}_{k+2} & & & \\ & & & \ddots & & \\ & & & & \mathbf{I} & \hat{\mathbf{M}}_{k+m} \end{bmatrix} \begin{bmatrix} \mathbf{L}_k \\ \mathbf{L}_{k+1} \\ \vdots \\ \mathbf{L}_{k+m} \end{bmatrix} = \begin{bmatrix} \mathbf{1}_n \\ \mathbf{1}_n \\ \vdots \\ \mathbf{1}_n \end{bmatrix}$$

where

$\hat{\mathbf{M}}_k = \mathbf{M}_k - [\mathbf{\Phi}_k \ \mathbf{0}]$;

$\mathbf{0}$ is the $n \times (n-1)$ zero matrix;

$\mathbf{L}_k = \left[L_\theta^k(y_1), L_\theta^k(y_2), \cdots, L_\theta^k(y_n)\right]^T$.

Clearly, there are $nm$ equations but $n(m+1)$ unknowns. So $n$ additional independent equations are needed to enable a unique solution, similar as the SPRT case. For the two cases in Sec. 5.4, unique solutions are enabled and can be solved similarly.

## 5.7 Illustrative Examples

Numerical examples are provided in this section to illustrate our methods for finding OC and ASN of SPRT and TSPRT and the ARL of CUSUM. Our numerical or analytical solutions are compared with the Monte Carlo (MC) simulation with $10,000$ runs. On each MC run, the LLR sequence $\langle s_t \rangle$ is generated based on the assumed distribution to implement the test. The results (OC, ASN or ARL) of the MC simulation are computed based on the results and run length of these tests.

## Example 1: OC and ASN of SPRT, with Converging $\langle s_t \rangle$

Assume $s_t$ have Gaussian distributions

$$f_\theta^t(s) = \mathcal{N}(s; \mu_t, \sigma^2), \quad \forall t \tag{5.38}$$

where $\mu_t = \theta + ce^{-t}$, $\theta$ is the ground truth of the underlying parameter, $c$ a constant, and $\sigma^2$ the variance. It is easy to verify that $\langle s_t \rangle$ converges in distribution: $\lim_{t\to+\infty} F_t(s) = \check{F}(s)$, where $\check{F}(s)$ is the Gaussian CDF with mean $\theta$ and variance $\sigma^2$. Two groups of parameters (in Table 5.1) were used in the simulation and the results were compared with those of our algorithm, given in Fig. 5.2. The constant bounds $A$ and $B$ were computed by Wald's approximation with type I and type II error rates both setting to 0.01. $m$ is the parameter in Eqs. (5.14)–(5.15) and $n$ is the number of Gaussian quadrature points.

It can be observed from Fig. 5.2 that $P_\theta^k(s)$ monotonically decreases w.r.t. $s$, as expected. This should happen by the definition of OC. Since the distribution of $s_t$ converges pretty fast (the term $ce^{-t}$ diminishes exponentially), the differences of the OC $P_\theta^k(s)$ curves for different start times $k$ are almost unobservable for $k > 10$. Likewise for ASN $N_\theta^k(s)$. Further, as the mean of $s_t$ is converging to $\theta$ exponentially, for $\theta > 0$ the mean of the cumulative sum $S_t$ is increasing, rendering $S_t$ less likely to drop below the lower bound $A$. For $\theta < 0$, the mean of $S_t$ will finally decrease, increasing the chance that $S_t$ drops below $A$. Hence, $P_\theta^k(s)$ for group one is much smaller than for group two. For $N_\theta^k(s)$, when the initial value $s$ of the test is close to the bounds, it is more likely that the test statistic crosses the bounds in fewer steps. It makes sense for the test to take more steps if the test starts around the middle of the two bounds.

Table 5.1: Parameters for Example 1

| | $\theta$ | $\sigma^2$ | $c$ | $A$ | $B$ | $m$ | $n$ |
|---|---|---|---|---|---|---|---|
| Group 1 | 1 | 9 | 10 | $-4.6$ | 4.6 | 10 | 20 |
| Group 2 | $-1$ | 9 | 10 | $-4.6$ | 4.6 | 10 | 20 |

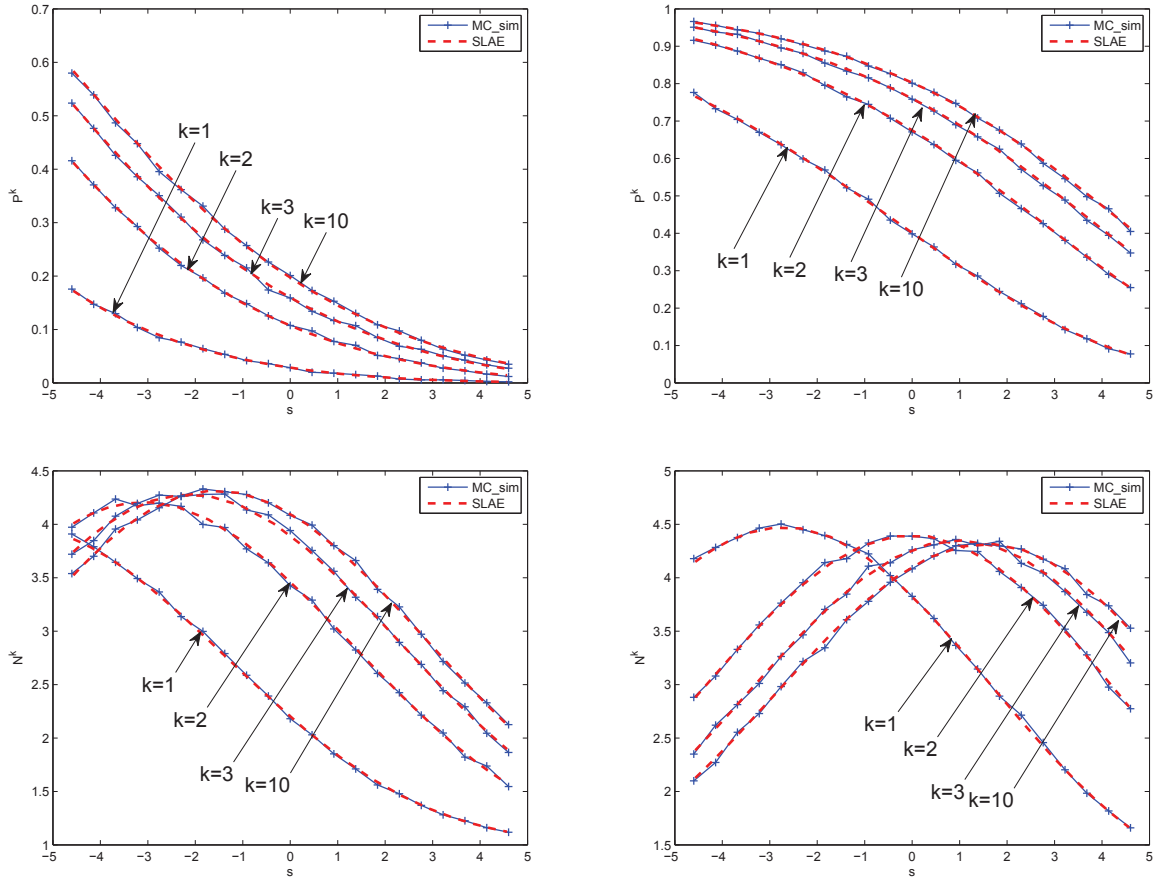



Figure 5.2: Results for Example 1. The modified SLAE solutions and the MC simulation results of $P_\theta^k(s)$ and $N_\theta^k(s)$ are compared. The left and right columns correspond to parameters in the first and the second groups in Table 5.1, respectively. The abscissa of each figure represents the initial value $s$ and the vertical axis is the OC $P_\theta^k(s)$ or ASN $N_\theta^k(s)$. The different curves in the figures correspond to the different start times $k$ of the test. From the plots, it is clear that the MC simulation results match the modified SLAE solutions very well.

## Example 2: ARL of CUSUM, with Convergent $\langle s_t \rangle$

Now we compute the ARL of CUSUM with $s_t$ distributed similarly as in Example 1 but with different bounds, as given in Table 5.2. Our numerical solutions and the results of MC simulation are given in Fig. 5.3. All the curves monotonically decrease w.r.t. the initial value $s$. This makes sense since the larger the initial value $s$, the more likely the cumulative sum $S_t$ will cross the upper bound $B$ in fewer steps. It can be seen by comparing the two figures in Fig. 5.3 that the ARLs for the first group of parameters are significantly smaller than for the second group. Since the mean of $s_t$ approaches $\theta$ fast, if $\theta > 0$, then the mean of $S_t$ will increase eventually, making the test more likely to terminate with a small ARL. If $\theta < 0$, the mean of $S_t$ is decreasing, rendering $S_t$ more likely to drop below $A$ and hence the test is restarted. This makes the test last longer. Further, since $\langle s_t \rangle$ approaches an i.i.d. process as $k$ increases, the curves can not be distinguished when $k > 10$ in this example. Finally, it is evident that our solutions agree well with the results of MC simulation.

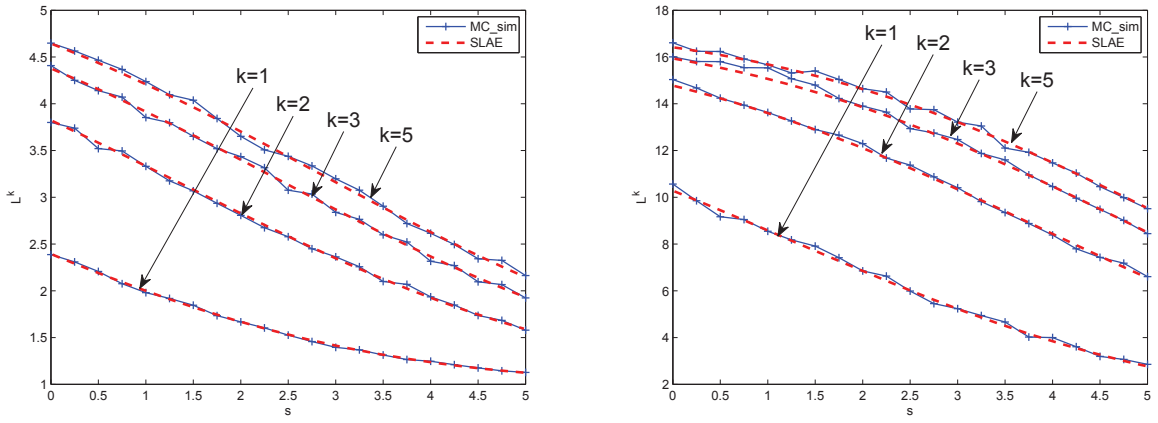



Figure 5.3: Results for Example 2, where the modified SLAE solutions and MC simulation results of the $L_\theta^k(s)$ are compared. The left and right figures correspond to parameters of the first and second groups in Table 5.2, respectively. The start time $k$ of CUSUM is indicated. Since the distributions of $s_t$ converge exponentially, the curves also converge and are almost indistinguishable when $k > 10$.

Table 5.2: Parameters for Example 2

| | $\theta$ | $\sigma^2$ | $c$ | $A$ | $B$ | $n$ | $m$ |
|---|---|---|---|---|---|---|---|
| Group 1 | 1 | 9 | 10 | 0 | 5 | 20 | 10 |
| Group 2 | $-1$ | 9 | 10 | 0 | 5 | 20 | 10 |

## Example 3: OC and ASN of SPRT, with Periodically Distributed $s_t$

Again, $s_t$ is assumed of the Gaussian distribution (5.38), but the mean $\mu_t$ is changing periodically: $\mu_t = \theta + \cos(\frac{2\pi t}{T})$, where the period $T = 9$. It is obvious that $f_\theta^t(s) = f_\theta^{t+T}(s)$. Two groups of parameters, given in Table 5.3, were simulated by the MC method and the results were compared with our solutions. The results are plotted in Fig. 5.4. The differences between the MC results and our numerical solutions are tiny. Some patterns of the curves—e.g., $P_\theta^k(s)$ monotonically decreases and $N_\theta^k(s)$ has its peak value in the middle of $A$ and $B$—are similar as in Example 1 for the same reasons. But in this case, the curve for $k$ and $k + T$ are exactly overlapped, meaning that there are only $T$ different curves.

Table 5.3: Parameters for Example 3

| | $\theta$ | $\sigma^2$ | $T$ | $A$ | $B$ | $n$ |
|---|---|---|---|---|---|---|
| Group 1 | 0.6 | 4 | 9 | $-4.6$ | 4.6 | 20 |
| Group 2 | $-0.6$ | 4 | 9 | $-4.6$ | 4.6 | 20 |

## Example 4: ARL of CUSUM, with Periodically Distributed $s_t$

In this example, the ARL of CUSUM is computed. Assume the $s_t$ has a similar distribution as in Example 3. Two groups of parameters, as given in Table. 5.4, are simulated. The numerical solutions and results of MC simulation are given in Fig. 5.5.

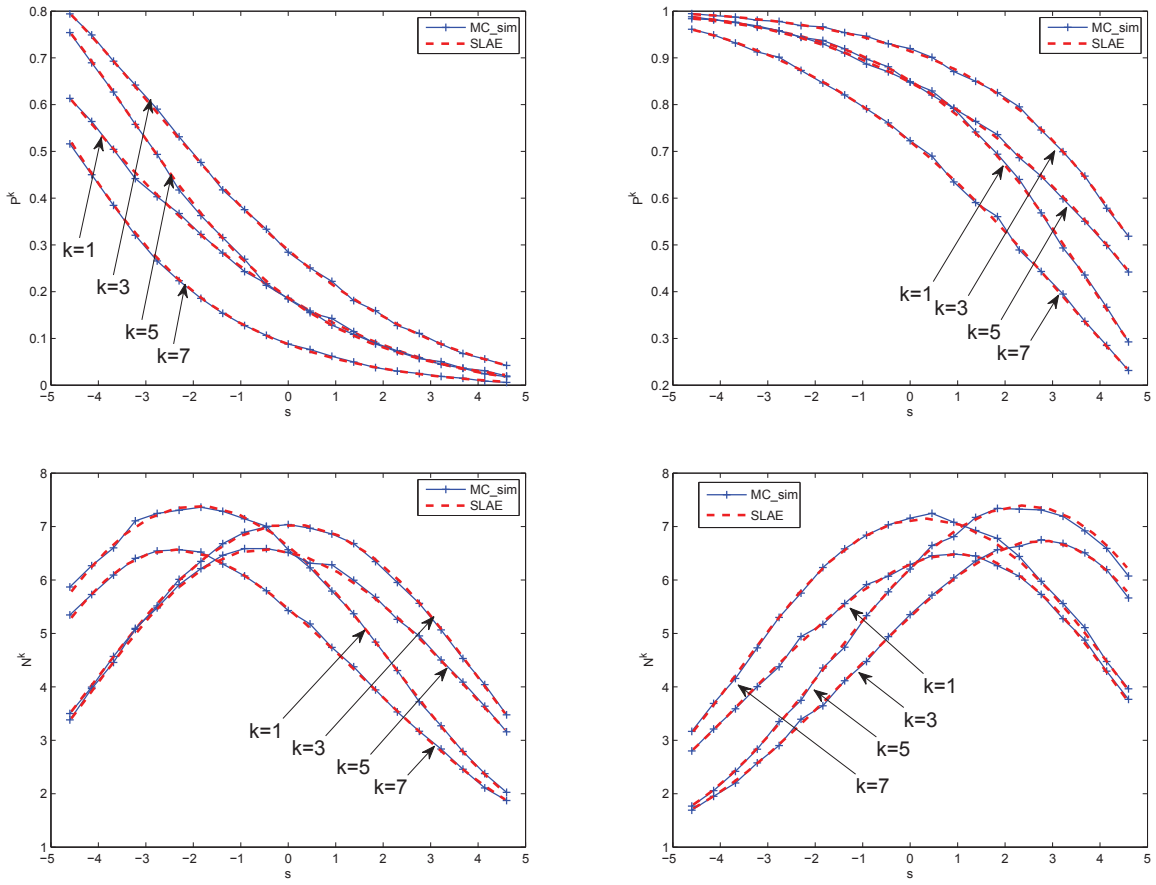



Figure 5.4: Results for Example 3. The left and right columns correspond to parameters in the first and the second groups in Table 5.3, respectively. Again, the MC simulation verifies our methods.

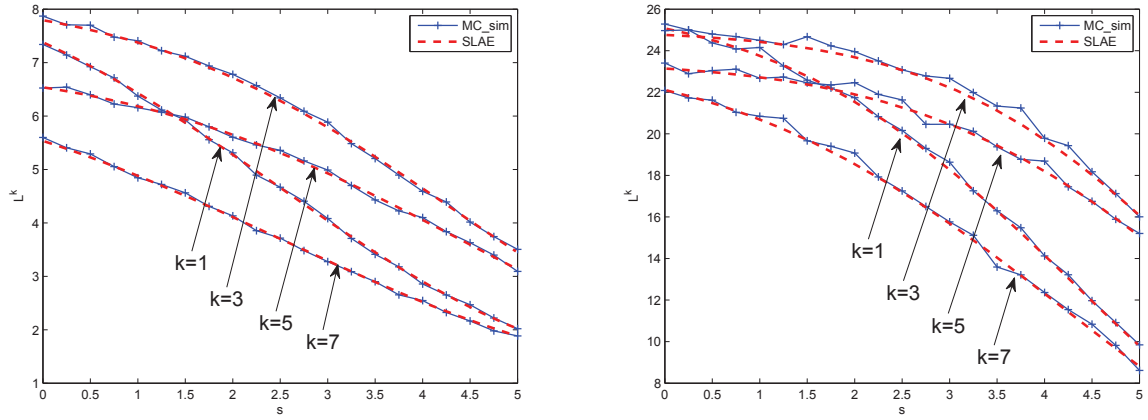



Figure 5.5: Results for Example 4. The left and right figures correspond to parameters of the first and second groups in Table 5.4, respectively. In this case, there are only $T$ different curves since LLRs have periodic distributions.

Table 5.4: Parameters for Example 4

| | $\theta$ | $\sigma^2$ | $T$ | $A$ | $B$ | $n$ |
|---|---|---|---|---|---|---|
| Group 1 | 0.6 | 4 | 9 | 0 | 5 | 20 |
| Group 2 | $-0.6$ | 4 | 9 | 0 | 5 | 20 |

## Example 5: Analytical Solutions for ASN of TSPRT

Consider the following binary hypothesis testing problem of the parameter $\theta$:

$$H_0 : \theta = \theta_0 = 1, \quad H_1 : \theta = \theta_1 = 0.01$$

Assume $z_t$ are i.i.d. and exponentially distributed

$$f_\theta^t(x) = f_\theta(x) = \theta e^{-\theta x}, \quad \theta, x > 0 \tag{5.39}$$

Note that the true $\theta$ need not be $\theta_0$ or $\theta_1$. The type I and type II error rates are set to be $\alpha = \beta = 0.1$ and constant bounds $A$ and $B$ are used and calculated by Wald's approximation [244]. The test is truncated at $K = 10$ and we want to evaluate the ASN of TSPRT. The TSPRT compares $S_t$ with the bounds $A$ and $B$, which is equivalent to testing

$$s + \sum_{i=1}^{t} z_i \leq A_t \quad \text{or} \quad s + \sum_{i=1}^{t} z_i \geq B_t$$

where (since $\theta_0 > \theta_1 > 0$)

$$A_t = \frac{A}{\theta_0 - \theta_1} + t\mu, B_t = \frac{B}{\theta_0 - \theta_1} + t\mu, \mu = \frac{\ln \theta_0 - \ln \theta_1}{\theta_0 - \theta_1}$$

By replacing $f_\theta^k(\cdot)$ in Eq. (5.21) with the PDF (5.39) of observation $z_k$ and backward induction of Eq. (5.21), the following solutions for ASN are obtained

$$\begin{cases} N_\theta^K(s) = 0,\ N_\theta^{K-1}(s) = 1 \\ N_\theta^{K-i}(s) = 1 + \sum_{j=1}^{i-1} a_j^{K-i}(s), \quad i = 2, \cdots, K \end{cases} \tag{5.40}$$

where $a_j^{K-i}(s)$ is given as below [169]

$$\begin{aligned} a_j^{K-i}(s) =& \int_{A_{K-i+1}}^{B_{K-i+1}} \cdots \int_{A_{K-j-1}}^{B_{K-j-1}} \int_{A_{K-j}}^{B_{K-j}} f_\theta^{K-j}\left(x_{K-j} - x_{K-j-1}\right) \cdots f_\theta^{K-i+2}\left(x_{K-i+2} - x_{K-i+1}\right) \\ & \times f_\theta^{K-i+1}\left(x_{K-i+1} - s\right) dx_{K-j} dx_{K-j-1} \cdots dx_{K-i+1}, \quad 2 \le i \le K \\ a_j^0(s) =& \int_{A_1}^{B_1} \cdots \int_{A_{K-j-1}}^{B_{K-j-1}} \int_{A_{K-j}}^{B_{K-j}} f_\theta\left(x_{K-j} - x_{K-j-1}\right) \cdots f_\theta\left(x_2 - x_1\right) \\ & \times f_\theta\left(x_1 - s\right) dx_{K-j} dx_{K-j-1} \cdots dx_1 \\ =& \begin{cases} \int_{\max\{A_1,s\}}^{B_1} \cdots \int_{A_{K-j-1}}^{B_{K-j-1}} \int_{A_{K-j}}^{B_{K-j}} \theta^{K-j} e^{-\theta(x_{K-j}-s)} dx_{K-j} dx_{K-j-1} \cdots dx_1, & s < B_1 \\ 0, & s \ge B_1 \end{cases} \\ =& \begin{cases} \theta(B_1 - \max\{A_1, s\})(\theta d)^{K-j-2} e^{\theta s}(e^{-\theta A_{K-j}} - e^{-\theta B_{K-j}}), & j = 1, \cdots, K-2,\ s < B_1 \\ e^{\theta s}(e^{-\theta \max\{A_1,s\}} - e^{-\theta B_1}), & j = K-1,\ s < B_1 \\ 0, & s \ge B_1 \end{cases} \\ d =& B_k - A_k = \frac{B - A}{\theta_0 - \theta_1} \end{aligned}$$

The analytical solution of $N_\theta^0(s)$ and results of MC simulation are plotted in Fig. 5.6. The OC can be computed similarly, although more tediously.

## Example 6: Finite Element Solutions for ASN and OC of TSPRT with Time-Varying Bounds

This example computes the OC and ASN of TSPRT when the LLR sequence $\langle s_t \rangle$ is i.i.d. and Gaussian distributed with mean $\mu = 1$ and variance $\sigma^2 = 4$ (conditioned on the true $\theta$).

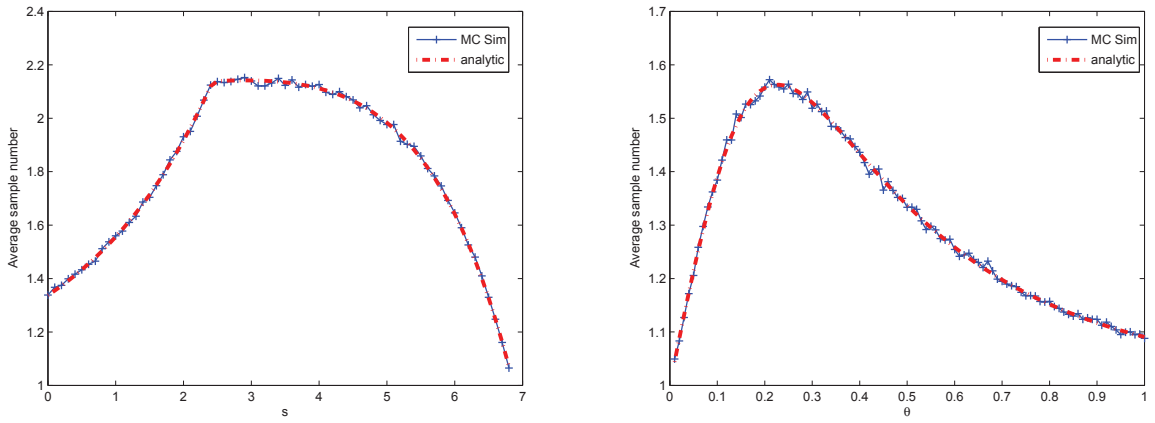



Figure 5.6: Analytical solutions and MC simulation results of the ASN for Example 5. The left figure shows ASN $N_\theta^0(s)$ vs. initial value $s$ for $\theta = 0.5$. The right figure shows $N_\theta^0(0)$ vs. $\theta$.

The test is terminated at time $K = 20$ with the truncation rule

$$\Upsilon(x) = \begin{cases} A_K \text{ with probability } \frac{B_K - x}{B_K - A_K} \\ B_K \text{ with probability } \frac{x - A_K}{B_K - A_K} \end{cases}, \quad A_K < x < B_k$$

That is, the probability that the test is truncated as an acceptance test is proportional to the distance between the test statistic at time $K$ and the upper bound $B_K$. Hence, $\mathbf{U}_K$ of (5.31) can be calculated readily from this truncation rule (see Eq. (5.20)). The bounds are time-varying and set to be $A_t = A + 0.1t$ and $B_t = B - 0.1t$ (with $-A = B = 5$, $t = 1, 2, \cdots, K$). The nodal points $x_e$ are spread on interval $[A, B]$ with an equal distance and $n_e = 100$. It is clear that $A_t$ and $B_t$ are all included in the set $\{x_e\}_{e=0}^{n_e}$. This choice is beneficial to the integration since a segment $[x_e, x_{e+1}]$ lays either entirely inside or entirely outside the interval of integration (see Appendix D). The FEA solutions are given in Fig. 5.7 and compared with the results of MC simulation. The exact solutions are not given due to the difficulty of evaluating the convolution analytically.

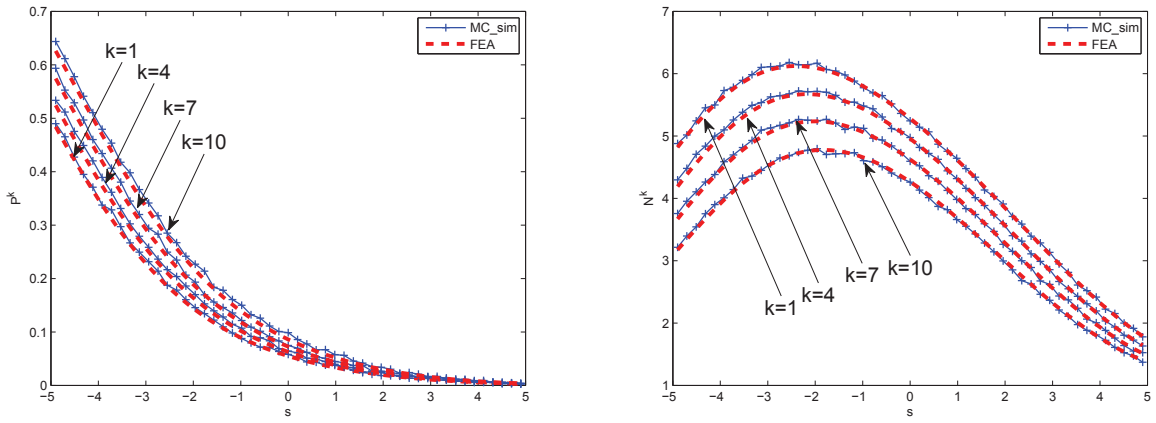



Figure 5.7: Results of FEA and MC simulation for Example 6. The left figure shows $P_\theta^k$ vs. the initial value $s$, and the right figure shows $N_\theta^k$. Different curves are for different start time $k$.

## 5.8 Summary

In this chapter we have developed two inductive equations governing the OC and ASN of SPRT, respectively, when the LLR sequence is independent but non-stationary and the bounds are time-varying. They can be viewed as a generalization of the Fredholm integral equation of the second kind for the case with i.i.d. LLR sequence and constant bounds. Numerical algorithms for two special cases, (a) the LLR sequence converges in distribution and (b) LLRs are periodically distributed, have been obtained by the modified SLAE method. Our methods have also been applied to compute the OC and ASN of truncated SPRT, which may lead to analytical solutions if the convolution involved can be evaluated. In addition to the SLAE method, a finite element analysis has been applied to obtain the numerical solutions for the TSPRT in the general case. Further, takeing advantage of the tight connection between SPRT and CUSUM, our methods have also been applied to computing the ARL function of CUSUM. An inductive integral equation governing the ARL has been derived and numerical solutions explored for the same two special cases for SPRT. Several numerical examples are provided and compared with the results of MC simulation to demonstrate our methods.

# Chapter 6

# Conclusion and Future Work

GLMMSE estimation and RJDE have been proposed for nonlinear point estimation and JDE problems, respectively. They are demonstrated by applications to target tracking. Further, the performance for SPRT and CUSUM with independent but non-stationary LLR sequence are analyzed in this dissertation.

GLMMSE estimation generalizes LMMSE estimation by employing a vector-valued MTF and finds the best estimator among all estimators that are linear in MTF, rather in the measurement itself (as LMMSE estimation does). Hence, the MTF introduces the flexibility of choosing different kinds of estimators for different problems. GLMMSE estimation with a proper MTF can be superior to LMMSE estimation in performance. Design guidelines for MTF have been studied to facilitate the design process. GLMMSE estimation has been applied to radar tracking, space-object tracking and multi-target tracking. Further applications will be exploited. We intend to develop GLMMSE estimation toward a standard tool for nonlinear point estimation, readily to be implemented when a problem is at hand. One major competitor to GLMMSE estimation is the density based methods, e.g., particle filter. Although very powerful, density estimation may be a overkill. As such, GLMMSE has the potential to outperform density estimation for nonlinear point estimation problem. Further study on the design of MTF is also needed. Although a systematic design procedure for

general applications may be difficult to come up with, good MTF may be obtained for some specific applications, e.g., radar/sonar measurements and angle/range only tracking, which are frequently encountered in tracking applications.

The RJDE algorithm has been developed for dynamic JDE problems. The inter-dependence between decision and estimation is accounted for by iterations of estimation-step and decision-step, which make the generalized Bayes risk non-increasing during the iteration. RJDE has been applied to joint tracking and classification and joint detection and tracking. Future work includes further improving the computational efficiency for applications with large data set, e.g., data fusion. Finding guidelines for determining the design parameters in realistic applications are also desired. Further, evaluating the performance gap between the batch JDE and RJDE may provide insight and feedback to improve our RJDE procedure. Additional analysis of our joint performance measure for JDE algorithm are also important and worth more effort.

A performance analysis for SPRT and CUSUM has been also studied. Two inductive equations governing the OC and ASN functions, respectively, of SPRT have been developed for independent but non-stationary LLR sequence. They can be viewed as a generalization of the Fredholm integral equations of the second kind for the i.i.d. case. Numerical algorithms have been obtained for two special cases: the LLR sequence converges in distribution or it has periodic distributions. Our methods can be readily applied to the truncated SPRT and the ARL of CUSUM. Identifying more cases that have unique solutions of the inductive equations are under further investigation. Some lower- or upper-bounds of these functions for the general cases may be obtained by our methods if the numerical solutions are difficult to have. Basically, the performance of a sequential test can be approximated by its truncated version. Intuitively, the impact of the truncation on the performance should diminish as the truncation time increases. The bounds of the approximation errors may be obtained based on the inductive equations we developed. Future work also includes extending our analysis to other sequential tests that are more sophisticated than SPRT or CUSUM (e.g., Multi-

hypothesis sequential test and two-sided CUSUM), and consequently providing a more solid alternative to Monte Carlo simulation for performance evaluation for sequential tests.

# Appendix A

# Joint Probability Data Association Filter

Assume there are $m_k^t$ (known) targets present in the surveillance region and $m_k^z$ measurements are received at scan $k$. The posterior probability of each feasible association event $\theta_k^l$ (conditioned on Assumptions (2) in Sec. 4.6) is computed:

$$\begin{aligned} P\{\theta_k^l | z_k, Z^{k-1}\} &= P\{\theta_k^l | z_k, m_k^z, Z^{k-1}\} \\ &= \frac{1}{c} f(z_k | \theta_k^l, m_k^z, Z^{k-1}) P\{\theta_k^l | m_k^z\} \end{aligned}$$

Assuming conditional independence of each measurement yields

$$f\{z_k | \theta_k^l, m_k^z, Z^{k-1}\} = \prod_{j=1}^{m_k^z} f(z_k^j | \theta_k^l, m_k^z, Z^{k-1})$$

and

$$f(z_k^j | \theta_k^l, m_k^z, Z^{k-1}) = \begin{cases} f_t^i(z_k^j), \text{ if } z_k^j \text{ is associated to target } i \text{ in } \theta_k^l \\ f_f(z_k^l), \text{ if } z_k^j \text{ is not from any target in } \theta_k^l \end{cases}$$

where $f_t^i(\cdot)$ and $f_f(\cdot)$ are the probability density functions of the true and false measurements. Given $\theta_k^l$, the correspondences between the targets and measurements as well as the number of true measurements $n_k^t$ are all known. Given $m_k^z$ and $n_k^t$, there are $m_k^z!/(m_k^z - n_k^t)!$ different $\theta_k^l$, which are assumed equally probable. Then $P\{\theta_k^l | m_k^z\}$ can be derived by

$$P\{\theta_k^l | m_k^z\} = P\{\theta_k^l, n_k^t | m_k^z\} = P\{\theta_k^l | n_k^t, m_k^z\} P\{n_k^t | m_k^z\}$$

where

$$P\{\theta_k^l | n_k^t, m_k^z\} = \frac{(m_k^z - n_k^t)!}{m_k^z!}$$
$$P\{n_k^t | m_k^z\} = (P_d P_G)^{n_k^t} (1 - P_d P_G)^{m_k^t - n_k^t} P_f\{m_k^z - n_k^t\}$$

and $P_f\{\cdot\}$ is the probability mass function of the number of false measurements. $P_d$ and $P_G$ are the detection and gate probabilities, respectively. A validation gate is not necessary but could increase the computational efficiency. If no gate is used, $P_G = 1$. Once all $P\{\theta_k^l | Z_k\}$ are computed, the probability $\mu_k^{ij}$ of associating $z_k^j$ to target $i$ is obtained by summing up all the probabilities of $\theta_k^l$ that contains this association. Then the track of target $i$ can be updated by

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k \bar{\tilde{z}}_k \tag{A.1}$$

$$P_k = P_k^0 \mu_k^{i0} + (1 - \mu_k^{i0}) P_k^{KF} + \tilde{P}_k \tag{A.2}$$

where $\mu_k^{i0} = 1 - \sum_{j=1}^{m_k^z} \mu_k^{ij}$ is the probability that no measurement is associated to the target, $\bar{\tilde{z}}_k = \sum_{j=1}^{m_k^z} \tilde{z}_k^j \mu_k^{ij}$ is the average measurement residual, and

$$
\begin{aligned}
P_k^{KF} &= P_{k|k-1} - K_k S_k^{-1} K_k' \\
\tilde{P}_k &= K_k [\sum_{j=1}^{m_k^z} \tilde{z}_k^j (\tilde{z}_k^j)' \mu_k^{ij} - (1 + \mu_k^{i0}) \bar{\tilde{z}}_k \bar{\tilde{z}}_k'] K_k' \\
P_k^0 &= P_{k|k-1} + \frac{1-\beta}{1 - P_d P_G} K_k S_k^{-1} K_k'
\end{aligned}
$$

$K_k$ is the KF gain at time $k$, $\beta = \frac{\Gamma_{\gamma/2}(n_z/2+1)}{n_z/2\Gamma_{\gamma/2}(n_z/2)}$, and $\Gamma$ is the incomplete Gamma function. The derivation for $P_k^0$—the MSE matrix for the case that none of the measurements is associated to the target—is given in [141]. A simple track management scheme is included in JPDAF, that is, if a track has $n$ ($n = 4$ in the simulation) successive steps that do not contain any validated measurement in a gating process, the track is deleted and re-initialized so that the number of tracks in JPDAF remains constant.

# Appendix B

# Derivation of Eqs. (5.9) and (5.10)

Since by definition $P_\theta^k(s)$ is the probability of the event that the test statistic crosses the lower bound with the start time $k$ and initial value $s$, this event can be partitioned into two mutually exclusive events:

$$\Lambda_1 = \{S_{k+1} = S_k + s_{k+1} = s + s_{k+1} \leqslant A_{k+1}\}$$
$$\Lambda_2 = \{A_{k+1} < S_{k+1} < B_{k+1}\} \cap \left\{S_{\tau_{k+1}} \leq A_{\tau_{k+1}} | S_{k+1}\right\}$$

For $\Lambda_1$, the test statistic crosses $A_{k+1}$ at time $k+1$, while for $\Lambda_2$, $S_{k+1}$ is between the two bounds (i.e., the SPRT does not stop at time $k+1$) and crosses the lower bound after time $k+1$. $\Lambda_2$ can be viewed as all events in which the test starts at time $k+1$ with the initial value of $S_{k+1}$. It is clear that the probability of $\Lambda_1$ equals

$$P_\theta\{\Lambda_1\} = \int_{-\infty}^{A_{k+1}-s} f_\theta^{k+1}(x)\, dx$$

The probability of $\Lambda_2$ equals

$$\begin{aligned} P_\theta\{\Lambda_2\} &= \int_{A_{k+1}}^{B_{k+1}} P_\theta\{\Lambda_2 | S_{k+1} = x\} P_\theta\{S_{k+1} = x\} dx \\ &= \int_{A_{k+1}}^{B_{k+1}} P_\theta\left\{S_{\tau_{k+1}} \leq A | S_{k+1} = x\right\} f_\theta^{k+1}(x-s) dx \\ &= \int_{A_{k+1}}^{B_{k+1}} P_\theta^{k+1}(x) f_\theta^{k+1}(x-s) dx \end{aligned}$$

Hence, Eq. (5.9) follows by $P_\theta\{\Lambda_1\} + P_\theta\{\Lambda_2\}$.

The derivation for $N_\theta^k(s)$ follows similarly. First, define two mutually exclusive events for the SPRT with the start time $k$

$$\begin{aligned} \Gamma_1 &= \{S_{k+1} \leqslant A_{k+1}\} \cup \{S_{k+1} \geqslant B_{k+1}\} \\ \Gamma_2 &= \{A_{k+1} < S_{k+1} < B_{k+1}\} \end{aligned}$$

It is clear that

$$\begin{aligned} P_\theta\{\Gamma_1\} &= 1 - P_\theta\{\Gamma_2\} \\ P_\theta\{\Gamma_2\} &= \int_{A_{k+1}}^{B_{k+1}} f_\theta^{k+1}(x-s) dx \end{aligned}$$

Denote by $N_\theta^k(s|\Gamma_1)$ and $N_\theta^k(s|\Gamma_2)$ the ASN conditioned on events $\Gamma_1$ and $\Gamma_2$, respectively. Obviously, conditioned on $\Gamma_1$, SPRT only needs one observation, and thus, $N_\theta^k(s|\Gamma_1) = 1$. Conditioned on $\Gamma_2$, after one observation, the test continues. It is equivalent to view it as the case where the test restarts at time $k+1$ with the initial value $S_{k+1}$ after one observation. This re-initialized test has ASN $N_\theta^{k+1}(x)$ if $S_{k+1} = x \in (A_{k+1}, B_{k+1})$. Hence, we have

$$N_\theta^k(s|\Gamma_2) = 1 + \int_{A_{k+1}}^{B_{k+1}} N_\theta^{k+1}(x) P_\theta\{S_{k+1} = x|\Gamma_2\} dx$$

Hence,

$$
\begin{aligned}
N_\theta^k(s) &= N_\theta^k(s|\Gamma_1)P_\theta\{\Gamma_1\} + N_\theta^k(s|\Gamma_2)P_\theta\{\Gamma_2\} \\
&= 1\cdot(1-P_\theta\{\Gamma_2\}) + (1+\frac{\int_{A_{k+1}}^{B_{k+1}} N_\theta^{k+1}(x)\,P_\theta\{S_{k+1}=x,\Gamma_2\}\,dx}{P_\theta\{\Gamma_2\}})P_\theta\{\Gamma_2\} \\
&= 1+\int_{A_{k+1}}^{B_{k+1}} N_\theta^{k+1}(x)\,f_\theta^{k+1}(x-s)\,dx
\end{aligned}
$$

This is Eq. (5.10).

# Appendix C

# Modified SLAE Method

The modified SLAE method can be applied if the bounds are constant (i.e., $A_t = A$ and $B_t = B$). Replacing each integral term in Eqs. (5.9) and (5.10) with the $n$-point Gaussian quadrature yields

$$\int_A^B P_\theta^{k+1}(x)\, f_\theta^{k+1}(x-s)\, dx \approx \sum_{i=1}^{n} \omega_i P_\theta^{k+1}(y_i)\, f_\theta^{k+1}(y_i - s)$$

where $\omega_i$ and $y_i$ are the weights and points of the Gaussian quadrature, respectively. Applying this approximation to Eq. (5.9) yields

$$P_\theta^k(s) = F_\theta^{k+1}(A-s) + \sum_{i=1}^{n} \omega_i P_\theta^{k+1}(y_i)\, f_\theta^{k+1}(y_i - s)$$

where $F_\theta^k(\cdot)$ is the CDF of $s_k$. Let $s = y_1, \cdots, y_n$. Then, a system of linear equations is obtained

$$P_\theta^k(y_j) - \sum_{i=1}^{n} \omega_i P_\theta^{k+1}(y_i)\, f_\theta^{k+1}(y_i - y_j) = F_\theta^{k+1}(A - y_j)$$

$j = 1, 2, \cdots, n$, which is, in the matrix form,

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}_{k+1} \end{bmatrix} \begin{bmatrix} \mathbf{P}_k \\ \mathbf{P}_{k+1} \end{bmatrix} = \mathbf{\Phi}_{k+1} \tag{C.1}$$

where

$\mathbf{M}_{k+1} = \left(\mathbf{M}_{k+1}^{(ij)}\right)$ and $\mathbf{M}_{k+1}^{(ij)} = -\omega_i f_\theta^{k+1}(y_i - y_j)$;

$\mathbf{\Phi}_{k+1} = \left[F_\theta^{k+1}(A - y_1), \cdots, F_\theta^{k+1}(A - y_n)\right]^T$;

$\mathbf{P}_k = \left[P_\theta^k(y_1), P_\theta^k(y_2), \cdots, P_\theta^k(y_n)\right]^T$.

Eq. (C.1) has $n$ equations but $2n$ unknowns, so it is under-determined and there are infinitely many solutions. Combining Eq. (C.1) for different times $k$ yields Eq. (5.11). Eq. (5.12) can be derived similarly.

# Appendix D

# Derivation of Eqs. (5.31) and (5.32)

Replacing $P_\theta^k(s)$ in Eq. (5.9) with Eq. (5.30) yields

$$\sum_{e=1}^{n_e} P_\theta^{k,(e)}(s) = F_\theta^{k+1}(A_{k+1} - s) + \sum_{e=1}^{n_e} [I_e^{k+1} \int_{x_{e-1}}^{x_e} f_\theta^{k+1}(x - s)\, \phi^{(e)}(x)\, dx] U^{k+1,(e)} \tag{D.1}$$

where

$$I_e^k = \begin{cases} 1, \text{ if } [x_{e-1}, x_e] \subseteq [A_k, B_k] \\ 0, \text{ else} \end{cases}$$

is an indicator function. Since the bounds $A_i$ and $B_i$ are included in the set of nodal points (see Eq. (5.26)), a segment $[x_{e-1}, x_e]$ lays either entirely inside or entirely outside the interval $[A_k, B_k]$. From Eq. (5.29) we have

$$x = x_{e-1} + l_e \xi, \quad \xi \in [0, 1]$$

and thus

$$\int_{x_{e-1}}^{x_e} f_\theta^{k+1}\left(x-s\right)\phi^{(e)}\left(x\right)dx = l_e\int_0^1 f_\theta^{k+1}(l_e\xi+x_{e-1}-s)\phi^{(e)}\left(\xi\right)d\xi$$

where the row vector $\phi^{(e)}\left(x\right)=[\phi_1^{(e)}\left(x\right),\phi_2^{(e)}\left(x\right)]$ is given in Eqs. (5.27) and (5.28). In general, this integral can be evaluated analytically and straightforwardly for most distributions, and then a system of linear equations is obtained. We proceed with the derivation by assuming $s_t$ are independent and Gaussian distributed, i.e., $f_\theta^t\left(s\right)=\mathcal{N}\left(s;\mu_t,\sigma_t^2\right)$, but the method is generally applicable to other distributions. Denote

$$\begin{aligned}
Q_1^{k+1,(e)}\left(s\right) =& I_e^{k+1}l_e\int_0^1 f_\theta^{k+1}(l_e\xi+x_{e-1}-s)\left(1-\xi\right)d\xi \\
=& I_e^{k+1}q_1^{k+1,(e)}\left(s\right) \qquad \text{(D.2)} \\
Q_2^{k+1,(e)}\left(s\right) =& I_e^{k+1}l_e\int_0^1 f_\theta^{k+1}(l_e\xi+x_{e-1}-s)\xi d\xi \\
=& I_e^{k+1}q_2^{k+1,(e)}\left(s\right) \qquad \text{(D.3)}
\end{aligned}$$

where for the Gaussian distribution,

$$\begin{aligned}
q_1^{k+1,(e)}\left(s\right) &= l_e\int_0^1 f_\theta^{k+1}(l_e\xi+x_{e-1}-s)\left(1-\xi\right)d\xi \\
&= F\left(1\right)-F\left(0\right)-q_2^{k+1,(e)}\left(s\right) \qquad \text{(D.4)} \\
q_2^{k+1,(e)}\left(s\right) &= l_e\int_0^1 f_\theta^{k+1}(l_e\xi+x_{e-1}-s)\xi d\xi \\
&= \frac{-\sigma^2}{l_e}[f_\theta^{k+1}(l_e+x_{e-1}-s)-f_\theta^{k+1}(x_{e-1}-s)] \\
&\quad -\frac{\left(x_{e-1}-s-\mu\right)}{l_e}\left[F\left(1\right)-F\left(0\right)\right] \qquad \text{(D.5)} \\
F(x) &= F_\theta^{k+1}(x;\frac{-x_{e-1}+s+\mu}{l_e},\frac{\sigma^2}{l_e^2})
\end{aligned}$$

and $F_\theta^{k+1}(x;\bar{x},\sigma^2)$ is the Gaussian CDF of $s_{k+1}$ with mean $\bar{x}$ and variance $\sigma^2$. So plugging Eqs. (D.2) and (D.3) into Eq. (D.1) yields

$$\sum_{e=1}^{n_e} P_\theta^{k,(e)}(s) = F_\theta^{k+1}(A_{k+1}-s) + Q_{k+1}(s)\mathbf{U}_{k+1} \tag{D.6}$$

where

$$\begin{aligned}
&\mathbf{U}_k = [\ U_0^k, U_1^k, \cdots, U_{n_e}^k\ ]^T \\
&Q_k(s) = Q_k^1(s) + Q_k^2(s) \\
&Q_k^1(s) = [\ Q_1^{k,(1)}(s),\ \ Q_1^{k,(2)}(s),\ \ \cdots\ \ Q_1^{k,(n_e)}(s),\ \ 0\ ]; \\
&Q_k^2(s) = [\ 0,\ \ Q_2^{k,(1)}(s),\ \ Q_2^{k,(2)}(s),\ \ \cdots\ \ Q_2^{k,(n_e)}(s)\ ].
\end{aligned}$$

By choosing $s = x_0, x_1, \cdots, x_{n_e}$, a system of linear equations can be obtained,

$$\begin{bmatrix} U_0^k \\ U_1^k \\ \vdots \\ U_{n_e}^k \end{bmatrix} = \begin{bmatrix} F_\theta^{k+1}(A_{k+1}-x_0) \\ F_\theta^{k+1}(A_{k+1}-x_1) \\ \vdots \\ F_\theta^{k+1}(A_{k+1}-x_{n_e}) \end{bmatrix} + \begin{bmatrix} Q_{k+1}(x_0) \\ Q_{k+1}(x_1) \\ \vdots \\ Q_{k+1}(x_{n_e}) \end{bmatrix} \begin{bmatrix} U_0^{k+1} \\ U_1^{k+1} \\ \vdots \\ U_{n_e}^{k+1} \end{bmatrix}$$

which can be written in a matrix form as

$$\mathbf{U}_k = \mathbf{F}_{k+1} + \mathbf{Q}_{k+1}\mathbf{U}_{k+1} \tag{D.7}$$

Combining Eq. (D.7) with $k = 0, 1, \cdots, K-1$ yields Eq. (5.31), where

$$\mathbf{Q}_k = [Q_k(x_0)^T, Q_k(x_1)^T, \cdots, Q_k(x_{n_e})^T]^T$$

and $\hat{\mathbf{U}}_K$ can be obtained from the termination rule (in the same way as Eq. (5.25)). Eq. (5.32) can be derived similarly. Note that for other distributions of $s_k$, only $\boldsymbol{F}_k$ and $\mathbf{Q}_k$ $(k = 1, 2, \cdots, K)$ need be re-calculated accordingly.

# Appendix E

# Derivation of Eq. (5.37)

The governing equation for $L_\theta^k(s)$ can be derived as follows. If CUSUM starts at time $k$ with the initial value $s$ (i.e., $S_k = s$), then one of the following three mutually exclusive events must occur:

$$\Gamma_1 = \{S_{k+1} \geq B_{k+1}\}, \quad \Gamma_2 = \{A_{k+1} < S_{k+1} < B_{k+1}\}, \quad \Gamma_3 = \{S_{k+1} = A_{k+1}\}$$

Applying the total expectation theorem, Eq. (5.36) becomes

$$\begin{aligned} L_\theta^k(s) = & E_\theta\left[\tau_k - k | \Gamma_1, S_k = s\right] P_\theta\left\{\Gamma_1 \,|S_k = s\right\} \\ & + E_\theta\left[\tau_k - k | \Gamma_2, S_k = s\right] P_\theta\left\{\Gamma_2 \,|S_k = s\right\} \\ & + E_\theta\left[\tau_k - k | \Gamma_3, S_k = s\right] P_\theta\left\{\Gamma_3 \,|S_k = s\right\} \end{aligned} \tag{E.1}$$

The conditional probabilities are computed straightforwardly

$$P_\theta\left\{\Gamma_1 \,|S_k = s\right\} = \int_{B_{k+1}}^{+\infty} f_\theta^{k+1}(x - s)\, dx \tag{E.2}$$

$$P_\theta\left\{\Gamma_2 \,|S_k = s\right\} = \int_{A_{k+1}}^{B_{k+1}} f_\theta^{k+1}(x - s)\, dx \tag{E.3}$$

$$P_\theta\left\{\Gamma_3 \,|S_k = s\right\} = \int_{-\infty}^{A_{k+1}} f_\theta^{k+1}(x - s)\, dx \tag{E.4}$$

Further, it is clear that

$$E_\theta\left[\tau_k - k | \Gamma_1, S_k = s\right] = 1 \tag{E.5}$$

since conditioned on $\Gamma_1$, CUSUM stops at time $k+1$ (i.e., $\tau_k = k+1$). For event $\Gamma_3$, if $S_k + s_{k+1} < A_{k+1}$, then $S_{k+1}$ will be reset to $A_{k+1}$ (see Eq. (5.33)). Hence after one step, the test restarts at time $k+1$ with the initial value $A_{k+1}$. This newly initialized CUSUM has an ARL $L_\theta^{k+1}(A_{k+1})$. Thus, the ARL conditioned on $\Gamma_3$ is

$$E_\theta\left[\tau_k - k | \Gamma_3, S_k = s\right] = 1 + L_\theta^{k+1}(A_{k+1}) \tag{E.6}$$

If $\Gamma_2$ occurs, after one step, the test statistic $S_{k+1}$ falls between $A_{k+1}$ and $B_{k+1}$, and CUSUM continues. Then, the PDF of $S_{k+1}$ conditioned on $\Gamma_2$ is

$$f_\theta^{k+1}(x | \Gamma_2, S_k = s) = \frac{f_\theta^{k+1}(x-s)}{\int_{A_{k+1}}^{B_{k+1}} f_\theta^{k+1}(x-s)dx}, \quad A_{k+1} < x < B_{k+1}$$

By the total expectation theorem, we have

$$E_\theta\left[\tau_k - k | \Gamma_2, S_k = s\right] = \int_{A_{k+1}}^{B_{k+1}} E_\theta\left[\tau_k - k | S_{k+1} = x, \Gamma_2, S_k = s\right] f_\theta^{k+1}(x | \Gamma_2, S_k = s)dx$$

Conditioned on $\Gamma_2$ and $S_{k+1} = x$, it is known that the test does not terminate at $k+1$. It is equivalent to the new test starting at $k+1$ with the initial value $S_{k+1} = x$, and hence

$$E_\theta\left[\tau_k - k | S_{k+1} = x, \Gamma_2, S_k = s\right] = 1 + L_\theta^{k+1}(x)$$

So

$$
\begin{aligned}
E_\theta\left[\tau_k - k | \Gamma_2, S_k = s\right] =& 1 + \int_{A_{k+1}}^{B_{k+1}} L_\theta^{k+1}(x) f_\theta^{k+1}(x|\Gamma_2, S_k = s) dx \\
=& 1 + \frac{\int_{A_{k+1}}^{B_{k+1}} L_\theta^{k+1}(x) f_\theta^{k+1}\left(x - s\right) dx}{\int_{A_{k+1}}^{B_{k+1}} f_\theta^{k+1}\left(x - s\right) dx}
\end{aligned} \tag{E.7}
$$

Plugging Eqs. (E.2)–(E.7) into Eq. (E.1) yields Eq. (5.37).

# Bibliography


[1] B. D. O. Anderson and J. B. Moore. *Optimal filtering*. Prentice Hall, Englewood Cliffs, NJ, 1979.

[2] T. W. Anderson. A modification of the sequential probability ratio test to reduce the sample size. *The Annals of Mathematical Statistics*, 31(1):165–197, March 1960.

[3] M. M. Andreasen. Non-linear DSGE models, the central difference Kalman filter, and the mean shifted particle filter. *CREATES Research Paper*, 33, June 2008. Available at SSRN: http://ssrn.com/abstract=1148079.

[4] I. Arasaratnam. *Cubature Kalman Filtering: theory & applications*. PhD thesis, Department of Electrical & Computer Engineering, McMaster University, April 2009.

[5] I. Arasaratnam and S. Haykin. Square-root quadrature Kalman filtering. *IEEE Transactions on Signal Processing*, 56(6):2589–2593, June 2008.

[6] I. Arasaratnam and S. Haykin. Cubature Kalman filters. *IEEE Transactions on Automatic Control*, 54(6):1254–1269, June 2009.

[7] I. Arasaratnam, S. Haykin, and R. J. Elliott. Discrete-time nonlinear filtering algorithms using Gauss–Hermite quadrature. *Proceedings of the IEEE*, 95(5):953–977, May 2007.

[8] I. Arasaratnam, S. Haykin, and T. R. Hurd. Cubature Kalman filtering for continuous-discrete systems: theory and simulations. *IEEE Transactions on Signal Processing*, 58(10):4977–4993, October 2010.

[9] P. Armitage. Restricted sequential procedures. *Biometrika*, 44(1/2):9–26, 1957.

[10] L. A. Aroian. Exact truncated sequential tests for the exponential density function. In *Proceedings of Ninth National Symposium on Reliability and Quality Control*, pages 470–486, 1963.

[11] L. A. Aroian. Sequential analysis, direct method. *Technometrics*, 10(1):125–132, February 1968.

[12] L. A. Aroian. Applications of the direct method in sequential analysis. *Technometrics*, 18(3):301–306, August 1976.

[13] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particles filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactionson Signal Processing*, 50(2):174–188, February 2002.

[14] M. Asadzadeh and K. Eriksson. On adaptive finite element methods for Fredholm integral equations of the second kind. *SIAM Journal on Numerical Analysis*, 31(3):831–855, 1994.

[15] M. Athans, R. P. Wishner, and A. Bertolini. Suboptimal state estimation for continuous-time nonlinear systems from discrete noisy measurements. *IEEE Transactions on Automatic Control*, 13(5):504–514, October 1968.

[16] K. E. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, 1997.

[17] J. W. Austin and C. T. Leondes. Statistically linearized estimation of reentry trajectories. *IEEE Transactions on Aerospace and Electronic Systems*, 17(1):54–61, January 1981.

[18] C. T. H. Baker. *The Numerical Treatment of Integral Equations*. Clarendon Press, Oxford, England, 1977.

[19] Y. Bar-Shalom, editor. *Multitarget-Multisensor Tracking: Advanced Applications*. Artech House, 1990.

[20] Y. Bar-Shalom and W. D. Blair, editors. *Multitarget-Multisensor Tracking: Applications and Advances Volume III*. Artech House, 2000.

[21] Y. Bar-Shalom and X. R. Li. *Multitarget-Multisensor Tracking: Principles and Techniques*. YBS Publishing, Storrs, CT, 1995.

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

[23] Y. Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 11(5):451–460, September 1975.

[24] R. D. Bass, V. D. Norum, and L. Swartr. Optimal multichannel nonlinear filtering. *Journal of Mathematical Analysis and Applications*, 16:152–164, 1966.

[25] M. Basseville and I. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. Prentice Hall, Englewood Cliffs, NJ, 1993.

[26] M. Baum, B. Noack, F. Beutler, D. Itte, and U. D. Hanebeck. Optimal Gaussian filtering for polynomial systems applied to association-free multi-target tracking. In *Proceedings of the 14th International Conference on Information Fusion*, Chicago, IL, 2011.

[27] B. Baygun and A. O. Hero. Optimal simultaneous detection and estimation under a false alarm constraint. *IEEE Transactions on Information Theory*, 41:688–703, 1995.

[28] B. M. Bell and F. W. Cathey. The iterated Kalman filter update as a Gauss-Newton method. *IEEE Transactions on Automatic Control*, 38(2):294–297, February 1993.

[29] R. L. Bellaire, E. W. Kamen, and S. M. Zabin. A new nonlinear iterated filter with applications to target tracking. In *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, volume 2561, pages 240–251, 1995, July 1995.

[30] D. P. Bertsekas. The auction algorithm for assignment and other network flow problems: a tutorial. *Interfaces*, 20(4):133–149, July – August 1990.

[31] S. S. Blackman. *Multiple Target Tracking with Radar Applications*. Artech House, Norwood, MA, 1986.

[32] S. S. Blackman. Multiple hypothesis tracking for multiple target tracking. *IEEE Aerospace and Electronic Systems Magazine*, 19(1):5–18, January 2004.

[33] S. S. Blackman and R. Popoli. *Design and analysis of modern tracking systems*. Artech House, 1999.

[34] H. A. P. Blom. A sophisticated tracking algorithm for ATC surveillance data. In *Proceedings of the International Radar Conference*, Paris, France, May 1984.

[35] F. Bourgeois and J. C. Lassalle. An extension of the Munkres algorithm for the assignment problem to rectangular matrices. *Communications of the ACM*, 14(12):802–804, December 1971.

[36] R. E. Burkard, M. Dell'Amico, and S. Martello. *Assignment problems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2009.

[37] D. L. Burkholder and R. A. Wijsman. Optimum properties and admissibility of sequential tests. *Annals of Mathematical Statistics*, 34:1–17, 1963.

[38] D. A. Castanon. New assignment algorithms for data association. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, April 1992.

[39] S. Challa and G. W. Pulford. Joint target tracking and classification using radar and ESM sensors. *IEEE Transactions on Aerospace and Electronic Systems*, 37(3):1039–1055, July 2001.

[40] S. Colegrove and J. Ayliffe. An extension of probabilistic data association to include track initiation and termination. In *Digest of the 20th IREE Intemational Convention*, pages 853–856, September 1985.

[41] S. B. Colgrove, A. W. Davis, and J. K. Ayliffe. Track initiation and nearest neighbours incorporated into probabilistic data assodation. *Journal of Electrical and Electronics Engineering, Australia*, 6(3):191–197, September 1986.

[42] R. Cools. Advances in multidimensional integration. *Journal of Computational and Applied Mathematics*, 149(1):1–12, December 2002.

[43] O. L. V. Costa. Linear minimum mean square error estimation for discrete-time Markovian jump linear systems. *IEEE Transactions on Automatic Control*, 39(8):1685–1689, August 1994.

[44] D. R. Cox and H. D. Miller. *The Theory of Stochastic Processes*. Willey, New York, 1965.

[45] I. J. Cox and S. L. Hingorani. An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(2):138–150, February 1996.

[46] I. J. Cox and M. L. Miller. On finding ranked assignments with application to multitarget tracking and motion correspondence. *IEEE Transactions on Aerospace and Electronic Systems*, 31(1):486–489, January 1995.

[47] D. A. Darling and A. J. F. Siegert. The first passage problem for a continuous Markov process. *The Annals of Mathematical Statistics*, 24(4):624–639, December 1953.

[48] F. Daum. Nonlinear filters: beyond the Kalman filter. *IEEE Aerospace and Electronic Systems Magazine*, 20(8):57–69, August 2005.

[49] S. J. Davey and D. A. Gray. Integrated track maintenance for the PMHT via the hysteresis model. *IEEE Transactions on Aerospace and Electronic Systems*, 43(1):93–111, January 2007.

[50] S. Deb, M. Yeddanapudi, K. Pattipati, and Y. Bar-Shalom. A generalized S-D assignment algorithm for multisensor-multitarget state estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 33(2):523–538, April 1997.

[51] 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, Series B (Methodological)*, 39(1):1–38, 1977.

[52] W. F. Denham and S. Pines. Sequential estimation when measurement function nonlinearity is comparable to measurement error. *AIAA Journal*, 4(6):1071–1076, 1966.

[53] A. Doucet, N. de Freitas, and . N. Gordon, editors. *Sequential Monte Carlo methods in practice*. Springer-Verlag, Series Statistics for Engineering and Information Science, New York, NY, USA, 2001.

[54] A. Doucet and A. M. Johansen. *A tutorial on particle filtering and smoothing: fifteen years later*. Oxford Handbook of Nonlinear Filtering, 2011.

[55] O. E. Drummond. Methodologies for performance evaluation of multitarget multisensor tracking. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, volume 3809, pages 355–369, 1999.

[56] O. E. Drummond, D. A. Castanon, and M. S. Bellovin. Comparison of 2-D assignment algorithms for rectangular, floating point cost matrices. In *Proceedings of SDI Panels on Tracking*, number 4, pages 81–97, December 1990.

[57] O. E. Drummond and B. E. Fridling. Ambiguities in evaluating performance of multiple target tracking algorithms. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, volume 1698, pages 326–337, 1992.

[58] Z. Duan, Y. Liu, and X. R. Li. Recursive LMMSE filtering for target tracking with range and direction cosine measurements. In *Proceedings of 13th International Conference on Information Fusion*, Edinburgh, UK, July 2010.

[59] D. Dunham and R. Hutchins. Tracking multiple targets in cluttered environments with a probabilistic multihypothesis tracker. In *Proceedings of SPIE Conference on Acquisition, Tracking, and Pointing XI*, volume 3086, pages 284–295, June 1997.

[60] B. Eisenberg, B. K. Ghosh, and G. Simons. Properties of generalized sequential probability ratio test. *The Annals of Statistics*, 4(2):237–251, 1976.

[61] B. Epstein, A. A. Patterson, and C. R. Qualls. The exact analysis of sequential life tests with particular application to agree plans. In *Proceedings of Aerospace Reliability and Maintainability Conference*, pages 284–311, Washington, D. C., 1963.

[62] A. Fairna, D. Benvenuti, and B. Ristic. A comparative study of the Benes filtering problem. *Signal Processing*, 82(2):133–147, February 2002.

[63] A. Fairna, B. Ristic, and D. Benvenuti. Tracking a ballistic target: comparison of several nonlinear filters. *IEEE Transactions on Aerospace and Electronic Systems*, 38(3):1916–1624, July 2002.

[64] R. J. Fitzgerald. *"Development of Practical PDA Logic for Multitarget tracking by Microprocessor", in Multitarget-Multisensor tracking: Advanced Applications*, chapter 1. Artech House, 1990.

[65] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal of Oceanic Engineering*, 8:173–183, 1983.

[66] A. J. Fredriksen, D. Middleton, and V. Vandelinde. Simultaneous detection and estimation under multiple hypotheses. *IEEE Transactions on Information Theory*, 18(5):607–614, September 1972.

[67] D. Freeman and L. Weiss. Sampling plans which minimize the maximum expected sample size. *Journal of American Statistic Association*, 59:67–88, 1964.

[68] H. Gauvrit, J. P. L. Cadre, and C. Jauffret. A formulation of multitarget tracking as an incomplete data problem. *IEEE Transactions on Aerospace and Electronic Systems*, 33(4):1242–1257, October 1997.

[69] A. Gelb, editor. *Applied Optimal Estimation*. MIT Press, 1974.

[70] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 18(4):209–232, 1998.

[71] E. Giannopoulos, R. Streit, and P. Swasxek. Probabilistic multi-hypothesis tracking in a multi-sensor, multi-target environment. In *Proceedings fo the First Australian Data Fusion Symposium*, pages 184–189, Adelaide, SA , Australia, November 1996.

[72] A. L. Goel and S. M. Wu. Determination of A.R.L. and a contour nomogram for CUSUM charts to control normal mean. *Technometrics*, 13(2):221–230, May 1971.

[73] Z. Govindarajulu. *Sequential Statistics*. World Scientific Publishing Company, Inc., Singapore, 2004.

[74] P. Gurftl and N. J. Kasdin. Two-step optimal estimator for three dimensional target tracking. In *Proceedings of the American Conbol Conference*, pages 209–214, Anchorage, AK, May 2002.

[75] W. Hackbusch. *Integral Equations: Theory and Numerical Treatment*. Birkhuser Basel, June 1995.

[76] M. T. Hadidi and C. S. Schwartz. Linear recursive state estimators under uncertain observations. *IEEE Transactions on Automatic Control*, 24(6):944–948, December 1979.

[77] G. Haupt, N. Kasdin, G. Keiser, and B. Parkinson. An optimal recursive iterative algorithm for discrete nonlinear least square state estimation. In *Proceedings of AIAA Guidance, Navigation, and Control Conference*, pages 404–417, Baltimore, MD, USA, May 1995.

[78] G. T. Haupt, N. J. Kasdin, G. M. Keiser, and B. W. Parkinson. An optimal recursive iterative algorithm for discrete nonlinear least square state estimation. *AIAA Journal of Gudance, Control and Dynamics*, 19(3):643–649, 1996.

[79] M. Havlicek, K. J. Friston, J. Jan, M. Brazdil, and V. D. Calhoun. Dynamic modeling of neuronal responses in fMRI using cubature Kalman filtering. *NeuroImage*, 56(4):2109–2128, June 2011.

[80] S. Hawking. *A Brief History of Time*. Bantam Dell Publishing Group, 1988.

[81] K. Heine. A survey of sequential Monte Carlo methods. Master's thesis, Department of Science and Engineering, Tampere University of Technology, September 2005.

[82] R. E. Helmick and G. A. Watson. Interacting multiple model integrated probabilistic data association filters IMM-IPDAF for track formation on maneuvering targets in cluttered environments. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, Orlando, FL, USA, April 1994.

[83] R. Henriksen. A correction of a common error in truncated second order nonlinear filters. *Modeling, Identification and Control*, 1(3):187–193, 1980.

[84] R. Henriksen. The truncated second-order nonlinear filter revisited. *IEEE Transactions on Automatic Control*, 27(1):247–251, February 1982.

[85] S. Herman and P. Moulin. A particle filtering approach to FM-band passive radar tracking and automatic target recognition. In *Proceedings of the IEEE Aerospace Conference*, volume 4, pages 1789–1808, 2002.

[86] J. Hoffman and R. Mahler. Multitarget miss distance via optimal assignment. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 34(3):327–336, 2004.

[87] J. Hou, X. R. Li, V. P. Jilkov, and Z. Jing. Sequential detection of RGPO in target tracking by decomposition and fusion approach. In *Proceedings of 15th International Conference on Information Fusion*, Singapore, July 2012.

[88] A. Houles and Y. Bar-Shalom. Multisensor tracking of a maneuvering target in clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 25(2):176–189, March 1989.

[89] M. F. Huber. Adaptive Gaussian mixture filter based on statistical linearization. In *Proceedings of 14th International Conference on Information Fusion*, pages 562–569, Chicago, IL, USA, July 2011.

[90] C. Hue, J.-P. L. Cadre, and P. Perez. Sequential Monte Carlo methods for multiple target tracking and data fusion. *IEEE Transactions on Signal Processing*, 50(2):309–325, February 2002.

[91] R. G. Hutchins and D. T. Dunham. Evaluation of a probabilistic multihypothesis tracking algorithm in cluttered environments. In *Proceedings of the Thirtieth Asilomar Conference on Signals, Systems and Computers*, 1996.

[92] Y. Ikebe. The Galerkin method for the numerical solution of Fredholm integral equations of the second kind. *SIAM Review*, 14(3):465–491, July 1972.

[93] A. Irle. Asymptotic optimality of general sequential probability ratio tests. *Scandinavian Journal of Statistics*, 17(4):321–332, 1990.

[94] K. Ito and K. Xiong. Gaussian filters for nonlinear filtering problems. *IEEE Transactions on Automatic Control*, 45(5):910–927, May 2000.

[95] A. H. Jazwinski. Filtering for nonlinear dynamical system. *IEEE Transactions on Automatic Control*, 11(4):765–766, October 1966.

[96] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. Academic Press, New York, 1970.

[97] A. J. Jerri. *Introduction to Integral Equations with Applications*. Marcel Dekker Inc., March 1985.

[98] C. Jian and Z. Xiaoping. A design on adaptive two-step filter applied in nonlinear system. In *Proceedings of International Conference on Intelligent Computation Technology and Automation (ICICTA)*, pages 929–931, Shenzhen, Guangdong, China, March 2011.

[99] R. Jonker and A. Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38:325–340, 1987.

[100] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3):477–482, March 2000.

[101] S. J. Julier. The scaled unscented transformation. In *Proceedings of the American Control Conference*, volume 6, Anchorage, AK, USA, May 2002.

[102] S. J. Julier. The spherical simplex unscented transformation. In *Proceedings of the American Control Conference*, volume 3, Denver, CO, USA, June 2003.

[103] S. J. Julier and J. K. Uhlmann. A consistent, debiased method for converting between polar and cartesian coordinate systems. In *Proceedings of SPIE Acquisition, Tracking, and Pointing XI*, volume 3086, 1997.

[104] S. J. Julier and J. K. Uhlmann. A new extension of the Kalman filter to nonlinear systems. In *Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, Simulation and Controls*, 1997.

[105] S. J. Julier and J. K. Uhlmann. Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations. In *Proceedings of the American Control Conference*, volume 2, Anchorage, AK, 2002.

[106] S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of The IEEE*, 92(3):401–422, March 2004.

[107] S. J. Julier, J. K. Uhlrnann, and H. F. Durrant-Whyte. A new approach for filtering nonlinear systems. In *Proceedings of the American Control Conference*, volume 3, Seattle, WA , USA, June 1995.

[108] I. Kadar, E. R. Eadan, and R. R. Gassner. Comparison of robustized assignment algorithms. In *Proceedings of Signal Processing, Sensor Fusion, and Target Recognition VI*, Orlando, FL, April 1997.

[109] T. T. Kadota. Optimal, causal, simultaneous detection and estimation of random signal fields in a Gaussian noise field. *IEEE Transactions on Information Theory*, 24(3):297–308, May 1987.

[110] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[111] R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. *Journal of Basic Engineering*, 83:95–107, 1961.

[112] E. W. Kamen. Multiple target tracking based on symmetric measurement equations. *IEEE Transactions on Automatic Control*, 37(3):371–374, 1992.

[113] E. W. Kamen and C. R. Sastry. Multiple target tracking using products of position measurements. *IEEE Transactions on Aerospace and Electronic Systems*, 29(2):476–493, 1993.

[114] K. W. Kemp. Formal expressions which can be used for the determination of the operating characteristic and average sample number of a simple sequential test. *Journal of the Royal Statistical Society*, B-29(2):248–262, 1967.

[115] K. W. Kemp. A simple procedure for determining upper and lower limits for the average sample run length of a cumulative sum scheme. *Journal of the Royal Statistical Society. Series B (Methodological)*, 29(2):263–265, 1967.

[116] J. Kiefer and L. Weiss. Some properties of generalized sequential probability ratio tests. *The Annals of Mathematical Statistics*, 28(1):57–74, March 1957.

[117] T. Kirubarajan and Y. Bar-shalom. Probabilistic data association techniques for target tracking in clutter. *Proceedings of the IEEE*, 92(3):536–557, March 2004.

[118] W. Koch and G. V. Keuk. Multiple hypothesis track maintenance with possibly unresolved measurements. *IEEE Transactions on Aerospace and Electronic Systems*, 33(3):883–892, July 1997.

[119] P. Konstantinova, A. Udvarev, and T. Semerdjiev. A study of a target tracking algorithm using global nearest neighbor approach. In *Proceedings of the 4th international conference conference on Computer systems and technologies*, New York, NY, 2003.

[120] C. Kraft. Some conditions for consistency and uniform consistency of statistical procedures. *In University of California Publication in Statistics*, 1:125–142, 1955.

[121] M. L. Krieg and D. A. Gray. Multi-sensor probabilistic multi-hypothesis tracking using dissimilar sensors. In *Proceedings of SPIE Conference 3086*, Orlando, FL, April 1997.

[122] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, (2):83–97, 1955.

[123] H. Kushner. Approximations to optimal nonlinear filters. *IEEE Transactions on Automatic Control*, 12(5):546–556, October 1967.

[124] P. K. Kythe and D. Wei. *An Introduction to Linear and Nonlinear Finite Element Analysis: A Computational Approach*. Birkhauser, Boston, 2003.

[125] T. L. Lai. Optimal stopping and sequential tests which minimize the maximum expected sample size. *The Annals of Statistics*, 1(4):659–673, July 1973.

[126] T. L. Lai. Asymptotic optimality of invariant sequential probability ratio tests. *The Annals of Statistics*, 9(2):318–333, 1981.

[127] T. L. Lai. Sequential analysis: some classical problems and new challenges. *Statistica Sinica*, 11:303–408, 2001.

[128] T. Lefebvre, H. Bruyninckx, and J. D. Schuller. Comment on "a new method for the nonlinear transformation of means and covariances in filters and estimators" (and authors' reply). *IEEE Transactions on Automatic Control*, 47(8):1409–1409, August 2002.

[129] T. Lefebvre, H. Bruyninckx, and J. D. Schutter. Kalman filters for non-linear systems: a comparison of performance. *International Journal of Control*, 77(7):639–653, May 2004.

[130] T. Lefebvre, H. Bruyninckx, and J. D. Schutter. *Nonlinear Kalman Filtering for Force-Controlled Robot Tasks*. Springer, Berlin, 2005.

[131] E. L. Lehmann. *Testing Statistical Hypotheses*. CRC Press, Boca Raton , FL, 2 edition, 1986.

[132] D. Lerro and Y. Bar-Shalom. Automated tracking with target amplitude information. In *Proceedings of American Control Conference*, pages 2875–2880, San Diego, CA, May 1990.

[133] D. Lerro and Y. Bar-Shalom. Automatic track formation with target amplitude information. In *Proceedings of the Ocean Technologies and Opportunities in the Pacific for the 90's'*, pages 1460–1467, October 1991.

[134] D. Lerro and Y. Bar-Shalom. Tracking with debiased consistent converted measurements versus EKF. *IEEE Transactions on Aerospace and Electronic Systems*, 29(3):1015–1022, 1993.

[135] M. Levedahl. Performance comparison of 2D assignment algorithms for assigning truth objects to measured tracks. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, Orlando, FL, April 2000.

[136] W. E. Leven and A. D. Lanterman. Multiple target tracking with symmetric measurement equations using unscented Kalman and particle filters. In *Proceedings of the Thirty-Sixth Southeastern Symposium on System Theory*, 2004.

[137] W. F. Leven and A. D. Lanterman. Unscented Kalman filters for multiple target tracking with symmetric measurement equations. *IEEE Transactions on Automatic Control*, 54(2):370–375, 2009.

[138] N. Li and X. R. Li. Target perceivability: an integrated approach to tracker analysis and design. In *Proceedings of International Conference on Information Fusion*, pages 174–181, Las Vegas, NV, USA, July 1998.

[139] N. Li and X. R. Li. Target perceivability and its applications. *IEEE Transactions on Signal Processing*, 49(11):2588–2604, November 2001.

[140] N. Li and X. R. Li. Tracker design based on target perceivability. *IEEE Aerospace and Electronic Systems*, 37(1):214–225, January 2001.

[141] X. R. Li. Tracking in clutter with strongest neighbor measurements, part I: theoretical analysis. *IEEE Transactions on Automatic Control*, 43(11):1560–1578, November 1998.

[142] X. R. Li. Recursibility and optimal linear estimation and filtering. In *Proceedings of IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, December 2004.

[143] X. R. Li. Optimal Bayes joint decision and estimation. In *Proceedings of International Conference on Information Fusion*, pages 1316–1323, Quebec City, Canada, July 2007.

[144] X. R. Li. Lecture notes: estimation and Kalman filter. 2011.

[145] X. R. Li. Measure of nonlinearity for stochastic systems. In *Proceedings of 15th International Conference on Information Fusion*, Singapore, July 2012.

[146] X. R. Li and Y. Bar-Shalom. Theoretical analysis and performance prediction of tracking in clutter with strongest neighbor filters. In *Proceedings of 34th IEEE Conference on Decision and Control*, pages 2758–2763, New Orleans, LA, December 1995.

[147] X. R. Li and Y. Bar-Shalom. Tracking in clutter with nearest neighbor filters: analysis and performance. *IEEE Transactions on Aerospace and Electronic Systems*, 32(3):995–1010, July 1996.

[148] X. R. Li and V. P. Jilkov. A survey of maneuvering target tracking – part III: Measurement models. In *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, volume 4473, 2001.

[149] X. R. Li and V. P. Jilkov. A survey of maneuvering target tracking – part IV: Decision-based methods. In *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, pages 423–446, Orlando, FL, USA, April 2002.

[150] X. R. Li and V. P. Jilkov. Survey of maneuvering target tracking – part I: dynamic models. *IEEE Transaction on Aerospace and Electronic Systems*, 39(4):1333–1364, Oct. 2003.

[151] X. R. Li and V. P. Jilkov. A survey of maneuvering target tracking: approximation techniques for nonlinear filtering. In *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, volume 5428, 2004.

[152] X. R. Li and V. P. Jilkov. Survey of maneuvering target tracking—part V: multiple-model methods. *IEEE Transaction on Aerospace and Electronic Systems*, 41(4):1255–1321, Oct. 2005.

[153] X. R. Li and V. P. Jilkov. Survey of maneuvering target tracking – part II: motion models of ballistic and space targets. *IEEE Transactions on Aerospace and Electronic Systems*, 46(1):96–119, 2010.

[154] X. R. Li and V. P. Jilkov. A survey of maneuvering target tracking – part VIa: density-based exact nonlinear filtering. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, volume 7698, Orlando, FL, April 2010.

[155] X. R. Li and V. P. Jilkov. A survey of maneuvering target tracking – part VIb: approximate nonlinear density filtering in mixed time. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, volume 7698, Orlando, FL, April 2010.

[156] X. R. Li and V. P. Jilkov. A survey of maneuvering target tracking – part VIc: approximate nonlinear density filtering in discrete time. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, volume 8393, Baltimore, MD, 2012.

[157] X. R. Li and N. Li. Intellegent PDAF: refinement of IPDA for tracking in clutter. In *Southeastern Symposium on System Theory*, pages 133–137, Cookeville, TN, USA, March 1997.

[158] X. R. Li, M. Yang, and J. Ru. Joint tracking and classification based on Bayes joint decision and estimation. In *Proceedings of International Conference on Information Fusion*, pages 1421–1428, Quebec City, Canada, July 2007.

[159] X. R. Li and Z. Zhao. Measuring estimator's credibility: noncredibility index. In *Proceedings of the 9th International Conference on Information Fusion*, 2006.

[160] X. R. Li, Z. Zhao, and V. P. Jilkov. Practical measures and test for credibility of an estimator. In *Proceedings of Workshop on Estimation, Tracking, and Fusion — A Tribute to Yaakov Bar-Shalom*, pages 481–495, Monterey, CA, USA, May 2001.

[161] X. R. Li, Z. L. Zhao, and X. B. Li. Evaluation of estimation algorithms – part II: credibility tests. *IEEE Transactions on Systems, Man, and CyberneticsPart A: Systems and Humans*, 42(1):147–163, 2012.

[162] X. R. Li and X. Zhi. Probabalistic strongest neighbor filter for tracking in clutter. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, volume 2759, pages 230–241, 1996.

[163] X. R. Li and X. Zhi. PSNF: a refined strongest neighbor filter for tracking in clutter. In *Proceedings of the 35th IEEE Decision and Control*, volume 3, pages 2557–2562, Kobe, Japan, December 1996.

[164] J. S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer, New York, NY, USA, 2001.

[165] Y. Liu and S. D. Blostein. Optimality of the sequential probability ratio test for nonstationary observations. *IEEE Transactions On Information Theory*, 38(1):177–182, 1992.

[166] Y. Liu and X. R. Li. Performance analysis of Wald's SPRT with independent but non-stationary log-likelihood ratios. In *proceedings of 14th International Conference on Information Fusion*, Chicago, Illinois, USA, July 2011.

[167] Y. Liu and X. R. Li. Recursive joint decision and estimation based on generalized Bayes risk. In *Proceedings of the 14th International Conference on Information Fusion*, pages 1–8, Chicago, IL, USA, July 2011.

[168] Y. Liu and X. R. Li. Sequential multiple-model detection of target maneuver termination. In *Proceedings of the 14th International Conference on Information Fusion*, Chicago, IL, USA, July 2011.

[169] Y. Liu and X. R. Li. Operating characteristic and average sample number functions of truncated sequential probability ratio test. In *Proceedings of 15th International Conference on Information Fusion*, Singapore, July 2012.

[170] Y. Liu and X. R. Li. Average run length function of CUSUM test with independent but non-stationary log-likelihood ratios. In *Proceedings of 2013 American Control Conference*, Washington, D. C., June 2013.

[171] A. Logothetis, V. Krishnamurthy, and J. Holst. On maneuvering target tracking via the PMHT. In *Proceedings of the 36th Conference on Decision and Control*, pages 5024–5029, San Diego, CA, USA, December 1997.

[172] G. Lorden. 2-SPRT's and the modified Kiefer-Weiss problem of minimizing an expected sample size. *The Annals of Statistics*, 4(2):281–291, March 1976.

[173] R. Mahler. PHD filters of higher order in target number. *IEEE Transactions on Aerospace and Electronic Systems*, 43(4):1523–1543, 2007.

[174] R. P. S. Mahler. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1152–117, 2003.

[175] D. B. Malkoff. Evaluation of the Jonker-Volgenant-Castanon (JVC) assignment algorithm for track association. In *Proceedings of Signal Processing, Sensor Fusion, and Target Recognition VI*, Orlando, FL, April 1997.

[176] T. K. Matthes. On the optimality of sequential probability ratio tests. *The Annals of Mathematical Statistics*, 34(1):18–21, March 1963.

[177] P. S. Maybeck. *Stochastic models, estimation, and control*, volume 2. Academic Press, New York, 1982.

[178] B. A. McElhoe. An assessment of the navigation and course corrections for a manned flyby of Mars or Venus. *IEEE Transactions on Aerospace and Electronic Systems*, 2(4):613–623, July 1966.

[179] R. K. Mehra. A comparison of several nonlinear filters for reentry vehicle tracking. *IEEE Transactions on Automatic Control*, 16(4):307–319, August 1971.

[180] D. Middleton and R. Esposito. Simultaneous optimum detection and estimation of signals in noise. *IEEE Transactions on Information Theory*, 14(3):434–444, May 1968.

[181] T. G. Mikhailova, I. N. Tikhonov, V. A. Mangushev, and I. V. Nikiforov. Automatic identification of overlapping of signals produced by two subsequent shocks on a seismogram. *Vulcanology and Seimology*, 5:94–102, 1990.

[182] L. Mo, X. Song, Y. Zhou, Z. K. Sun, and Y. Bar-Shalom. Unbiased converted measurements for tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3):1023–1027, 1998.

[183] K. J. Molnar and J. W. Modestino. Application of the EM algorithm for the multitarget multisensor tracking problem. *IEEE Transactions on Signal Processing*, 46(1):115–129, January 1998.

[184] J. Monahan and A. Genz. Spherical-radial integration rules for Bayesian computation. *Journal of the American Statistical Association*, 92(438):664–674, June 1997.

[185] S. Mori, C.-Y. Chong, E. Tse, and R. P. Wishner. Tracking and classifying multiple targets without a priori identification. *IEEE Transactions on Automatic Control*, 31(5):401–409, May 1986.

[186] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, March 1957.

[187] K. G. Murty. An algorithm for ranking all the assignments in order of increasing cost. *Operations Research*, 16(3):682–687, May - June 1968.

[188] D. Musicki and R. Evans. Integrated probabilistic data association-finite resolution. *Automatica*, 31(4):559–570, April 1995.

[189] D. Musicki and R. Evans. Joint integrated probabilistic data association – JIPDA. In *Proceedings of International Conference on Information Fustion*, volume 2, pages 1120–1125, November 2002.

[190] D. Musicki and R. Evans. Joint integrated probabilistic data association: JIPDA . *IEEE Transactions on Aerospace and Electronic Systems*, 40(3):1093–1099, July 2004.

[191] D. Musicki, R. Evans, and S. Stankovic. Integrated probabilistic data association. *IEEE Transactions on Automatic Control*, 39(6):1237–1241, June 1994.

[192] N. E. Nahi. Optimal recursive estimation with uncertain observation. *IEEE Transactions on Information Theory*, 15(4):457–462, July 1969.

[193] K. Nam and M.-J. Tahik. A second-order stochastic filter involving coordinate transformation. *IEEE Transactions on Automatic Control*, 44(3):603–608, March 1999.

[194] H. Niederreiter. Quasi-Monte Carlo methods and pseudo-random numbers. *Bulletin of the American Mathematical Society*, 84(6):957–1041, November 1978.

[195] M. Norgaard, N. K. Poulsen, and O. Ravn. Advances in derivative-free state estimation for nonlinear systems. Technical report, Department of Mathematical Modeling, DTU, April 2000.

[196] M. Norgaard, N. K. Poulsen, and O. Ravn. New developments in state estimation for nonlinear systems. *Automatica*, 36(11):1627–1638, Nov. 2000.

[197] S. Ohmatsu, Y. Tomita, and T. Soeda. Optimal filtering for discrete-time nonlinear systems. *IEEE Transactions on Automatic Control*, 21(1):116–118, February 1976.

[198] E. S. Page. Continuous inspection schemes. *Biometrika*, 41:100–115, 1954.

[199] E. S. Page. An improvement to Wald's approximation for some properties of sequential tests. *Journal of Royal Statistical Society*, B-16(1):136–139, 1954.

[200] H. Pesonen and R. Piche. Cubature-based Kalman filters for positioning. In *Proceedings of 7th Workshop on Positioning, Navigation and Communication*, Dresden, March 2010.

[201] G. M. Phillips. *Interpolation and Approximation by Polynomials*. Springer, 2003.

[202] J. L. Poage. A sequential nonparametric pattern classification algorithm based on the Wald SPRT. *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, 5(1):134–137, 1975.

[203] A. D. Polyanin and A. V. Manzhirov. *Handbook of Integral Equations*. Chapman & Hall/CRC Press, Boca Raton, second edition, 2008.

[204] A. B. Poore, N. Rijavec, M. Liggins, and V. Vannicola. Data association problems posed as multidimensional assignment problems: problem formulation. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, pages 552–573, Orlando, FL, USA, 1993.

[205] A. B. Poore and A. J. Robertson. A new multidimensional data association algorithms for class of multisensor multitarget tracking. In *Proceedings of Signal and Data Processing of Small Targets*, pages 448–459, San Diego, CA, USA, 1995.

[206] A. B. Poore, A. J. Robertson, and P. J. Shea. New class of Lagrangian-relaxation-based algorithms for fast data association in multiple hypothesis tracking applications. In *Proceedings of Signal Processing, Sensor Fusion, and Target Recognition IV*, pages 184–194, Orlando, FL, USA, 1995.

[207] C. Rago, P. Willett, and R. Streit. Direct data fusion using the PMHT. In *Proceedings of the American Control Conference*, volume 3, pages 1698–1702, Seattle, WA, June 1995.

[208] D. B. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, December 1979.

[209] S. R. Rogers. Diffusion analysis of track loss in clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 27(2):380–387, March 1991.

[210] R. L. Rothrock and O. E. Drummond. Performance metrics for multiple-sensor, multiple-target tracking. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, volume 4048, pages 521–531, 2000.

[211] J. Ru, V. P. Jilkov, X. R. Li, and A. Bashi. Detection of target maneuver onset. *IEEE Transactions on Aerospace and Electronic Systems*, 45(2):536–554, April 2009.

[212] Y. Ruan and P. Willett. Multiple model PMHT and its application to the benchmark radar tracking problem. *IEEE Transactions on Aerospace and Electronic Systems*, 40(4):1337–1350, October 2004.

[213] Y. Ruan, P. Willett, and R. Streit. The PMHT for maneuvering targets. In *Proceedings of the American Control Conference*, volume 4, pages 2432–2433, Philadelphia, PA, USA, June 1998.

[214] Y. Ruan and P. K. Willett. Maneuvering PMHTs. In *Proceedings of SPIE Signal and Data Processing of Small Targets*, San Diego, CA, July 2001.

[215] C. Sastry and E. Karnen. SME filter approach to multiple target tracking with radar measurements. *IEE Proceedings-F*, 140(4):251–261, 1993.

[216] T. Sathyan, S. Arulampalam, and M. Mallick. Multiple hypothesis tracking with multiframe assignment using range and range-rate measurements. In *Proceedings of 14th International Conference on Information Fusion*, Chicago, IL, July 2011.

[217] I. R. Savage and L. J. Savage. Finite stopping time and finite expected stopping time. *Journal of the Royal Statistical Society*, 27(2):284–289, 1965.

[218] T. S. Schei. A finite-difference approach to linearization in nonlinear estimation algorithms. In *Proceedings of the American Control Conference*, volume 1, pages 114–118, Seattle, WA , USA, June 1995.

[219] T. S. Schei. A finite-difference method for linearization in nonlinear estimation algorithms. *Automatica*, 33(11):2053–2058, November 1997.

[220] D. Schuhmacher, B.-T. Vo, and B.-N. Vo. A consistent metric for performance evaluation of multi-object filters. *IEEE Transactions on Signal Processing*, 56(8):3447–3457, 2008.

[221] R. G. Sea. An efficient suboptimal decision procedure for associating sensor data with stored tracks in real-time surveillance systems. In *Proceedings of IEEE Conference on Decision and Control*, pages 33–37, Miami Beach, FL, December 1971.

[222] R. G. Sea. Optimal correlation of sensor data with tracks in surveillance systems. In *Proceedings of 6th International Conference on systems Sciences*, pages 424–426, Honolulu, HI, Janurary 1973.

[223] J. Shao. *Mathematical Statistics*. Springer, New York, 2nd edition, 2003.

[224] D. Siegmund. *Sequential Analysis: Tests and Confidence Intervals*. Springer-Verlag, 1985.

[225] M. Simandl and O. Straka. Sampling densities of particle filter: a survey and comparison. In *Proceedings of the 2007 American Control Conference*, pages 4437–4442, New York, NY, July 2007.

[226] R. A. Singer and R. G. Sea. New results in optimizing surveillance system tracking and data correlation performance in dense multitarget environments. *IEEE Transactions on Automatic Control*, 18(6):571–582, December 1973.

[227] R. A. Singer, R. G. Sea, and K. B. Housewright. Derivation and evaluation of improved tracking filters for use in dense multitarget environments. *IEEE Transactions on Information Theory*, 20(4):423–432, July 1974.

[228] S. K. Singh, M. Premalatha, and G. Nair. Ellipsoidal gating for an airborne track while scan radar. In *Proceedings of the IEEE International Radar Conference*, pages 334–339, Alexandria, VA, May 1995.

[229] R. W. Sittler. An optimal data association problem in surveillance theory. *IEEE Transactions on Military Electronics*, 8(2):125–139, April 1964.

[230] I. H. Sloan and S. Joe. *Lattice Methods for Multiple Integration*. Oxford University Press, USA, 1994.

[231] G. L. Smith, S. F. Schmidt, and L. A. McGee. *Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle*. National Aeronautics and Space Administration (NASA), 1962.

[232] S. A. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Soviet Mathematics, Doklady*, 4:240–243, 1963.

[233] T. L. Song, Y. T. Lim, and D. G. Lee. A probabilistic strongest neighbor filter algorithm for $m$ validated measurements. *IEEE Transactions on Aerospace and Electronic Systems*, 45(2):431–442, April 2009.

[234] H. W. Sorenson and A. R. Stubberud. Recursive filtering for systems with small but non-negligible non-linearities. *International Journal of Control*, 7(3):271–280, 1968.

[235] R. F. Stengel. *Optimal Control and Estimation*. Dover, New York, 1994.

[236] R. L. Streit and T. E. Luginbuhl. Maximum likelihood method for probabilistic multihypothesis tracking. In *Proceedings of Signal and Data Processing of Small Targets*, volume 2335-24, Orlando FL, April 1994.

[237] R. L. Streit and T. E. Luginbuhl. Probabilistic multi-hypothesis tracking. Technical report, NUWC-NPT Technical Report 10428, Naval Undersea Walfare Center, Newport, RI, February 1995.

[238] A. H. Stroud. *Approximate Caculation of Multiple Integrals*. Prentice Hall, Englewood Cliffs, NJ, USA, 1971.

[239] P. Suchomski. Explicit expressions for debiased statistics of 3D converted measurements. *IEEE Transactions on Aerospace and Electronic Systems*, 35(1):368–370, 1999.

[240] A. Tartakovsky. Asymptotically optimal sequential tests for nonhomogeneous processes. *Sequential Analysis*, 17(1):33–61, 1998.

[241] S. Vardeman and D. Ray. Average run lengths for CUSUM schemes when observations are exponentially distributed. *Technometrics*, 27(2):145–150, May 1985.

[242] P. K. Varshney and A. H. Haddad. A receiver with memory for fading channel. *IEEE Transactions on Communications*, 26(2):278–283, February 1978.

[243] A. Wald. Sequential tests of statistical hypotheses. *Annals of Mathematical Statistics*, 16:117–186, 1945.

[244] A. Wald. *Sequential Analysis*. Wiley, New York, 1947.

[245] A. Wald and J. Wolfowitz. Optimum character of the sequential probability ratio test. *Annals of Mathematical Statistics*, 19:326–339, 1948.

[246] L. Weiss. Testing one simple hypothesis against another. *The Annals of Mathematical Statistics*, 24(2):273–281, June 1953.

[247] P. Willett, Y. Ruan, and R. Streit. PMHT: problems and some solutions. *IEEE Transactions on Aerospace and Electronic Systems*, 38(3):738–754, July 2002.

[248] R. P. Wishner, J. A. Tabaczynski, and M. Athans. A comparison of three non-linear filters. *Automatica*, 5(4):487–496, July 1969.

[249] P. J. Wong and A. J. Korsak. Reachable sets for tracking. *Operations Research*, 22:497–508, May–June 1974.

[250] M. Yang. *When Decision Meets Estimation: theory and applications*. PhD thesis, University of New Orleans, December 2007.

[251] J. R. V. Zandt. A more robust unscented transform. In *Proceedings of SPIE*, volume 4473, 2001.

[252] Z. Zhao, X. R. Li, and V. P. Jilkov. Best linear unbiased filtering with nonlinear measurements for target tracking. *IEEE Transactions on Aerospace and electronic systems*, 40(4):1324–1336, October 2004.

[253] J. Zhu, N. Zheng, Z. Yuan, Q. Zhang, X. Zhang, and Y. He. A SLAM algorithm based on the central difference Kalman filter. In *IEEE Intelligent Vehicles Symposium*, 2009.

# Vita

Yu Liu received the B.S. and M.S. degrees from Xi'an Jiaotong University, Xi'an, China, in 2006 and 2008, respectively, both in Electrical Engineering. From Aug. 2008 to Dec. 2013, he is a research assistant in the department of Electrical Engineering, University of New Orleans, New Orleans, LA, USA. His research interests include estimation, decision, sequential analysis and their applications to target tracking and navigation. He has authored or coauthored more than 10 journal and conference papers.