8-10-2005

Adaptive Estimation and Detection Techniques with Applications

Jifeng Ru
University of New Orleans

ADAPTIVE ESTIMATION AND DETECTION TECHNIQUES WITH APPLICATIONS

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

Jifeng Ru

B.S., Hefei University of Technology, 1996
M.S., Hefei University of Technology, 1999

August 2005

Acknowledgment

I would like to express my deepest gratitude to my major advisor, Dr. X. Rong Li, for his invaluable inspiration, continuous support, and constructive suggestions, which made it possible for me to complete the Ph.D. dissertation. During my graduate study, I gained considerable knowledge and insights into problems and techniques in estimation and decision theory, statistical signal processing and information fusion under Dr. Li's guidance. All academic accomplishments I have made during this period are reflection of his efforts and motivations.

I would also thank Dr. Huimin Chen, Dr. Juliette Ioup, Dr. Vesselin Jilkov, and Dr. Tumulesh K.S. Solanky for serving on my thesis committee, and for their constructive and valuable comments on the dissertation. Special thanks go to Dr. Jilkov and Dr. Chen for their long-term collaboration, fruitful discussions and insightful ideas. Many thanks go to Dr. Ioup for her suggestions and constant support.

I also want to acknowledge all my friends and members in the Information and Systems Laboratory, especially Ming Yang, Keshu Zhang, Zhanlue Zhao, Anwer Bash, Peng Zhang, Ning Li, Lei Lu, Trang Nguyen, Ryan Pitre and Victor Alvarado. With their support and friendship, I had the great pleasure of carrying out research in such a pleasant working environment. Furthermore, I would like to thank all the members and staff of the Department of Electrical Engineering at the University of New Orleans for their support.

This research was supported in part by NSF grant ECS-9734285, NASA/LEQSF grant (2001-4)-01, ARO grant W911NF-04-1-0274, SBIR N04-241 through Naval Airwarfare Center, and High-Performance Networking Program of the Office of Science, U. S. Department of Energy under Contract DE-AC05-00OR22725 with UT-Battelle, LLC.

Last but not least at all, I want to thank my parents for their unconditional, unending support and love. I dedicate this thesis to them.

Contents

List of Figures

List of Tables

Abstract

Hybrid systems have been identified as one of the main directions in control theory and attracted increasing attention in recent years due to their huge diversity of engineering applications. Multiple-model (MM) estimation is the state-of-the-art approach to many hybrid estimation problems. Existing MM methods with fixed structure usually perform well for problems that can be handled by a small set of models. However, their performance is limited when the required number of models to achieve a satisfactory accuracy is large due to time evolution of the true mode over a large continuous space. In this research, variable-structure multiple model (VSMM) estimation was investigated, further developed and evaluated. A fundamental solution for on-line adaptation of model sets was developed as well as several VSMM algorithms. These algorithms have been successfully applied to the fields of fault detection and identification as well as target tracking in this thesis. In particular, an integrated framework to detect, identify and estimate failures is developed based on the VSMM. It can handle sequential failures and multiple failures by sensors or actuators.

Fault detection and target maneuver detection can be formulated as change-point detection problems in statistics. It is of great importance to have the quickest detection of such mode changes in a hybrid system. Traditional maneuver detectors based on simplistic models are not optimal and are computationally demanding due to the requirement of batch processing. In this presentation, a general sequential testing procedure is proposed for maneuver detection based on advanced sequential tests. It uses a likelihood marginalization technique to cope with the difficulty that the target accelerations are unknown. The approach essentially utilizes a priori information about the accelerations in typical tracking engagements and thus allows improved detection performance. The proposed approach is applicable to change-point detection problems under similar formulation, such as fault detection.

Chapter 1

Introduction

1.1 Background

Hybrid systems are complex systems whose states exhibit both continuous and discrete dynamics. This combination and interaction of the discrete and continuous nature make this emerging area a particularly challenging field of research. Hybrid systems have been identified as one of the main directions in control theory and attracted increasing attention in recent years due to their huge diversity in engineering applications, including system fault detection, air traffic control, industrial process control, target tracking and communication networks, etc.

Estimation of the states in a hybrid system subject to structure or parametric uncertainty is known as hybrid estimation. The conventional solutions to hybrid estimation are decision-based, where the states are estimated after a decision is made. Specifically, a hypothesis test is first performed to determine the structure currently in effect and then estimation techniques are applied as if the selected structure were the true one. The decision-based estimation has been extensively studied in many areas, and an excellent review of the existing approaches in the area of target tracking can be found in [50]. Even though the decision-based approaches have been successfully applied in various applications of hybrid estimation, this type of solution has clear drawbacks. Since estimation is employed based on a decision, any possible decision errors on the models are not accounted for in the estimation. Moreover, a hard decision is taken irrevocably before estimation even though estimation results are often beneficial to decision making [52].

In recent years, the multiple model approach has become the mainstream approach for hybrid estimation. In theory, there are certain advantages to use the multiple model approach. First, for complex systems such as flight control systems, it is often impractical to represent the whole system using a single model. Second, multiple model representations provide an easy way to incorporate system information from different sources [91].

The basic idea of the *multiple model (MM)* estimation is to assume a set of models describing a hybrid system. It consists of a bank of elemental filters running in parallel, each based on a particular model representing a possible system behavior or structure, to obtain model-conditional estimates; the overall estimate is a certain combination of these model-conditional estimates; the jumps in the system modes can be modeled as switches/transitions between the assumed models. Due to these unique features, the MM estimation fits well in the framework of hybrid estimation and has great success in various fields including target tracking, air traffic control, fault detection and diagnosis, and communications, etc. [42, 52].

The MM method was initiated in [62] and is now the state-of-the-art approach for many estimation problems. It has been developed through three generations as identified in [43]. According to the structure of the model set used in the MM estimation, three generations first two generations, the same set of models is used at all times and thus referred to as a *fixed structure-based MM (FSMM)* estimation. The third generation allows a variable set of models adapting to data, leading to a *variable structure-based MM (VSMM)* estimation.

The existing MM methods with fixed structure usually perform well for problems that can be handled with a small set of models. Consequently, these methods have a great success in solving many estimation problems involving structural as well as parametric uncertainty, particularly in target tracking. However, due to the fact that the true system mode is often unknown and/or time varying over a large space, an algorithm using a fixed set of a small number of models cannot yield accurate results. Apart from the dramatic increase in computation, the research in [43, 48] shows that use of more models in a fixed structure does not guarantee a performance improvement. To find a way to overcome this dilemma, the concept and structure of the variable structure MM estimation were introduced in [43, 44, 48, 49, 54], particularly for target tracking, fault detection and identification.

The VSMM is potentially much more advanced than the fixed-structure MM estimation: besides inheriting the first two generations' superior processing capabilities, it adapts to the real environment by augmenting new models or eliminating some existing models according to applications. This built-in learning mechanism for the model set in the VSMM leads to an improved performance over the FSMM estimator. The VSMM estimation is especially powerful for the case when the model set used does not match the possible set of the true system mode. Due to its open structure and successful applications, the VSMM is gaining momentum rapidly [42, 43, 44, 55, 59, 86, 87].

Fault detection and diagnosis (FDD) and maneuvering target tracking (MTT) are two typical applications of hybrid estimation due to their common features: both applications involve continuous-valued parameter estimation such as system states, and discrete hypotheses decision (e.g., possible sensor or actuator failures in FDD, and different target motions such as constant velocity or constant accelerations in MTT). Thus, it is natural to pose such problems as one of hybrid estimation.

FDD has been a major issue for modern engineering systems which requires reliability, availability and security with increasing complexity. In the last two decades, many FDD techniques have been developed that include hardware redundancy and analytical redundancy based approaches, as surveyed in [9, 23, 26, 29, 99]. However, conventional fault detection and identification (FDI) algorithms do not deal with failure estimation and state estimation simultaneously. On the other hand, maneuvering target tracking is a challenging research topic since target maneuvers are usually unknown. Most of the target tracking algorithms are hybrid estimation. Therefore, improvement in hybrid estimation, especially variable-structure based MM estimation, is beneficial to applications of target tracking, fault detection and identification.

For a hybrid system, since the true system mode jumps due to structure or parametric change, it is desirable to have a quickest detection, which will provide useful information for the subsequent state estimation; for example, decision-based approaches for target tracking or fault tolerant control. Assume that a measurement sequence before and after an unknown change time has different probability distributions. The objective is to detect the occurrence of the change as soon as possible under some constraints. This type of problem is usually formulated as a binary hypothesis testing, known as change point detection in the statistical literature. Change point detection has been extensively studied in statistics and engineering.

According to the sample size, two types of approaches are usually used to detect changes: fixed-size batch detection including Schewhart's control chart, geometric moving average, and cumulative sum control chart; and sequential detection based on sequential probability ratio test (SPRT) such as Page's test and Shiryayev SPRT [10, 33, 35, 34]. Due to unknown maneuver accelerations, target maneuver onset detection can be treated as a typical application of change point detection.

Many algorithms and techniques have been developed to detect target maneuvers [82], which can be categorized into two classes: chi-square based and likelihood ratio based tests. Such detectors are based on simple models and batch processing, and are thus non-optimal and computationally demanding. Moreover, for tracking applications measurements are usually available in a sequential manner. As such, it motivates us to investigate the sequential processing of maneuver detection based on advanced statistical tests. Clearly, this study could be generalized to other applications under the same formulation such as system fault detection.

1.2 Research Objectives

The main objective of this research is to formulate problems and develop new algorithms with regard to hybrid estimation and change point detection for the detection of target maneuvers and system faults; more specifically, to develop novel variable-structure algorithms for multiple model hybrid estimation and apply proposed techniques to target tracking, fault detection and identification. In essence, a general approach is preferred that adjusts the model set in real time based on data to cover a large continuous mode space by a relatively small number of models with a desired accuracy level. The proposed approach is expected to overcome the limitations of the fixed-structure MM estimation and facilitates various applications of hybrid estimation by providing high cost-effective and robust algorithms.

- The first objective is to present a general variable-structure MM scheme referred to as expected-mode augmentation (EMA), originally proposed in [49], to enhance the EMA algorithm with its practical implementation, and to evaluate and analyze its performance via a generic maneuvering target tracking problem.

- The second objective is to develop an integrated framework to detect, identify, and estimate failures, including abrupt total, partial and multiple failures, in a dynamic system. Meanwhile, the proposed approach should provide accurate state estimation even during failures. This proposed framework based on VSMM is to overcome the limitations of the existing FDD algorithms.

- The third objective is to investigate the existing target maneuver detection algorithms, to formulate the problem in a sequential hypothesis setting, and to develop new sequential detection algorithms based on advanced sequential tests to improve the detection performance, and consequently the tracking performance as well.

1.3 Thesis Outline

This thesis contains seven chapters and five appendices that are organized as below:

Chapter 1 presents the motivation and objectives of this research work.

Chapter 2 reviews several aspects of the hybrid system, such as hybrid estimation, structure and development of multiple model estimation, in particular, the interacting multiple model estimator.

Chapter 3 presents a new class of variable-structure algorithms for multiple-model estimation referred to as the expected-mode augmentation (EMA). In the EMA approach, the original set of fixed models is augmented by a variable set of models intended to match the expected value of an unknown true mode. These models are generated adaptively in real time as (globally or locally) probabilistically weighted sums of modal states over the model set. General formulation, theoretical analysis and justification of the EMA approach are presented along with three algorithms for its practical implementation. The performances of the proposed EMA algorithms are evaluated via simulation of a generic maneuvering target tracking problem.

Chapter 4 proposes two schemes for failure detection, identification and estimation, including abrupt total, partial and multiple failures, in a dynamic system. The proposed algorithms are based on the variable-structure multiple model estimation, which improves performance due to online adaptation. Using two aircraft examples, the proposed approaches are evaluated and compared with a widely used single-model residual based generalized likelihood ratio (GLR) approach in terms of detection and estimation performance as well as robustness in the presence of the uncertain noise statistics. Model set design issues are also discussed along with conclusions and further discussions.

Chapter 5 introduces background information for the change point detection problem followed by several key sequential detection algorithms.

Chapter 6 addresses target maneuver onset detection based on sequential statistical tests. Cumulative sums (CUSUM) test and Shiryayev sequential probability ratio test (SSPRT) are developed by using a likelihood marginalization technique to cope with the difficulty of unknown target maneuver accelerations. The approach essentially utilizes a priori information about the maneuver accelerations in typical tracking engagements and thus allows improved detection performance as compared with traditional maneuver detectors. Simulation results are presented to demonstrate the developed capabilities of the maneuver detectors. The feasibility of applying proposed sequential algorithms to fault detection is illustrated at the end.

Chapter 7 draws major conclusions from the research work and provides some further research directions.

Chapter 2

Hybrid Estimation

2.1 Hybrid System

A class of simple discrete-time hybrid systems is described by

$$x_{k+1} = F_k(s_{k+1})x_k + G_k(s_{k+1})w_k(s_{k+1}) \tag{2.1}$$

$$z_{k+1} = H_k(s_k)x_{k+1} + v_k(s_k) \tag{2.2}$$

where x is the base state, s is the mode or modal state, z is the measurement, w and v are independent process and measurement noise. The state in a hybrid system $\xi = [x', s']'$ is referred to as a *hybrid state*. Clearly such a system is not linear since x or z does not depend on the system state ξ in a linear fashion. The system could be deemed as linear if s is given. For actual systems s could jump at unknown time instants. Jumps between different modes are used to model abrupt system changes. System (2.1)-(2.2) is known as a *Markov jump linear system* if s is a Markov chain

$$P\{s_{k+1} = s_j | s_k = s_i\} = p_{ij}, \qquad \forall i, j, k$$

Often s is assumed a homogeneous Markov chain, that is, p_{ij} is a constant for all time k.

Base state x is a continuous-valued variable, often referred to as a state variable in a conventional system, such as position and velocity of a moving object. Mode state s is a discrete-valued variable for mathematical characterization of a certain behavior pattern or structure of the system. For example, in the context of fault detection, a normal mode corresponds to the normal operation of a system. A fault model can be used to represent a certain failure/degradation in some part of the system. In the context of air traffic control, the mode of straight and level motion corresponds to the constant velocity motion while maneuvering modes can be applied to turning or accelerating motions [42]. Different system modes could be described by different equations with known or unknown parameters.

Such systems (2.1)-(2.2) can be used to model situations whose system behavior pattern undergoes sudden changes, such as system failures and target maneuvers. It provides a framework particularly suitable for problems with structural as well as parametric uncertainties.

2.2 Hybrid Estimation

In hybrid systems, state estimation subject to structural/parametric uncertainty is called *hybrid estimation* in the sense that it combines state estimation and parameter estimation to deal with simultaneously continuous- and discrete-valued uncertainties. The problem of hybrid estimation is to estimate the base state and model state based on the sequence of noisy state measurements along with prior information.

The conventional solution to hybrid estimation is decision-based approaches, where the state is estimated after a decision is made. Consequently, the decision errors on models or possible contributions of estimation to decisions are ignored. Although the limitation is clear for decision-based approaches, it is still very difficult to come up with effective remedies to overcome their drawbacks within this framework. Currently, the mainstream approach to hybrid estimation is the *multiple-model (MM) approach*, which uses multiple models and each model represents a possible system behavior or structure. As a result, the multiple model approach overcomes the difficulty of model uncertainty that conventional decision-based approaches have to face. The MM approach provides a natural solution for hybrid estimation.

2.3 Multiple Model Estimation

The basic idea of the *multiple model estimation* approach is to assume a set of models as possible candidates of the true mode in the hybrid system. Unique features of MM estimation include 1) a bank of elemental filters run in parallel to obtain model-conditional estimates; 2) the overall estimate is fused by these model-conditional estimates; 3) transitions between models are used to model jumps in system mode.

For a Markov jump linear system, the ith model in the MM method is represented by

$$x_{k+1} = F_k^i x_k + G_k^i u_k + T_k^i w_k^i \tag{2.3}$$

$$z_k = H_k^i x_k + v_k^i \tag{2.4}$$

where superscript i denotes quantities pertinent to model m^i in model set M, and the jumps of the system mode are assumed to have the following transition probabilities

$$\pi^{ij} = P\{m_{k+1}^j | m_k^i\} \tag{2.5}$$

where m_k^i denotes that the ith model is in effect at k.

2.3.1 Structure of MM Algorithms

The operation of MM estimation for hybrid systems is depicted in Fig. 2.1 with only two models. In general, the application of MM estimation consists of the following steps:

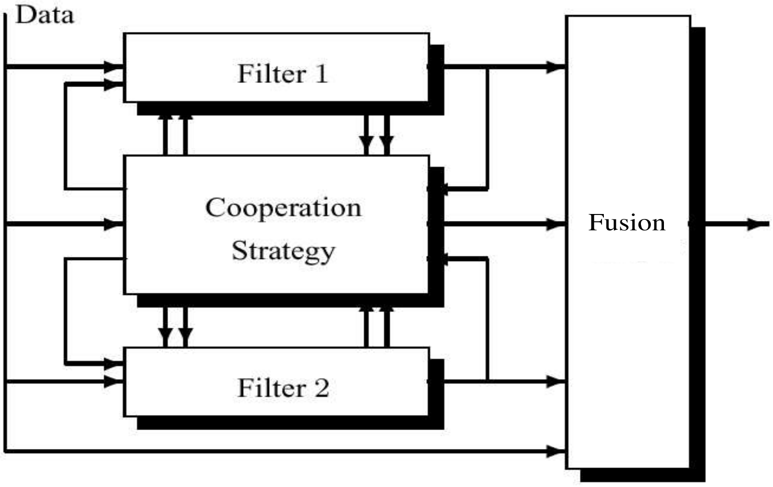


Figure 2.1: General structure of MM estimation algorithms

- Model-set design: A major difficulty in the application of MM estimation. It is in general application dependent and can be done offline or through online adaptation. Detailed discussions can be found in [52]. The main task in model set design is to select or construct a set of models possibly covering the system mode space. The performance of an MM estimator largely depends on the designed model set, especially for problems involving a large number of modes.

- Filter selection: The single-model based filter for each model has to be selected, such as Kalman filter for a linear problem, an extended Kalman filter for nonlinear estimation, or a nonlinear filter. Filters based on different models can be of different types. This step relies on classical estimation theory based on the problem under investigation.

- Cooperation strategy: A main research focus for MM estimation. All possible cooperative actions among filters are determined to achieve better performance, such as pruning of unlikely model sequences, merging of similar model sequences, individualized reconditioning of each filter (e.g., Interacting Multiple Model (IMM) algorithm), or iterative strategies (like Expectation-Maximization (EM) based algorithms).

- Estimate fusion: This step determines the procedure to combine the individual model-conditional estimate to yield the overall estimate. It can be achieved either by a procedure based on a hard decision (i.e., select the estimate from the most likely or at least not unlikely) or a soft decision (e.g. weighted sum of estimates from every filter).

2.3.2 Development of MM Algorithms

The MM method, state-of-the-art approach for many estimation problems, has been developed into three generations [52]:

- The first generation, Autonomous MM (AMM) estimation, was initiated in [61, 36, 37] and widely applied in [65, 66, 67, 72, 96, 22]. In the first generation, each of its elemental filters operates independently without any interaction with each another. Its advantage over many non-MM approaches stems from its superior output processing: the overall estimate is generated by fusing the estimate from elemental filters. Due to the underlying assumption that the mode does not jump, AMM is not suitable for problems with frequent changes in system behavior. Instead, it is particularly popular for problems involving unknown parameters.

- The second generation inherits the first generation's power of output processing, and its elemental filters work together as a team via effective internal cooperation. Generalized Pseudo Bayesian of order n (GPBn) and especially Interacting Multiple-Model (IMM) are popular algorithms in the second generation. Especially, IMM with its further development has been successfully applied to a significant number of applications [15, 14, 16, 2, 3, 6, 7, 8]. There exist many cooperation techniques, such as reinitialization in IMM, iterative iterations for performance enhancement in EM based algorithms, and other hypothesis reduction strategy.

The model groups in the first two generations have a *fixed* membership over time and thus have a fixed structure.

- The third generation allows a variable membership, that is, a *variable* set of models. This generation has been known as *variable-structure MM* (VSMM) method. It is most suitable in the case where the model set used does not match the set of possible true modes. The third generation was initiated in [41, 45, 47] and advances have been further continued in [42, 43, 44, 55, 59, 86, 87, 49, 54, 53].

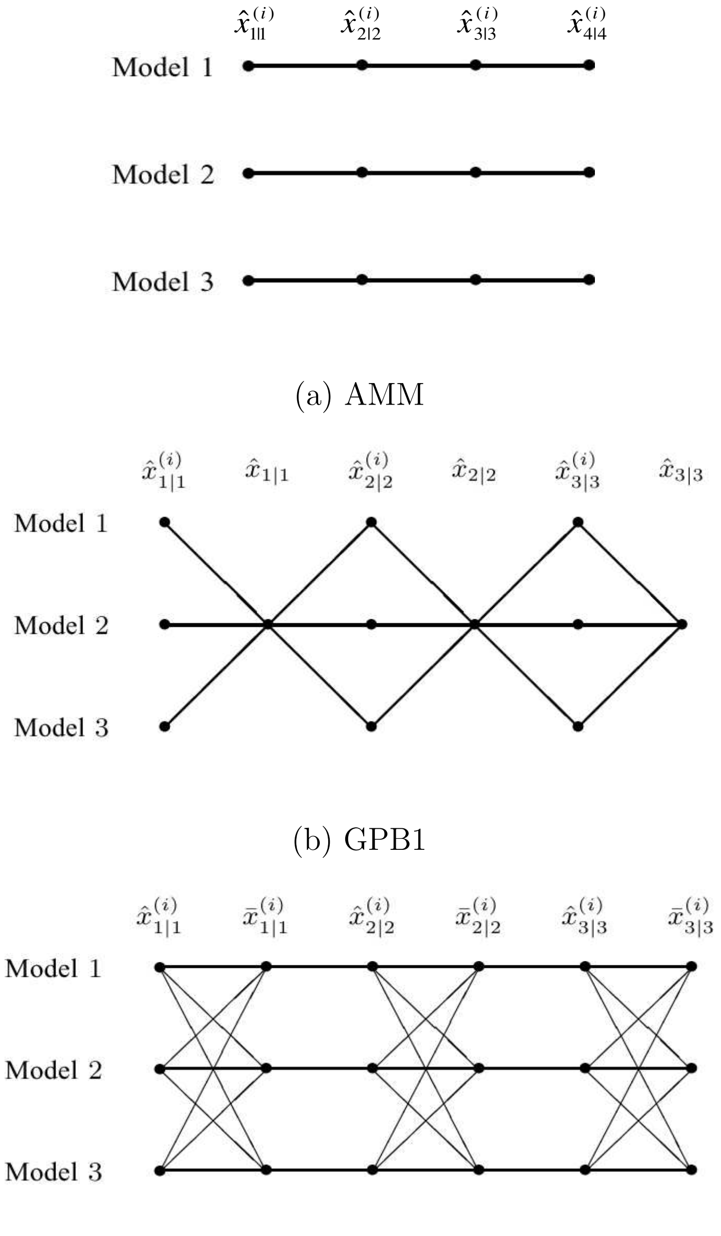
The first two generations mainly differ in the *reinitialization* of each elemental filer in the cooperation strategy. Fig. 2.2 illustrates the difference for three typical algorithms. In the AMM, every elemental filter runs individually without interactions with each another. The

GPB1 algorithm reinitializes each filter with the previous overall estimate, which carries information from all filters and thus is the “best possible” common single quasi-sufficient statistic. In the IMM, each filter has its own reinitialization $\bar{x}^i_{k-1|k-1}$ (and $\bar{P}^i_{k-1|k-1}$), which forms the best quasi-sufficient statistic of all old information and the knowledge/assumption that model m^i_k matches the system mode at k. The superiority of mixing processing in the IMM reinitialization has been evidenced by numerous applications [52]. Details of such algorithms can be found in [6, 7, 8, 52] and references therein.

2.3.3 Interacting Multiple Model Algorithm

The IMM algorithm, originally proposed in [62], has been the main-stream MM algorithm due to its cost-effective performance and simple scheme demonstrated by a significant number of successful applications for hybrid estimation. An IMM estimator consists of a bank of filters, each based on a model matching a particular mode of system; information is utilized via interaction among filters. The main feature of this algorithm relies on its nature that different system behavior modes can switch from one to another. This improves its ability to estimate the state of a dynamic system. The IMM algorithm has three desirable properties: it is recursive, model conditioned, and has fixed computational load for each cycle [70]. As aforementioned, compared with other MM algorithms, the superiority of the IMM stems from the smart individualized reinitialization scheme, which leads to improved performance.

The structure of the IMM estimation algorithm is illustrated in Fig. 2.3 with three models [52]. It comprises four major steps in each recursive cycle:



(a) AMM

(b) GPB1

(c) IMM

Figure 2.2: Filter initializations for AMM, GPB1 and IMM algorithms

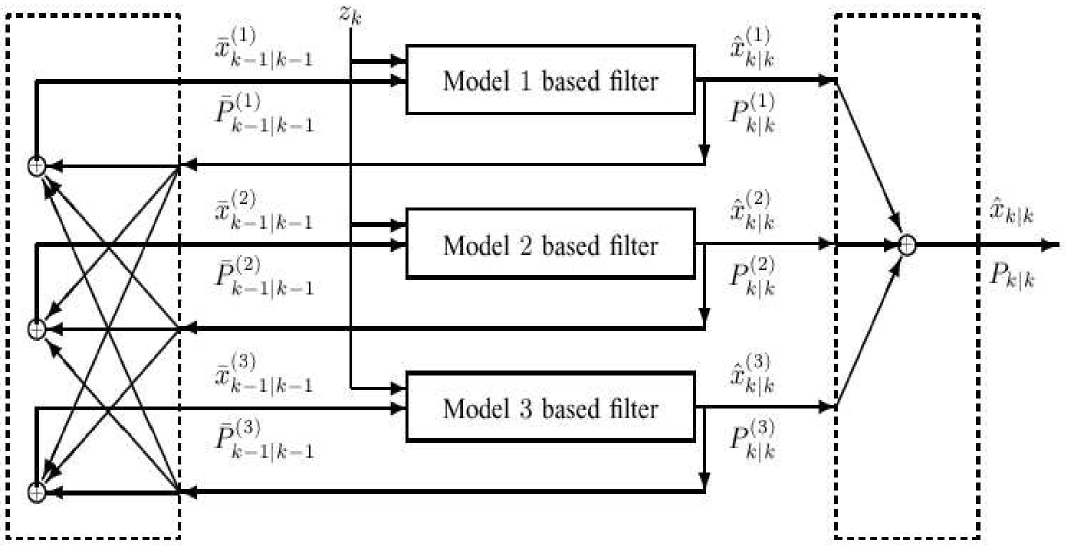


Figure 2.3: The structure of the IMM estimation algorithm with three models

1. Model-conditional re-initialization: A filter input matching the corresponding mode is obtained through a mixture of all filter estimates at the previous time, assuming that this particular mode is in effect at the present time. This step is unique for the IMM estimator compared with other MM algorithms.

Predicted model probability $\mu^i_{k|k-1} = \sum\limits_{m_j \in M_{k-1}} \pi_{ji}\mu^j_{k-1}$

Mixing weight $\mu^{j|i} = \pi_{ji}\mu^j_{k-1}/\hat{\mu}^i_{k|k-1}$

Mixing estimate $\bar{x}^i = \sum\limits_{m_j \in M_{k-1}} \hat{x}^j_{k-1|k-1}\hat{\mu}^{j|i}_{k-1}$

Mixing covariance $\bar{P}^i = \sum\limits_{m_j \in M_{k-1}} [P^j_{k-1|k-1} + (\bar{x}^i_{k-1|k-1} - \hat{x}^j_{k-1|k-1})(\bar{x}^i_{k-1|k-1} - \hat{x}^j_{k-1|k-1})']\mu^{j|i}_{k-1}$

2. Model-conditional filtering: This step performs regular filtering including estimation prediction and update, in a parallel structure for each filter.

Predicted state $\hat{x}^i_{k|k-1} = F^i_{k-1}\bar{x}^i_{k-1|k-1} + G^i_{k-1}\bar{w}^i_{k-1}$

Predicted covariance $P^i_{k|k-1} = F^i_{k-1}\bar{P}^i_{k-1|k-1}(F^i_{k-1})' + G^i_{k-1}Q^i_{k-1}(G^i_{k-1})'$

Measurement residual $\tilde{z}^i_k = z_k - H^i_{k-1}\hat{x}^i_{k|k-1} - \bar{v}^i_{k-1}$

Residual covariance $S_k^i = H_k^i P_{k|k-1}^i (H_k^i)' + R_k^i$

Filter gain $K_k^i = P_{k|k-1}^i (H_k^i)' (S_k^i)^{-1}$

Updated state $\hat{x}_{k|k}^i = \hat{x}_{k|k-1}^i + K_k^i \tilde{z}_k^i$

Updated covariance $P_{k|k}^i = P_{k|k-1}^i - K_k^i S_k^i (K_k^i)'$

3. Mode probability update based on the model-conditional likelihood functions. The model probability plays a key role in the weighting of the state mixing and the fusion.

Model likelihood $L_k^i = p[\tilde{z}_k^i \mid m_k^i, z^k] = \frac{\exp(-(1/2)(\tilde{z}_k^i)'(S_k^i)^{-1}(\tilde{z}_k^i))}{\left|2\pi S_k^i\right|^{1/2}}$

Model probability $\mu_k^i = p[m_k^i \mid z^k] = \frac{\mu_{k|k-1}^i L_k^i}{\sum\limits_j \mu_{k|k-1}^j L_k^i}$

4. Estimate combination: This yields a total state estimate as the probabilistically weighted sum of the updated state estimates of all filters.

Overall estimate $\hat{x}_{k|k} = E[x_k \mid z^k] = \sum\limits_i \hat{x}_{k|k}^i \mu_k^i$

Overall covariance $P_{k|k} = \sum\limits_i [P_{k|k}^i + (\hat{x}_{k|k} - \hat{x}_{k|k}^i)(\hat{x}_{k|k} - \hat{x}_{k|k}^i)']\mu_k^i$

2.4 Variable-Structure Multiple Model Method

The existing MM methods with the fixed structure cannot handel the system well if its true mode varies over a large space with time or changes frequently. The major reason for the poor performance of the existing FSMM estimators with a large model set is that many models in the set are very different from the true system mode in effect at a particular time, thus the unnecessary “competition” among models degrades the performance [42]. Due to the nature of the FSMM, it is hard to expect great improvement although further development is certainly possible, such as the design of a better set of models or the development of better implementable versions of the optimal FSMM estimator. The variable-structure based MM estimation was initiated in [41, 48] to overcome the fundamental limitations of the fixed structure MM. The continued research in [42, 43, 44, 55, 59, 86, 87, 49, 54, 53] lays down the theoretical foundation for hybrid estimation with a variable structure.

The basic idea of the VSMM is to adapt the model set online based on all available information, in particular, the sequence of the measurements. The real-time measurements carry valuable information about the system mode being in effect, and thus provide useful information about the model set.

2.4.1 Structure of VSMM

In VSMM a set of possible models varies with time by *online adaptation*. In general, each VSMM algorithm has two tasks:

- **Model set adaptation** determines at each time the model set to use for the MM estimation, utilizing posterior information as well as prior knowledge. This is unique and the most important topic in VSMM estimation. Existing approaches in model set adaptation can be classified as model-set reduction and model-set augmentation. It has been shown that combined model-set reduction and augmentation has significant advantages over model-set switching in terms of tractability, performance, and generality.

- **Model set sequence conditioned estimation** intends to provide best possible estimates given a model-set sequence, including filter initialization to new models and cooperation strategies and conditional filtering. The second step is similar to the FSMM.

Depending on whether the designed total model set can be specified in advance [52], existing model-set adaptation algorithms for the VSMM can be grouped into two broad families: *active model-sets* and *model-set generation*. In the active model-set family, the total model set is determined in advance, and a subset is activated adaptively at any given time. Clearly, the main task for this class of structures is the design of the model subsets, determination of the candidate subsets, and the decision procedure for switching. Model-group switching (MGS) is one of the classes with this structure, where the active set is determined by switching among a number of predetermined subsets of the total model set [59]. The switching can be done through a soft or hard decision. The likely-model set (LMS) is another simple class of active model-set structure, where the active set is formed by deleting the models unlikely to match the true mode at a given time [55]. Another class is based on a hierarchical structure, where the active set consists of hierarchical levels of models [18, 86]. The model subset at a lower level is activated under the guidance of the higher level.

In the model set generation, new models are generated in real time and thus the model set cannot be specified in advance. Estimated-mode augmentation is a class for model-set generation, where the original set is augmented by one or more models that matches an estimate of the true mode at a given time. The augmented mode can be estimated under different optimality criteria in principle. Another class is called adaptive grid structure, where the mode space is quantized unevenly and adaptively based on data as well as prior information [25, 48, 68]. More details of existing algorithms with adaptive structure can be found in [52].

Chapter 3

Variable-Structure MM Estimation with Application in Maneuvering Target Tracking

3.1 Introduction

In the past, considerable research on multiple model estimation has been undertaken in the field of hybrid estimation. It is of interest in both military and civilian applications. The advantage of MM estimation is rooted in the fact that the behavior of a hybrid system cannot be characterized by a single model for all time, instead a finite number of models may be adequate to describe the system. Based on the model set, a bank of filters is run in parallel to obtain model-conditioned estimates, which is used to generate the overall estimates through certain fusion techniques.

As is known, the performance of an MM algorithm depends highly on the model set used. Naturally, if the models in the model set are closer to the true mode of the system, the performance would be better. In many MM applications, the set of possible values of uncertain system parameters, known as mode space, is continuous. However in reality only a limited number of models can be used. The common practice is to design a finite set of models to approximate this mode space. Loosely speaking, the major objective here is to achieve the best modeling accuracy with a minimum number of models, which is still an open problem in a general setting, although significant progress has been reported in [45, 58].

To capture various possible unknown mode jumps and in the meantime to have at least one model close to the true mode, a natural idea is to augment the original model set M by one or more adaptive models that follow closely the true mode, leading to the so-called *estimated-mode augmentation*. Obtaining good candidates for the augmenting models is the basic idea of estimated-mode augmentation structure. A good candidate for the augmenting models is the expected value of the true mode since it is statistically closest to the true mode. This expected mode can be approximated by a sum of modal states weighted by the corresponding model probabilities, readily available from the underlying MM algorithm. This expected-mode augmentation (EMA) approach, introduced originally in [49] and furthered in [54], is systematic and general for all problems with a continuous mode space.

Several researchers have considered similar problems and proposed their solution techniques. For a static MM algorithm, [25] used an initial coarse grid and a subsequent fine grid while [68] presented a filter bank that moves over a predefined fixed grid according to a decision logic. It was proposed in [73] to use a moving set of acceleration models centered around a model whose acceleration is determined by an additional Kalman filter. In [48], it was suggested to employ the expected mode as the center of an adaptive grid for an example of nonstationary noise identification. In [38], an adaptive IMM algorithm for maneuvering target tracking was proposed that uses an acceleration model determined by a separate Kalman filter on top of a fixed set of models. Compared with these existing techniques, not only is the EMA approach much more general and systematic, but it is also highly cost-effective and easy to implement.

This chapter is organized as follows. In Section 1, the EMA approach is formulated in a general setting with its theoretical analysis and justification. Three practical baseline EMA algorithms are proposed in Section 2. Section 3 develops several EMA designs for maneuvering target tracking. Performance evaluation and comparison of the proposed EMA algorithms are presented in Section 4 via simulation of a generic maneuvering target tracking problem. Section 5 provides a conclusion.

3.2 Expected-Mode Augmentation (EMA)

3.2.1 Benefit of Model-Set Augmentation

Denote by s the true mode and by S the mode space (i.e., the set of possible values of s). Consider the problem of adding a model set C to the original model set M (hence $C \cap M = \varnothing$). Assume $(M \cup C) \subset S$. Let

$$\hat{x}_M = E[x|s \in M, z], \quad \hat{x}_C = E[x|s \in C, z]$$

$$\mu_M = P\{s \in M|s \in (M \cup C), z\}, \quad \mu_C = P\{s \in C|s \in (M \cup C), z\}$$

where z stands for measurements. Then according to the total probability theorem, the estimator of x based on the union of model sets M and C is

$$\hat{x} = E[x|s \in (M \cup C), z] = \mu_M \hat{x}_M + \mu_C \hat{x}_C$$

which is a convex combination of $\hat{x}_M$ and $\hat{x}_C$. Then, following the derivation in Appendix A, we have

Lemma If $\hat{x}_M$ and $\hat{x}_C$ are unbiased with uncorrelated estimation errors, which can be assumed in most cases, use of the union of M and C is better than use of M alone if and only if

$$\mu_C < \frac{2\text{mse}(\hat{x}_M)}{\text{mse}(\hat{x}_M) + \text{mse}(\hat{x}_C)}$$

where $\text{mse}(\hat{x}_M)$ and $\text{mse}(\hat{x}_c)$ stand for the mean-square error of $\hat{x}_M$ and $\hat{x}_C$, respectively.

This inequality is always satisfied if $\hat{x}_C$ is better than $\hat{x}_M$. Even if $\hat{x}_C$ is worse than $\hat{x}_M$, $\hat{x}$ is still better than $\hat{x}_M$ provided μ_C satisfies the above inequality. If $\hat{x}_M$ and $\hat{x}_C$ have correlated estimation errors (i.e., $E[\tilde{x}'_M \tilde{x}_C] \neq 0$), $\hat{x}$ is still better than $\hat{x}_M$ if and only if $E[\tilde{x}'_M \tilde{x}_C] < E[\tilde{x}'_M \tilde{x}_M] = \text{mse}(\hat{x}_M)$, where $\tilde{x} = x - \hat{x}$ is the estimation error.

Note that this result, which holds when $M \cup C \subset S$, does not contradict the finding presented in [48] that the optimal use of more models is not necessarily better, because the above result would not necessarily be correct if $C \subset S$ were not true. Since here we focus on problems with a continuous mode space, $M \cup C \subset S$ holds in general and thus $\hat{x}_\alpha = (1-\alpha)\hat{x}_M + \alpha\hat{x}_C$ with some α will be better than $\hat{x}_M$. As a consequence, optimal use of more models for such problems does improve performance because its estimate $\hat{x}$ cannot be worse than $\hat{x}_\alpha$. Of course, this holds true only under the simplifying assumption $s \in (M \cup C)$, which is not necessarily true in general.

3.2.2 Estimated-Mode Augmentation

In principle, the augmented model can be estimated under any optimality criterion, such as

- expected-mode augmentation: the augmented model is the one based on the condition mean

$$\hat{m}_k^{MMSE} = E[s_k | s_k \in M_k, z^k]$$

where M_k and z^k stand for the model set and measurements sequence at time k, respectively.

- maximum-likelihood model augmentation: the augmented model is the one with the largest likelihood

$$\hat{m}_k^{ML} = \arg\max_m f(z^k | s_k = m)$$

- maximum posterior model augmentation: the augmented model is the one with the maximum model probability

$$\hat{m}_k^{MAP} = \arg\max_m P(s_k = m | z^k)$$

A promising alternative is to augment the model set also by the predicted modes, such as $\hat{m}_{k+1|k}^{MMSE} = E[s_{k+1} | s_k \in M, M^{k-1}, z^k] = \sum_{m_j \in M} m_j \mu_{k|k}^{(j)}$, $\hat{m}_{k+1|k}^{ML} = \arg\max_m f(z^k | s_{k+1} = m)$, $\hat{m}_{k+1|k}^{MAP} = \arg\max_m P(s_{k+1} = m | z^k)$, to anticipate the next mode transition, leading to what can be called *predicted-mode augmentation.*

3.2.3 Expected-Mode Augmentation

An expected-mode augmentation (EMA) to the VSMM was originally proposed in [49]. Its general formulation as well as its theoretical analysis and justification have been done in

[53, 54]. The expected mode is the expected value of the true mode. It is a good model candidate since it is statistically closest to the true mode. This expected mode (conditional mean) can be approximated by a sum of modal states weighted by the corresponding model probabilities, readily available from the underlying MM algorithm

$$\hat{m}_k^{MMSE} = E[s_k | s_k \in M_k, z^k] = \sum_{m^j \in M} m^j \mu_{k|k}^j \tag{3.1}$$

where $\mu_{k|k}^j$ denotes the updated probabilities of model j being the correct one, and m^j is the parameter value that characterizes model j.

3.2.4 Practical EMA Algorithms

We now describe practical EMA algorithms. For simplicity of presentation, we assume that the IMM mechanism is used for model-conditioned reinitialization [52].

The proposed EMA algorithms involve the following main functional modules

- EMA $M_k := M^+(M_1, \ldots, M_q)$: *expected mode augmentation* procedure;
- VSIMM$[M_k, M_{k-1}]$: recursion for *variable structure IMM* estimation that uses model sets M_{k-1} and M_k at time $k-1$ and k, respectively;
- EF$[M_k', M_k''; M_{k-1}]$: procedure for *estimation fusion* of two estimates resulting from VSIMM$[M_k', M_{k-1}]$ and VSIMM$[M_k'', M_{k-1}]$ recursions, respectively, where M_k' and M_k'' are discussed later.

The VSIMM and EF functions have been developed, utilized, and documented in several publications on VSMM estimation [43, 59, 57, 55]. For the EMA procedure, a more detailed discussion is given next. We outline three EMA algorithms as follows.

Consider a generic cycle from time $k-1$ to k. Suppose that the model set M_{k-1} used at $k-1$ is given. Three basic EMA algorithms are given in Tables 1, 2, and 3, respectively, using different schemes for determination of the model set M needed to obtain the expected-mode set $E_k = E(M; M_1, \ldots, M_q)$ at time k. Choices of $M_1, \ldots, M_q$ are discussed later. The main difference among the three algorithms lies in how the expected-mode set E_k is determined. Algorithm A (Step 1) uses M_{k-1} (including E_{k-1}) but not the current measurement z_k to determine E_k. On the contrary, Algorithm B (Step 2) uses z_k but not E_{k-1} to determine E_k. Algorithm C (Step 3) uses both z_k and E_{k-1} to determine E_k. In general, Algorithm B should outperform Algorithm A at the time instant of a system mode jump (e.g., with a faster response and hence a smaller peak error) because of the timely information included in z_k, while Algorithm A should have a better steady-state performance due to the more direct utilization of the old expected modes. Algorithm C provides a trade off between steady-state performance and fast response.

Algorithm A is the simplest, while Algorithm C is the most sophisticated. Thanks to the optimal estimation fusion formulas described in [43], the computational complexities of Algorithms B and C increased by the use of the current measurement z_k to determine E_k are quite limited.

The above algorithms can be integrated to yield more sophisticated algorithms with improved performance. For example, we can use $E_k = E_k^A \cup E_k^B$ as the set of expected modes, where E_k^A and E_k^B are the sets of (predicted and updated) expected modes obtained by Algorithms A (Step 1) and B (Step 2), respectively; or more preferably, we may use $E_k = E_k^A \cup E_k^C$ as the set of expected modes, where E_k^A and E_k^C are the sets of (predicted and updated) expected modes obtained by Algorithm A (or C) in Step 1 and Algorithm C in Step 3, respectively, which is equivalent to replacing Step 5 of Algorithm C by running EF$[M_k', E_k; M_{k-1}]$.

In Step 1 of Algorithm A, use of the *predicted* model probabilities at the *current* time step $\{\mu_{k|k-1}^{(i)}\}_{m_i \in M_{k-1}}$ amounts to $\bar{m}_k = \bar{m}_{k|k-1}$ and should be superior to use of the *updated* model probabilities at the *previous* time step $\{\mu_{k-1|k-1}^{(i)}\}_{m_i \in M_{k-1}}$, which amounts to assuming $\bar{m}_k = \bar{m}_{k-1|k-1}$. The same is true for Algorithm C. Both sets of model probabilities are readily available from an MM estimator.

Table 1. One cycle of EMA Algorithm A

S1. Obtain $E_k = E(M_{k-1}; M_1, \ldots, M_q)$ using the *predicted* model probabilities $\{\mu_{k|k-1}^{(i)}\}_{m_i \in M_{k-1}}$

S2. For $M_k = E_k \cup (M_{k-1} - E_{k-1})$, run VSIMM$[M_k, M_{k-1}]$ to obtain the overall estimates, error covariances, and model probabilities $\left\{\hat{x}_{k|k}^{(i)}, P_{k|k}^{(i)}, \mu_{k|k}^{(i)}\right\}_{m_i \in M_k}$

Table 2. One cycle of EMA Algorithm B

S1. For $M_f = M_{k-1} - E_{k-1}$, run VSIMM$[M_f, M_{k-1}]$ to obtain $\left\{\hat{x}_{k|k}^{(i)}, P_{k|k}^{(i)}, \mu_{k|k}^{(i)}\right\}_{m_i \in M_f}$

S2. Obtain $E_k = E(M_f; M_1, \ldots, M_q)$ using the *current* updated model probabilities $\{\mu_{k|k}^{(i)}\}_{m_i \in M_f}$

S3. Run VSIMM$[E_k, M_{k-1}]$ to obtain $\left\{\hat{x}_{k|k}^{(i)}, P_{k|k}^{(i)}, \mu_{k|k}^{(i)}\right\}_{m_i \in E_k}$

S4. Run EF$[M_f, E_k; M_{k-1}]$ to obtain overall estimates, error covariances, and model probabilities $\left\{\hat{x}_{k|k}^{(i)}, P_{k|k}^{(i)}, \mu_{k|k}^{(i)}\right\}_{m_i \in M_k}$ in the set $M_k = M_f \cup E_k$

Table 3. One cycle of EMA Algorithm C

S1. Obtain $E'_k = E(M_{k-1}; M_1, \ldots, M_q)$ using the *predicted* model probabilities $\{\mu_{k|k-1}^{(i)}\}_{m_i \in M_{k-1}}$

S2. For $M'_k = E'_k \cup (M_{k-1} - E_{k-1})$, run VSIMM$[M'_k, M_{k-1}]$

S3. Obtain $E_k = E(M'_k; M_1, \ldots, M_q)$ using the *current* updated model probabilities $\{\mu_{k|k}^{(i)}\}_{m_i \in M'_k}$

S4. Run VSIMM$[E_k, M_{k-1}]$

S5. For $M_f = M_{k-1} - E_{k-1}$, run EF$[M_f, E_k; M_{k-1}]$ to obtain overall estimates, error covariances, and model probabilities $\left\{\hat{x}_{k|k}^{(i)}, P_{k|k}^{(i)}, \mu_{k|k}^{(i)}\right\}_{m_i \in M_k}$ in the set $M_k = M_f \cup E_k$

3.3 EMA-IMM Algorithms for Maneuvering Target Tracking

3.3.1 Maneuvering Target Tracking

Tracking is the estimation of the state of a moving target based on measurements. The target state usually consists of kinematic components (position, velocity, acceleration, etc.) and other parameters [5]. The target motion uncertainty is one of the major challenges of target tracking, i.e., a target may undergo a known or unknown maneuver during an unknown time period. Clearly, this problem involves both the estimation of continuous-valued parameters such as target states and the detection of target motions. Thus, it is natural to pose target tracking as a hybrid estimation problem involving both continuous and discrete uncertainties.

In general, a nonmaneuver motion and different maneuvers can be described only in different motion models. The use of an incorrect model often leads to unacceptable results. When tracking a maneuvering target, it is thus crucial to determine reliably and timely the right model to use [43]. Currently the multiple model method is a major approach to target tracking.

3.3.2 Tracking Problem

The target motion-measurement model is

$$x_{k+1} = Fx_k + G\left[a\left(k\right) + w_k\right]$$

$$z_{k+1} = Hx_{k+1} + v_{k+1}, \quad k = 0, 1, 2, \ldots$$

where $x \triangleq (\mathsf{x}, \dot{\mathsf{x}}, \mathsf{y}, \dot{\mathsf{y}})'$ denotes the target state, $a \triangleq (a_x, a_y)'$ is the acceleration, $w_k \sim \mathcal{N}\left[0, Q\right]$ is the acceleration process noise, $z = (z_x, z_y)'$ is the measurement, $v_k \sim \mathcal{N}\left[0, R\right]$ is the random measurement error, and $F = \text{diag}[F_2, F_2]$ and $G = \text{diag}[G_2, G_2]$ with

$$F_2 = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \quad G_2 = \begin{bmatrix} T^2/2 \\ T \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

The unknown true acceleration is assumed piecewise constant, varying over a given continuous planar region A^c. In the MM framework, we consider a generic finite set (grid) of acceleration values:

$$A_r \triangleq \{a_i \in A^c : i = 1, 2, \ \ldots, r\} \tag{3.2}$$

which defines the total model set. We approximate the evolution of the true acceleration over the quantized set A_r via a Markov chain model, that is, $a(k) \in A_r$ with given $P\left\{a(0) = a_i\right\} = P_i$ and $P\left\{a(k) = a_j | a(k-1) = a_i\right\} = \pi_{ij}$ for $i, j = 1, 2, \ \ldots, \ r$.

3.3.3 Designs of EMA Algorithms

Consider the following well-known example of [1], [46], [60], [73], [41], [57], [55]. The mode space is defined as

$$A^c \triangleq \{(a_x, a_y) : |a_x| + |a_y| \leq 40\}$$

i.e., the maximum acceleration in any coordinate direction is about $4g(g = 9.8m^2/s)$. It is, however, more appropriate that the target acceleration model be independent of the orientation of the observer's coordinate system. That is why we consider

$$A_0^c \triangleq \left\{(a_x, a_y) : \sqrt{a_x^2 + a_y^2} \leq 40\right\}$$

IMM13

The basic 13-model set design A_{13}, obtained after quantization of A^c, is

$$\left\{\begin{array}{lll} a_1 = \rho[0,0]' & a_2 = \rho[1,0]' & a_3 = \rho[0,1]' \\ a_4 = \rho[-1,0]' & a_5 = \rho[0,-1]' & a_6 = \rho[1,1]' \\ a_7 = \rho[-1,1]' & a_8 = \rho[-1,-1]' & a_9 = \rho[1,-1]' \\ a_{10} = \rho[2,0]' & a_{11} = \rho[0,2]' & a_{12} = \rho[-2,0]' \\ a_{13} = \rho[0,-2]' & & \end{array}\right\}$$

with $\rho = 20 \approx 2g$.

The transition relations among models are easily understood in terms of the directed graph (i.e., digraph) representation of an MM, introduced in [41]. The topology of model set A_{13} is depicted in Figure 3.1. Each model is viewed as a point in the mode (acceleration) space. An arrow from one model to another indicates a legitimate model switch (self-loops are omitted) with nonzero probability. All details can be found in [41], [57]. Note that for simplicity in A_{13} a model is allowed to switch to its nearest neighbor(s) only. Better results could be obtained if other types of model switching are allowed, such as those between second nearest neighbors (e.g., a_2 and a_3, and a_6 and a_{10}) (see [55]). The values of the transition probability matrix used in our implementation are the same as given by (7) of [57].

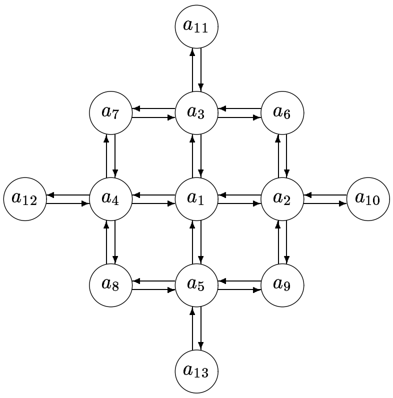


Figure 3.1: Diagraph representation for 13-model set

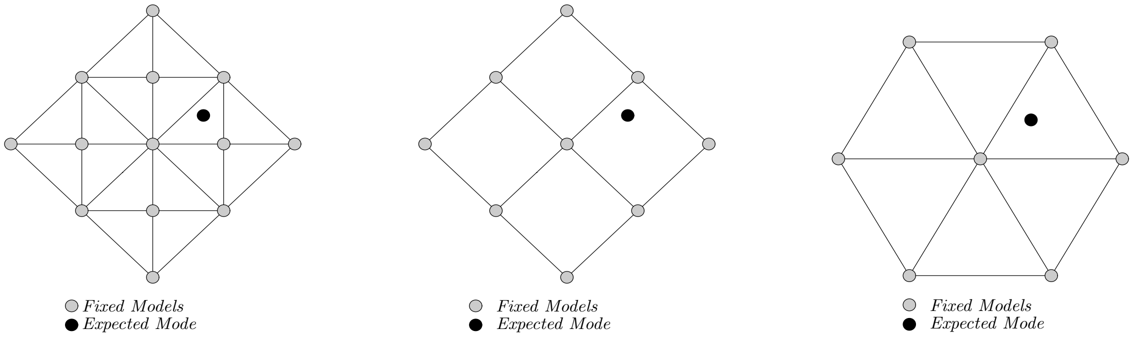


Figure 3.2: EMA for 13-, 9- and 7-Model Set Designs

EMA{13+1}

As illustrated in Figure 3.2 (a), the expected-mode augmented (EMA) set of the fixed-grid model A_{13} is $A_{13+1}(k) \triangleq \left\{A_{13}, \hat{a}_{k|k-1}\right\}$, where $\hat{a}_{k|k-1} = \sum_{a_j \in A_{13+1}(k-1)} \mu_{k|k-1}^{(j)} a_j$ and $\mu_{k|k-1}^{(j)}$ are the predicted model probabilities available from the IMM estimator. Note that $\hat{a}_{k|k-1}$, as a convex combination of the points of A_{13}, covers the entire continuous acceleration region A^c, i.e. $\hat{a}_{k|k-1}$ can be any point in A^c, depending on $\mu_{k|k-1}^{(j)}$. The values of the transition probability matrix (TPM) $\Pi = [\pi_{ij}]$ in the simulation were chosen based on the TPM

$P = [p_{ij}]$ of A_{13} as follows:

$$\pi_{1,14} = 0.01, \quad \pi_{i,14} = 0.05, \quad i = 2,\, 3,\, \ldots,\ 13$$

$$\pi_{jj} = p_{jj} - \pi_{j,14}, \quad \pi_{14,j} = 0.01, \quad j = 1, 2, \ldots,\ 13$$

and all other elements remain unchanged (i.e., $\pi_{ij} = p_{ij}$).

The implementation of the corresponding IMM estimator with EMA-Algorithm A as given in Table 1 is straightforward with $A_{13+1}(k)$. This implementation is referred to as the EMA-A{13+1}.

EMA{9+1}

This model, illustrated in Figure 3.2 (b), is obtained from EMA{13+1} by deleting its internal nonzero models (vertices) a_2, a_3, a_4, a_5.[1] Its TPM used in the simulation was

$$\begin{bmatrix}
.96 & .001 & .001 & .001 & .001 & .001 & .001 & .001 & .001 & .032 \\
.01 & .75 & .0 & .0 & .0 & .01 & .0 & .0 & .01 & .22 \\
.01 & .0 & .75 & .0 & .0 & .01 & .01 & .0 & .0 & .22 \\
.01 & .0 & .0 & .75 & .0 & .0 & .01 & .01 & .0 & .22 \\
.01 & .0 & .0 & .0 & .75 & .0 & .0 & .01 & .01 & .22 \\
.01 & .01 & .01 & .0 & .0 & .75 & .0 & .0 & .0 & .22 \\
.01 & .0 & .01 & .01 & .0 & .0 & .75 & .0 & .0 & .22 \\
.01 & .0 & .0 & .01 & .01 & .0 & .0 & .75 & .0 & .22 \\
.01 & .01 & .0 & .0 & .01 & .0 & .0 & .0 & .75 & .22 \\
.002 & .002 & .002 & .002 & .002 & .002 & .002 & .002 & .002 & .982
\end{bmatrix}$$

[1] The variant with deleting a_6, a_7, a_8, a_9 was also examined, but the one presented here showed better performance.

The implementation of the corresponding IMM estimator with EMA-Algorithm A for this design is referred to as the EMA-A{9+1}.

EMA{7+1} & EMA{7+2}

In order to cover more efficiently the true acceleration set A_0^c defined above we also considered the diamond fixed model set design A_7 [45], illustrated in Figure 3.2 (c):

$$\left\{\begin{matrix} a_1 = \rho\,[0,0]' & a_2 = \rho\,[2,0]' & a_3 = \rho\left[1,\sqrt{3}\right]' \\ a_4 = \rho\left[-1,\sqrt{3}\right]' & a_5 = \rho\,[-2,0]' & a_6 = \rho\left[-1,-\sqrt{3}\right]' \\ a_7 = \rho\left[1,-\sqrt{3}\right]' & & \end{matrix}\right\}$$

with $\rho = 20$. Two types of EMA designs based on A_7 were considered—single-model augmentation, denoted by $A_{7+1}(k) \triangleq \{A_7,\, \hat{a}_8(k)\}$ (Fig. 3.2) and two-model augmentation, denoted by $A_{7+2}(k) \triangleq \{A_7,\, \hat{a}_8(k),\, \hat{a}_9(k)\}$. All Algorithms A, B, and C presented above were implemented for both A_{7+1} and A_{7+2}. The corresponding algorithms are denoted as EMA-A{7+i}, EMA-B{7+i}, EMA-C{7+i}, $i = 1, 2$, respectively. The EMA{7+1} and EMA{7+2} algorithms always use $\hat{a}_8(k) = \bar{m}_1$ as computed by (3.1). The EMA{7+2} algorithms also compute $\hat{a}_9(k)$ as the probabilistically weighted sum of the accelerations of the three most probable models in the respective model set.

The following transition probability matrices for the Markov chains over A_{7+1} and A_{7+2}, respectively were used in the simulation

$$\begin{bmatrix} .894 & .001 & .001 & .001 & .001 & .001 & .001 & 0.1 \\ .05 & .65 & .05 & .0 & .0 & .0 & .05 & 0.2 \\ .05 & .05 & .65 & .05 & .0 & .0 & .0 & 0.2 \\ .05 & .0 & .05 & .65 & .05 & .0 & .0 & 0.2 \\ .05 & .0 & .0 & .05 & .65 & .05 & .0 & 0.2 \\ .05 & .0 & .0 & .0 & .05 & .65 & .05 & 0.2 \\ .05 & .05 & .0 & .0 & .0 & .05 & .65 & 0.2 \\ .001 & .001 & .001 & .001 & .001 & .001 & .001 & 0.993 \end{bmatrix}$$

and

$$\begin{bmatrix} .964 & .001 & .001 & .001 & .001 & .001 & .001 & .015 & .015 \\ .05 & .65 & .05 & .0 & .0 & .0 & .05 & .1 & .1 \\ .05 & .05 & .65 & .05 & .0 & .0 & .0 & .1 & .1 \\ .05 & .0 & .05 & .65 & .05 & .0 & .0 & .1 & .1 \\ .05 & .0 & .0 & .05 & .65 & .05 & .0 & .1 & .1 \\ .05 & .0 & .0 & .0 & .05 & .65 & .05 & .1 & .1 \\ .05 & .05 & .0 & .0 & .0 & .05 & .65 & .1 & .1 \\ .01 & .01 & .01 & .01 & .01 & .01 & .01 & .9 & .03 \\ .01 & .01 & .01 & .01 & .01 & .01 & .01 & .03 & .9 \end{bmatrix}$$

All other parameters of the IMM algorithms implemented in the simulation were the same as given in [57], e.g., $T = 1s$, $Q^1 = (0.003)^2 I$, $Q^j = (0.008)^2 I$, $j \neq 1$, $R = 1250I$.

3.4 Performance Evaluation

3.4.1 Test Scenarios

The performances of all nine MM tracking algorithms (viz., IMM13, EMA-A{13+1}, EMA-A{9+1} and EMA-A{7+i}, EMA-B{7+i}, EMA-C{7+i}, $i = 1, 2$) were investigated first over a large number of deterministic maneuver scenarios with fixed acceleration sequences. Deterministic scenarios serve to evaluate algorithms' peak errors, steady-state errors and response times. We present two of them, referred to as DS1 and DS2, in the sequel. Their acceleration values are given in Table 3.1. The other parameters for both scenarios are $T = 1s,\ Q = 0,\ R = 1250I, x_0 = [8000,\ 25,\ 8000,\ 200]'$.

Note that while the acceleration values in DS1 are relatively close to the fixed grid points of IMM13, in DS2 they are deliberately chosen far apart from the grid points. As such, for the fixed structure estimator IMM13 the scenario DS2 is more difficult than DS1.

To provide performance comparison as fair as possible performance comparison over an ensemble of maneuver trajectories, the algorithms were tested on a random scenario, developed in [57], [55]. With such a scenario, it is difficult, if not virtually impossible, to design an MM estimator with subtle tricks that are effective only for certain scenarios. In the random scenario the acceleration vector $\mathbf{a}(t) = a(t)\angle\theta(t)$ is a 2-dimensional semi-Markov process which undergoes sudden jumps from a state with a magnitude a and phase θ to another one after staying in it for a random period of time. Briefly, the model assumes: the sojourn time τ_k of the state and variance σ^2; the acceleration magnitude a_{k+1} has probability masses P_0 and P_M to be zero and maximum, respectively, and uniform in between; the angle θ_{k+1} of acceleration is uniform over 2π if $a_k = 0$ and Gaussian with mean θ_k and variance

Table 3.1: Deterministic Scenarios' Parameters

Scenario	DS1		DS2	
k	$a_x(k)$	$a_y(k)$	$a_x(k)$	$a_y(k)$
$1-30$	0	0	0	0
$31-45$	18	22	8	22
$46-55$	2	37	12	27
$56-80$	0	0	0	0
$81-98$	25	2	15	2
$99-119$	-2	19	-2	9
$120-139$	0	-1	0	-1
$140-150$	38	-1	28	-1
$151-160$	0	0	0	0

σ_θ^2 otherwise. All details and discussions were given in [57]. In the simulation we used the following parameters:

$$\bar{\tau} = \bar{\tau}_M + \tfrac{a_{\max}-a}{a_{\max}}(\bar{\tau}_0 - \bar{\tau}_M),\ \sigma_\tau = \tfrac{1}{12}\bar{\tau}_a$$

$$\bar{\tau}_M = 10,\ \bar{\tau}_0 = 30,\ P_M = 0.1, a_{\max} = 37$$

$$\sigma_\theta = \pi/12,\ P_0 = 0.8,\ a_k = a_{\max}$$

3.4.2 Simulation Results

Two main tasks were of interest in the simulation:

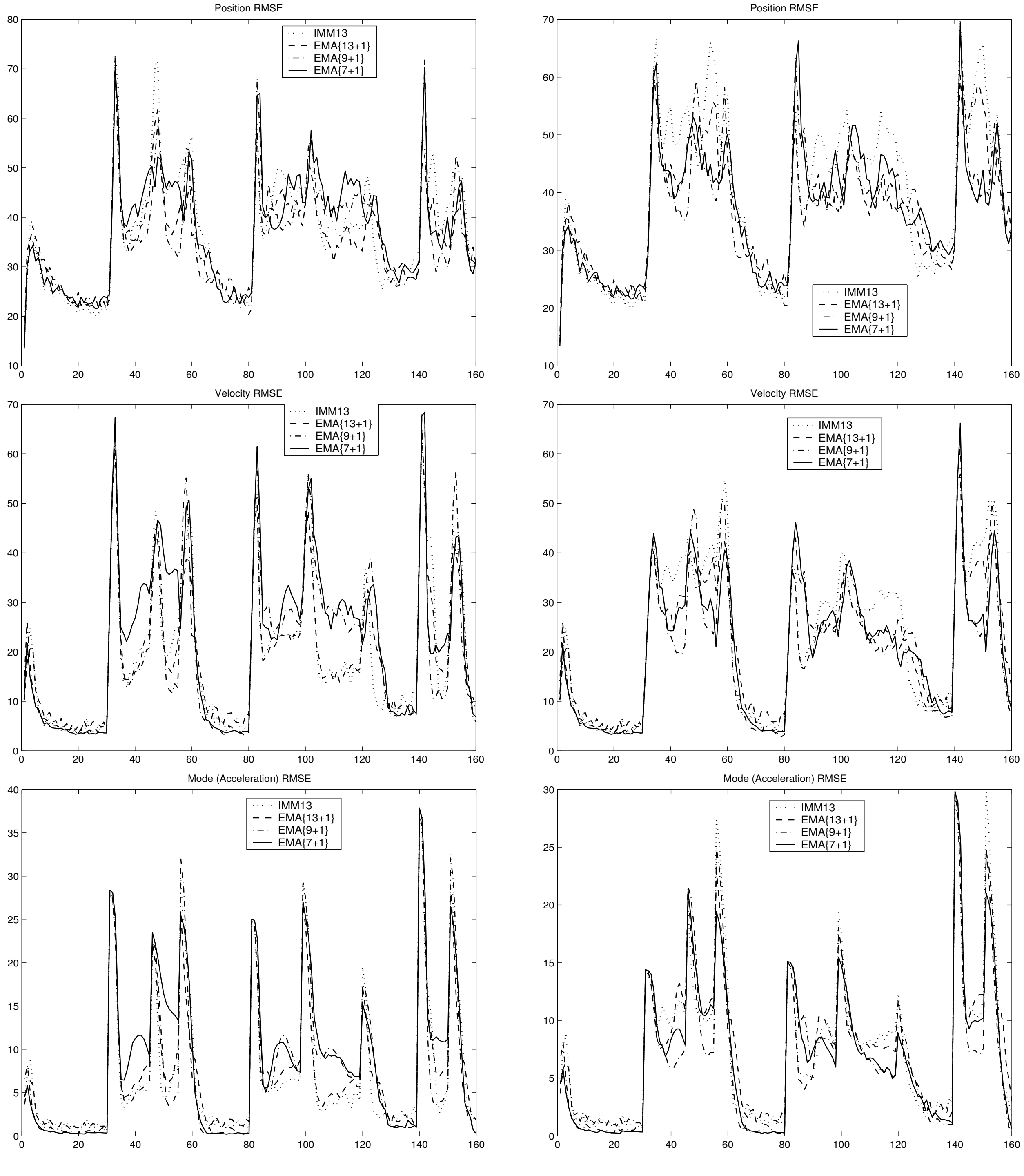
a) Evaluate the performance of a particular EMA algorithm (viz., EMA-A) with different designs (viz., EMA-A{13+1}, EMA-A{9+1} and EMA-A{7+1}) and compare with the fixed structure algorithm (IMM3).

b) Evaluate and compare the performances of the different EMA algorithms (EMA-A, EMA-B, EMA-C) with different number of models within a particular design (viz., A{7+i}).

EMA-A{13+1}, EMA-A{9+1} and EMA-A{7+1} vs. IMM

Results over 100 Monte Carlo runs of DS1 and DS2 are plotted in Figure 3.3. It is seen first that EMA-A{13+1} in both scenarios (and in all other simulated scenarios as well, not shown) outperforms the remaining algorithms, in particular the fixed-grid IMM13. The comparative results between IMM13 on the one hand and EMA-A{9+1} (EMA-A{7+1}) on the other are scenario dependent. While IMM13 provides less biased steady-state errors in DS1, EMA-A{9+1} and EMA-A{7+1} give better accuracy for almost all jumps in DS2. This is due to the fact (mentioned before) that for IMM13, DS1 and DS2 represent respectively easy and difficult scenarios, regarding the closeness of the true acceleration to the fixed-grid values. The EMA algorithms are less scenario-dependent.

Results over 500 runs of the random scenario are given in Figure 3.4. Clearly all EMA designs provide better "overall accuracy". What is somewhat surprising is the negligible difference between EMA-A{13+1} and the two other EMA designs, which have fewer models. A possible explanation is that the modal estimate provides a good "coverage" of the whole continuous region of possible (simulated) accelerations, even when the number of the fixed models is small (7 and 9 respectively).



(a) DS1

(b) DS2

Figure 3.3: Estimation Errors (DS1 & DS2)

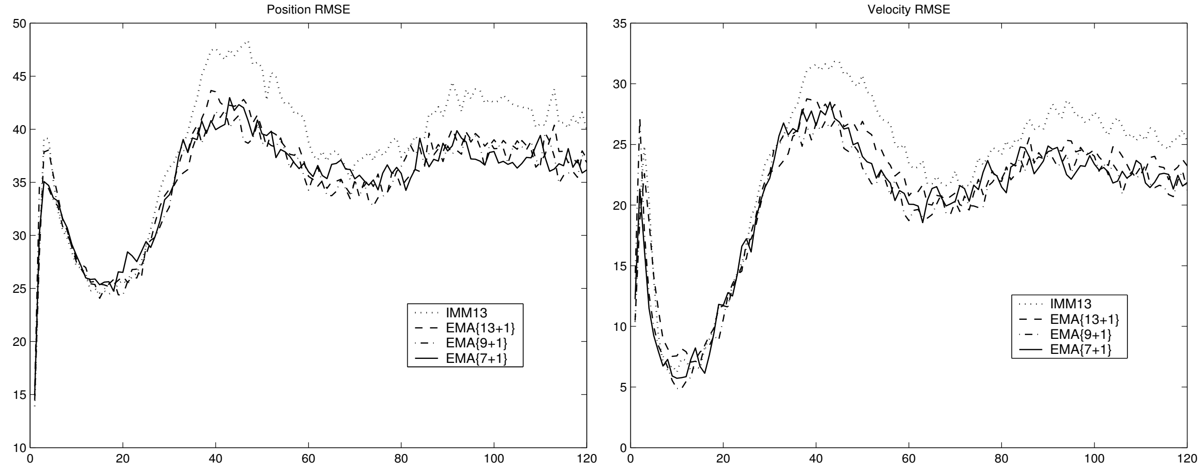


Figure 3.4: Estimation Errors (Random Scenario)

EMA-A{7+i}, EMA-B{7+i} and EMA-C{7+i}, $i = 1, 2$

Accuracy comparison results over 100 Monte Carlo runs for DS1 are plotted in Fig. 3.5 for the EMA{7+1} algorithms versus the EMA{7+2} algorithms. It is seen that for all three algorithms A, B, and C, two-expected-mode augmentation in the EMA{7+2} algorithms has substantially reduced peak errors compared with the EMA{7+1} counterpart. During the steady-state regimes (no-jumps present) the errors of the respective algorithms A, B, and C are virtually indistinguishable. The results for the other deterministic scenario DS2 (not presented here) are very similar. As shown in Fig. 3.6 over 500 runs of the random scenario, the EMA{7+2} algorithms have a smaller overall error reduction relative to the EMA{7+1} algorithms. Also plotted in this figure are the results of the standard fixed-structure IMM algorithm with 13 models (denoted as IMM13) [57]. It is seen that the EMA algorithms, which are of a variable structure, have a substantially better accuracy than the fixed-structure IMM algorithm. As evidenced by Table 3.2, this performance superiority is achieved by the EMA algorithms at a computational complexity only about half of that of the IMM13. The reason for this performance improvement is clear from Fig. 3.7—the augmenting model (i.e., the expected mode $\hat{a}_8$) is on average much better than every other model in the set.

Fig. 3.8 depicts comparative results between the three algorithms A, B, and C over DS2. Although the accuracy differences are not significant, a visible tendency is that Algorithm B appears to provide a faster response (and hence smaller peak errors) than the other two algorithms. As explained before, this is due to the fact that its set of the expected modes relies more on the current measurement information than Algorithms A and C. However, this results in larger steady-state errors.

Computational load

The computational loads of the algorithms evaluated in terms of *relative floating point operations* (FLOP) ratios with respect to the standard IMM13 are summarized in Table 3.2.

Table 3.2: Computational Load

IMM13	EMA$\{13+1\}$	EMA$\{9+1\}$	EMA$\{7+2\}$			EMA$\{7+1\}$		
	A	A	A	B	C	A	B	C
1	1.138	0.677	0.598	0.592	0.669	0.507	0.500	0.543

3.5 Summary

A new approach — Expected-Mode Augmentation — for MM estimation has been proposed and examined. Practical algorithms of expected-mode augmentation, which have a variable structure, for MM state estimation over a *continuous* mode space have been developed and investigated. Extensive simulations of maneuvering target tracking have been conducted that have demonstrated the capabilities of the proposed algorithms for performance enhancement and computation reduction.

The EMA approach is generally applicable, wherever the mode space is continuous. It can be applied to fixed- and variable-structure MM algorithms and supplements the existing methods of variable-structure MM estimation and facilitates the design of more efficient practical MM estimators.

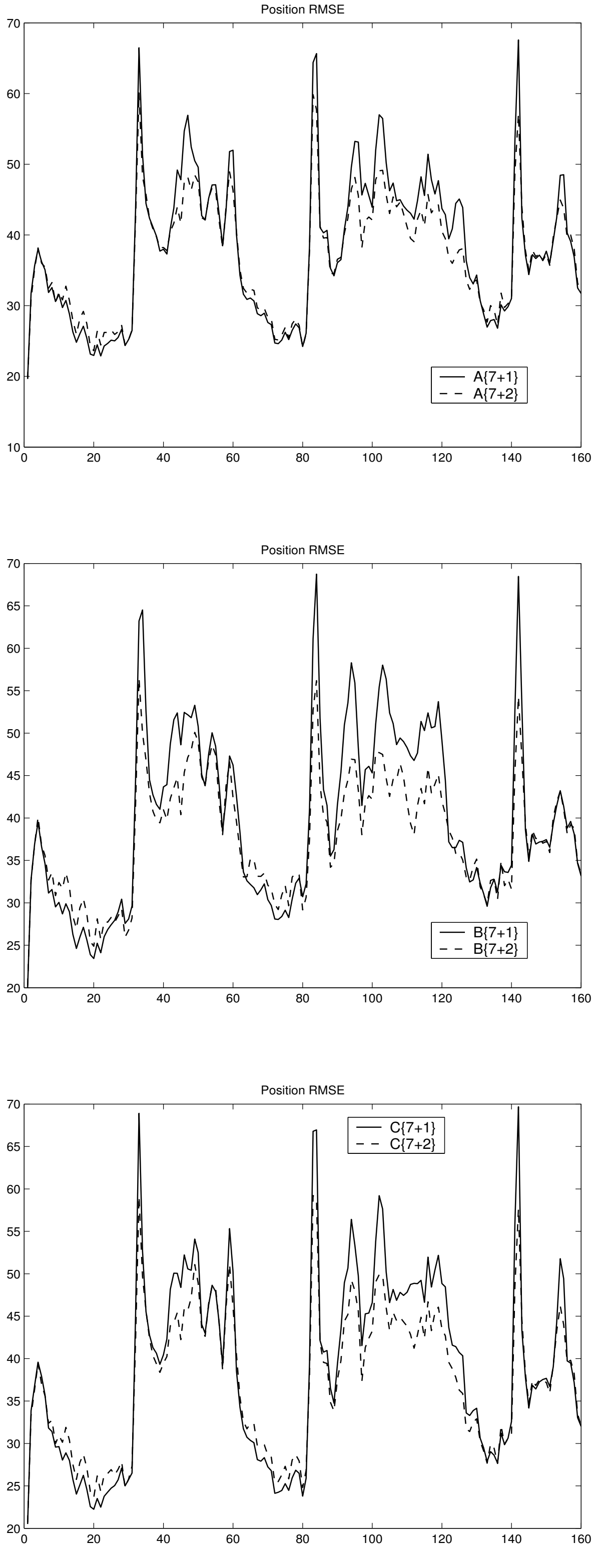


Figure 3.5: RMS Position Errors (DS1): $7 + 1$ vs. $7 + 2$

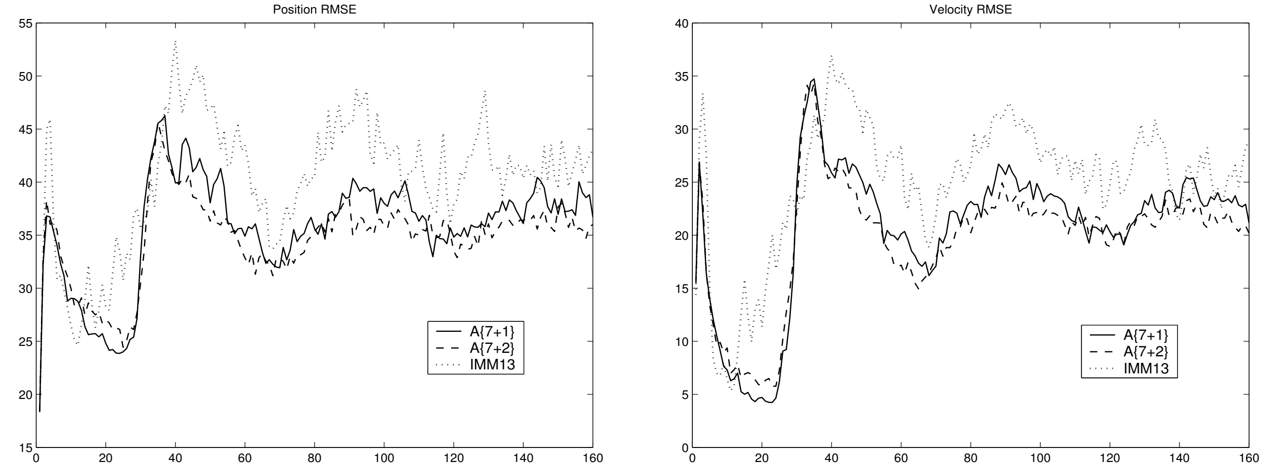


Figure 3.6: RMS Errors (RS): $7 + 1$, $7 + 2$, IMM

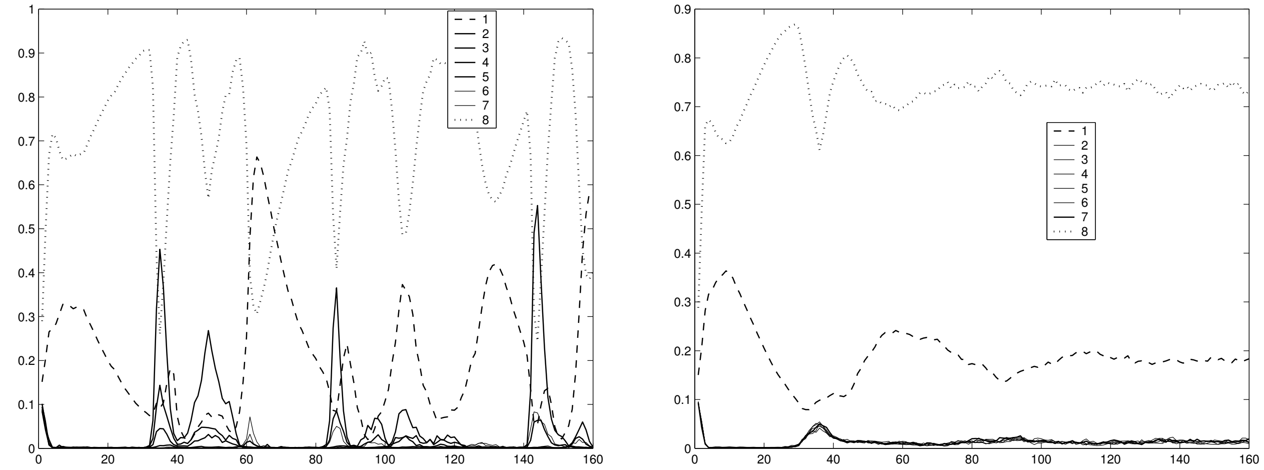


Figure 3.7: Average Model Probabilities

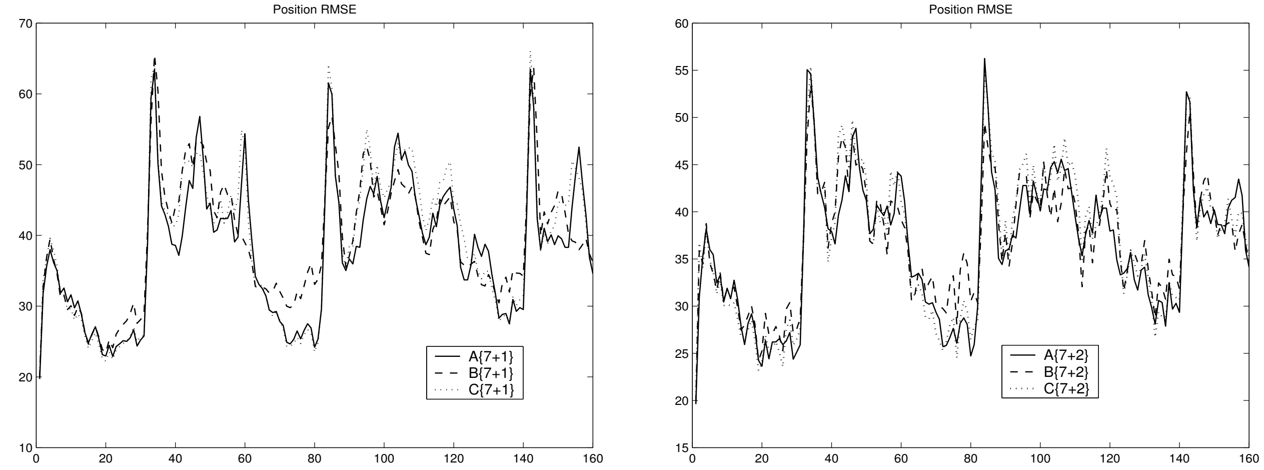


Figure 3.8: RMS Position Errors (DS2): Algs. A, B, and C

Chapter 4

Adaptive MM approach to Fault Detection, Identification, and Estimation

4.1 Introduction and Related Research

With increasing complexity of modern engineering systems, the requirement for reliability, availability and security is growing significantly. Fault detection and diagnosis (FDD) is becoming a major issue in safety critical systems such as modern flight control systems. In the last two decades many techniques have been developed for FDD. Different methods are surveyed in [9, 23, 26, 29, 99, 78].

In the FDD literature, two types of failures are usually studied:

- **Sensor failures** are discrepancies between the measured system variable and its actual value

- **Actuator failures** are discrepancies between the input of an actuator and its actual output

and the *goal* is to accomplish the following tasks:

- **Fault detection**: Determine the presence of a fault and detection time
- **Fault identification**: Determine the type of a fault. Follows fault detection
- **Fault estimation**: Determine the extent of a fault. Follows fault identification
- **Fault accommodation**: Reconfigurate the system using healthy components

The following performance measures are often used

- **False alarm**: A fault is declared when the system is normal
- **Miss detection**: No fault is indicated when a fault is occurred in the system
- **Detection delay**: The time delay when a fault is correctly detected under the fixed false alarm rate

4.1.1 Conventional Approaches to FDI

Conventional approaches to fault detection and identification (FDI) in dynamic systems can be classified into two categories: *hardware (physical) redundancy* and *analytical redundancy.* An approach relying on hardware redundancy uses multiple sensors and actuators. It has been applied in the control of safety-critical systems such as aircraft space vehicles and nuclear power plants. However, this approach is of limited effectiveness due to its extra cost and space. In the analytical redundancy approach, most methods are model-based and detect faults based on a residual signal, which is the difference generated by observations and the system mathematical model. In normal operation, the residuals should be small. Otherwise they are indicative of failures. There are various approaches to residual generation, such as diagnostic observers, parity relation, detection filters and parameter estimation, which are briefly introduced as follows. More details and discussions can be found in [9, 23, 26, 29, 99, 78, 19, 79].

Voting schemes Voting techniques are often used in systems that possess high degrees of parallel hardware redundancy [99]. The basic idea is to compare measurements of the same variable from different sensors. Any serious discrepancy is an indication of a fault. For example, if one of the three signals differs markedly from the other two, the differing signal is identified as faulty. Voting schemes are in general relatively easy to implement and are quick to identify mechanical faults in instruments. However, based on the assumption of independent faults, voting procedure requires the use of dissimilar redundancy, which is obtained by another system with the same function as the first but built according to different principles and technologies. This presents a difficulty for voting schemes, not to mention the extra cost. In addition, voting techniques can have difficulties in detecting "soft failures" such as a small bias shift.

Diagnostic observers The observer is to reconstruct the system outputs from the measurements using state estimation errors as a residual for the detection and identification of the faults. The main concern of observer-based FDI is the generation of a set of residuals which detect and uniquely identify different faults. To make fault identification possible, the method develops a set of observers, each of which is sensitive by design to an individual fault but insensitive to the remaining faults and the unmodelled disturbances. Once there is a fault, observers which are sensitive to the fault will deviate from the process significantly and result in large residuals. [23] provided a detailed discussion on the design of general diagnosis observers. Generation of diagnostic observers for nonlinear systems has also been investigated in the literature.

Parity relations Parity equations are rearranged input-output or state-space system models, subjected to a linear dynamic transformation. Parity equations only contain the transformed residuals due to faults, which serve for detection and identification. The basic idea of this approach is to check the parity (consistency) of the mathematical equations of the system (analytical relations) by using measurements. Under normal operation, the value of the parity equations is zero. When the value is larger than a preset threshold, a fault is declared. [23, 26, 79] showed that once the desired residual properties have been selected, parity equation and observer based design lead to equivalent residual generators.

The above techniques are well known in the automatic control community. However, they are basically for deterministic systems.

Fault detection filter This approach is to monitor the innovation process in a stochastic dynamic system. It usually involves the use of a Kalman filter, which is designed according to the system model under normal operation. Under the linear Gaussian assumption, the innovation sequence of the Kalman filter is a zero-mean white noise process with known covariance. Its mean becomes nonzero in the presence of faults. However, using a single filter is hard for fault identification. In general a bank of Kalman filters is designed for fault identification, each filter of which is designed for a possible failure. The innovations from different filters are used to compute the conditional probability of each model being true. In this manner, one can perform simultaneous fault identification and state estimation.

Parameter estimation Parameter estimation provides a natural solution when parametric faults are not measurable directly. It makes use of the fact that system faults are reflected in physical parameters such as mass, resistance, friction, inductance, etc. This approach usually needs accurate parametric models of the process as a reference model, which can be first identified in a fault-free situation. Then the parameters are repeatedly re-identified online. Differences from the reference model serve as a basis for detection and identification. The most important issue in the use of the parameter estimation approach for FDI is one of complexity arising from the process model. For example, if the system model is a complicated nonlinear model, the parameter estimation problem turns out to be a nonlinear optimization problem, which is usually hard and could be a bottleneck in the application of this approach.

The FDI procedure based on analytical redundancy consists of two steps: residual generation and decision making. In the literature, there are several statistical tests used to evaluate the residuals, which are briefly presented here and will be discussed with more details in Chapter 6. Simply put, decisions are made on error signals, or innovations or likelihood functions generated from the detection filters. They are designed to maximize the detection probability with a given false alarm rate.

Threshold decision The magnitude of the error signal is simply compared with a threshold according to the desired performance. This approach could give a fast detection of hard faults. However, for soft faults the threshold has to be designed carefully.

Innovation testing There are a number of possible statistical tests that can be performed on the innovation. One of these is a chi-square test. The constructed test statistic by innovations is a chi-square random variable and is used to evaluate residuals by the chi-square test. This method is simple but cannot provide fault identification.

Multiple hypothesis test This detection scheme involves the use of a bank of filters based on different hypotheses. Then innovations from different filters are used to compute the conditional model probabilities. The posterior model probabilities, as an indicator of faults, are tested by Bayesian decision theory.

Generalized likelihood ratio test The generalized likelihood ratio (GLR) is used to test different hypotheses depending on unknown parameters. It involves a double maximization with high computational cost.

Other approaches have also been explored for FDI including those combining hardware and analytical redundancy and knowledge-based approaches (expert systems). Readers are referred to reviews [9, 23, 26, 29, 99, 78, 19, 79] for details.

4.1.2 Multiple Model Approaches for FDI

Recently multiple model techniques have been applied to fault detection and identification. A bank of elemental filters runs in parallel, each based on a model matching a particular mode of the system. In theory, there are advantages in using a multiple model approach. As for fault diagnosis problems, multiple model approaches make it more effective to provide fault detection, identification and even estimation by using all available information, which is desirable for the reconfigurable control to keep systems running safely.

MM algorithms for FDI have been successfully applied to a number of particular problems under other names, such as multiple hypothesis testing [99], multiple adaptive estimate algorithm (MMAE) [65, 69, 72] and multiple dedicated observers [23]. All these algorithms are based on *autonomous* (or "noninteracting") multiple filters, which do not fit well into the framework of FDI in a dynamic system because the structure or parameter of such a system does change as its components or subsystems change. To make MM algorithms more suitable for the current problem, new IMM-FDI algorithms have been proposed [21, 71, 80, 81, 86, 103, 104]. The difference between IMM and the autonomous MM (AMM) algorithms lies in that elemental filters in the IMM interact with each other which leads to improved performance. [104] provides meaningful discussions among MM-based approaches, conventional single-model-based approaches such as generalized likelihood ratio (GLR), and observer-based approaches. Theoretical analysis and research results demonstrate that the IMM-FDI approach significantly improves the FDD performance in terms of fast detection and proper identification.

4.1.3 Motivation of VSMM for FDI

Prior studies of the IMM-FDI show good performance when the designed failure model matches the truth. However, in practice faults often occur with different magnitudes, such as total failures and partial failures, or occur concurrently, known as multiple failures. To have reliable identification and good state estimation under the failure condition, it is very important to estimate the severity of a failure. Most of these existing approaches do not deal with state estimation and fault estimation in dynamic systems. The focus of this chapter is on the development of an adaptive approach to fast detection and accurate estimation of partial failures as well as multiple failures.

In the case of partial or multiple failures, one disadvantage of the previous MM approach based on a fixed model set is that the number of models needed to cover all possible failures can be quite large, making implementation of a fixed-structure MM estimator infeasible. Further, use of more models and filters does not necessarily guarantee performance improvement [43]. However, nothing really prevents us from using a variable set of models.

In this chapter we propose two FDI schemes based on variable-structure MM estimation, which leads to better solution for the problems under investigation. The first approach, naturally based on a hierarchical structure, develops an efficient algorithm with models in two levels. An IMM estimator at the higher level is used to detect and identify failures while another IMM estimator at the lower level is used to estimate failures. Although [22] has a similar structure, it is not actually implemented in a hierarchical way and did not consider the possible jumps between models in the model set. The second proposed approach, called IM^{3}L, has the same mechanism as the first one for detection and identification, but it uses a maximum likelihood estimator to estimate the extent of failures and improve state estimation under failure conditions. This new approach is demonstrated for superior performance through simulations. Moreover, this scheme can handle more difficult cases such as multiple failures.

The proposed approaches belong to two different families of the VSMM algorithms, introduced in Sec. 2.4.2. The hierarchical IMM-FDI scheme belongs to the former family since its total model set is determined in advance and it opens an active subset only at a given time. The IM^{3}L scheme belongs to the second family, which generates a new model based on the failure estimate when a failure is declared, and so it is impossible to specify the total model set as a finite set beforehand. As shown later, the VSMM saves computation as well as improves FDI performance. In addition, the IMM-FDI approach can be naturally extended to fault-tolerant control.

The chapter is organized as follows. An FDD approach based on MM estimation is formulated in a general setting in section 2. An efficient hierarchical structure based on the IMM-FDI algorithm is proposed in section 3. The new approach IM^{3}L for FDI is developed in section 4, specially extended for multiple failures. In section 5, the proposed approaches are tested on two types of aircraft (vertical takeoff and landing aircraft and Boeing 747), and compared with a widely used residual-based GLRT technique over various sensor and actuator failures. Discussions are then presented. Conclusions are given in section 6.

4.2 Fault Detection Using MM

4.2.1 The Dynamic Model for Systems Subject to Failures

In flight control systems, there are several types of possible failures, such as sensor failures, actuator failures and component failures. Therefore, in the MM-FDI, a set of models (hypotheses) are used to represent the possible system structures due to different failures. Each failure hypothesis corresponds to a specific model with a corresponding set of F, G, H and covariance matrices Q and R. Let M denote the set of all designed system models and j a generic model in it. Then a linear dynamic system can be represented by

$$x_{k+1} = F_k^j x_k + G_k^j u_k + T_k^j w_k^j \tag{4.1}$$

$$z_k = H_k^j x_k + v_k^j \tag{4.2}$$

where x is the state vector; z is the measurement vector; u is the control input vector; $w_k \sim \mathcal{N}(\overline{w}_k, Q_k)$ and $v_k \sim \mathcal{N}(\overline{v}_k, R_k)$ are independent process and measurement noises, respectively. It is assumed that the initial state $x_0 \sim \mathcal{N}(\overline{x}_0, P_0)$ is independent of w_k and v_k. It is also assumed that the system mode sequence is a first order Markov chain with transition probabilities pi^{ij} defined in (2.5).

4.2.2 Failure Modeling

Here we consider complex failure situations, including total (hard) failures and partial (soft) failures of sensors and actuators. Partial sensor failures can be caused by reduced sensor power. Partial actuator failures can arise from damage to a control surface resulting in only a portion of control effectiveness delivered [65]. Most of the analytical redundancy based approaches formulate failures as additive or multiplicative system changes [99, 100]. Here we use multiplicative change to model possible failures. However, our proposed approaches are also applicable to additive faults in a straightforward way.

An effectiveness factor, denoted as α $(0 \leq \alpha \leq 1)$, is introduced to represent the extent of a failure. An effectiveness of 0% $(\alpha = 0)$ indicates a “total failure” while an effectiveness of 100% $(\alpha = 1)$ indicates “no failure”; $0 < \alpha < 1$ indicates “partial failures”. That is, $1 - \alpha$ presents the severeness of a failure. Clearly α is an unknown parameter in the model of a linear dynamic system. In model-set design, the value of α is quantized to represent different magnitudes of failure. As such, sensor failures can be modeled by multiplying the respective row of the H matrix by α. Soft sensor failures can also be modeled by changing the measurement noise mean or elements of the R matrix. This type of failure means the output of the sensor is corrupted by the noise. Similarly actuator failures can be modeled by multiplying the respective column of the G matrix by α.

4.2.3 IMM Estimator for FDI

As mentioned above, an IMM-FDI runs a bank of Kalman filters, each based on a different model of the system, where α represents the failure extent.

In the IMM-FDI, model probabilities are used as an indication of a failure because they provide a meaningful measure of how likely each fault mode is at a given time. The fault detection decision can be made by:

$$\begin{aligned} \mu_k^j = \quad & \max_i \mu_k^i > \mu_T \;\Rightarrow\; H_j : \text{fault } j \text{ occurred} \\ & \max_i \mu_k^i < \mu_T \;\Rightarrow : \text{no decision} \end{aligned} \tag{4.3}$$

where μ_T is a preset detection threshold, and $\mu_k^i \triangleq P\{m_k^i | z^k\}$ is the probability that the ith model is true based on the measurement sequence $z^k \triangleq \langle z_i \rangle_{i \leq k}$.

Fig. 4.1 shows the block diagram of the IMM algorithm for FDI. A complete cycle of the IMM-FDD scheme with Kalman filter as its mode-matched filter is summarized in Table 1 of [104].

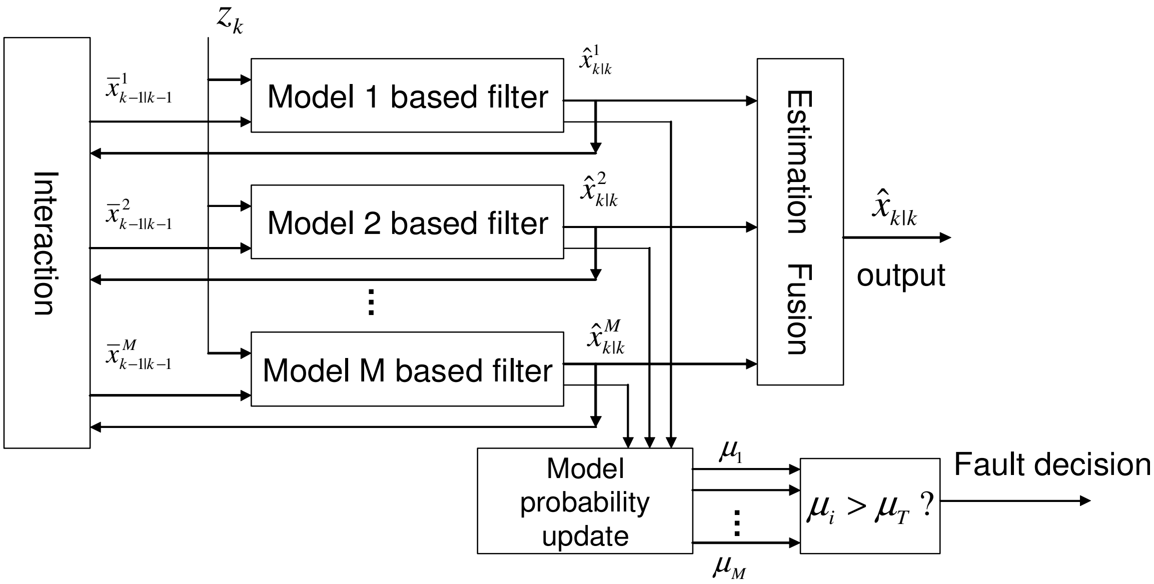


Figure 4.1: The block diagram of the IMM algorithm for FDI

4.3 Hierarchical IMM-FDI

For many applications of flight control systems, it is highly desirable to have reconfigurable control ability. Reliable fault detection and identification with robust state estimation provide very useful information for fault-tolerant control subsequent to the occurrence of the failure. In reality, failures quite often occur with different magnitudes (sizes). In order to get good state estimation, detection and estimation of partial failures are of major importance. The following sections will focus on this issue.

In most prior research, the designed model set only considers normal and total failure models without explicit inclusion of the partial failure models. Although some partial failures can be covered in principle by combinations of the normal and total failure models, the quantization level of the failure degree is too crude to produce an accurate estimate of the extent of failure. A natural idea is to have a finer parameter quantization. This means that filters for partial failures should be added. There are different ways to implement this. One way is to use all filters including additional filters at each time. The computational load of this approach is obviously heavy. Instead, a hierarchical structure of IMM estimators, referred to as hierarchical IMM (HIMM), can be used to reduce the computational complexity [22, 24, 98]. Fig. 4.2 illustrates such a hierarchical structure.

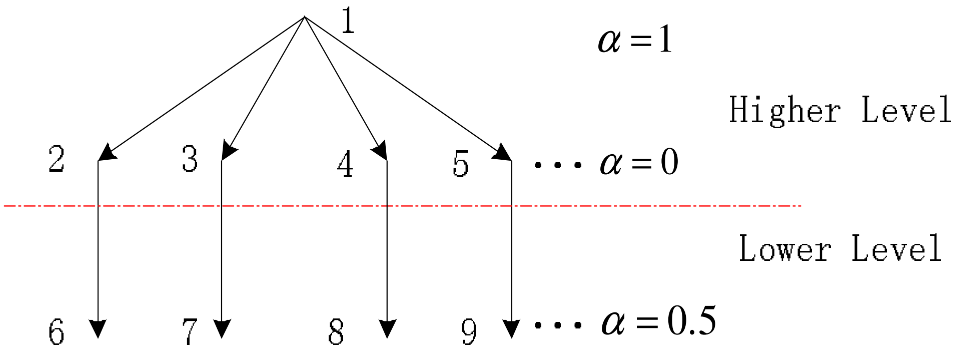


Figure 4.2: A hierarchical structure

The hierarchical structure consists of layers of multiple models. It has two levels in Fig. 4.2. At the higher level there are $M+1$ filters designed for one normal model ($\alpha = 1$) and M total sensor/actuator failure models ($\alpha = 0$). The higher level is used to detect which sensor or actuator has failed. Once a specific fault is declared at the higher level, the lower level is brought on line to run instead of the higher level. The lower level is to test how severe that specific fault is by estimating the effectiveness factor α. The lower level consists of M model sets, each designed for a specific sensor or actuator failure at the higher level. In Fig. 4.2, at the lower level each model set consists of one for a total failure ($\alpha = 0$), one for a half failure ($\alpha = 0.5$) and one for the normal operation ($\alpha = 1$). Including the normal model at the lower level allows the algorithm to change back to the higher level when there is no failure in fact. The total expectation theorem is used to fuse the parameter estimates

α_k^j at the lower level as well as the state estimates $\widehat{x}_{k|k}^j$, respectively:

$$\widehat{x}_{k|k} \triangleq E[x_k|z^k] = \sum_{j=1}^{n} \widehat{x}_{k|k}^j \mu_k^j \tag{4.4}$$

$$\widehat{\alpha}_k \triangleq E[\alpha_k|z^k] = \sum_{j=1}^{n} \alpha_k^j \mu_k^j \tag{4.5}$$

where $\widehat{x}_{k|k}^j$ is the state estimate from the jth elemental filter, μ_k^j is the corresponding model probability, α_k^j is the designed failure effectiveness factor for the model and n is the number of models in the model set in effect at time k at the lower level. The corresponding error covariances can be given similarly.

For the hierarchical MM estimators only one level is running at a time. Since the lower level is in effect only when the corresponding failure is detected at the higher level, it reduces the number of elemental filters running in parallel at each time. When a specific failure is declared at the higher level, corresponding filters at the lower level can be initialized automatically through the VSIMM[M_k, M_{k-1}] cycle [43]. Here for simplicity the estimate and covariance of the filters at the lower level are set equal to those of the corresponding filter at the higher level at the previous time. As such, the lower level filters have the same initial conditions. Their initial model probabilities are set equal.

4.4 The IM^{3}L Scheme

In the above, a predefined small model set for a failure is opened up to estimate the extent of failure when the failure is detected at the higher level. Next we develop an adaptive way to estimate partial failures using the maximum likelihood estimation (MLE) technique and then generate a new model based on the estimate [86]. Further, we propose techniques to handle multiple failures based on the proposed IM^{3}L algorithm. The IM^{3}L approach is systematic and can be generalized to many practical problems which contain sudden changes.

4.4.1 Benefit of Augmenting Models

For MM estimation with a fixed model set, the number of models needed to cover all possible failures could be very large. An idea is to augment the original model set by one or more adaptive models to cover various unknown faults. The augmented model should be statistically close to the true system mode. In chapter 3, a general approach to VSMM estimation was proposed by augmenting a variable set of models in the original set. A particular algorithm, referred to as expected-mode augmentation (EMA), was developed there, where models are generated adaptively online to match the expected value of the unknown true mode. Sec. 3.2.1 provides the theoretical justification that augmented models could be beneficial to MM estimation [53].

As stated in Sec. 3.2.2, in principle the augmented model can be estimated under any optimality criterion, which is usually problem dependent. Here based on the MLE criterion, we proposed a new approach to improve the FDI performance by augmenting an adaptive model with the maximum likelihood. We believe this approach fits the FDI problem well. It is well known that MLE is popular for parameter estimation, which in general needs an accurate parametric model. For the problem formulated in this context, the structure of different fault models is known and the effectiveness factor α is the only unknown parameter. This naturally leads us to pick MLE as a criterion to determine the type of the augmented model.

The proposed IM^{3}L combines the strength of two techniques: the MM estimation is used for fault detection, where each model has a known structure; the MLE is used for fault estimation with respect to an unknown parameter. This explains the power of the IM^{3}L algorithm.

4.4.2 IM^{3}L Algorithm

The IM^{3}L algorithm is described as follows: once a specific fault is determined, we estimate α by MLE, then a *new* failure model is generated and added into the model set based on the estimate $\widehat{\alpha}$ for that specific failure. This model is expected to improve state estimation if $\widehat{\alpha}$ is accurate so that the new model is statistically close to the truth.

MLE chooses as the estimate the "most likely" value of the true parameter given the observations by maximizing the likelihood function. Under the assumption of Kalman filtering, it is not difficult to obtain an analytical formula for $\widehat{\alpha}$. Assume at time k, a failure is detected by the IMM estimator, which is done in the same way as in the HIMM higher level. The likelihood function of $\alpha = [\alpha_1, ..., \alpha_k]'$ is

$$f(z^k|\alpha) = f(z_k|z^{k-1}, \alpha_k) \prod_{t=1}^{k-1} f(z_t|z^{t-1}, \alpha_t) \tag{4.6}$$

Since the fault occurs at time k and estimates of α at all previous times are assumed available, we only need to estimate α_k. As such, we only need to deal with the conditional likelihood function $f(z_k|z^{k-1}, \alpha_k)$. Assume

$$z_k \sim \mathcal{N}(\widehat{z}_{k|k-1}, S_k)$$

where $\widehat{z}_{k|k-1} = E[z_k|z^{k-1}]$ and S_k is the associated covariance. We have

$$f(z_k|z^{k-1}, \alpha_k) = \frac{1}{|2\pi S|^{1/2}} e^{-\frac{1}{2}(z_k - H_k \widehat{x}_{k|k-1})' S_k^{-1} (z_k - H_k \widehat{x}_{k|k-1})} \tag{4.7}$$

Then the MLE of α

$$\widehat{\alpha}_k = \arg\max_{\alpha_k} f(z_k|z^{k-1}, \alpha_k)$$

can be obtained (see Appendix B).

Both HIMM and IM^{3}L algorithms are based on VSMM. They have the same mechanism for fault detection. However, they differ in estimating the effectiveness factor α after the failure is detected. The HIMM uses an additional model set to estimate α, and so the estimation accuracy depends on the quantization level of that model set. The estimation of partial failures and state could be improved if more models were included at the lower level, which would cost more in computation. The IM^{3}L obtains $\widehat{\alpha}$ from MLE directly, which mainly depends on observations, and updates the state estimate by the new model based on $\widehat{\alpha}$

$$\widehat{x}_{k|k} \triangleq E[x_k|z^k] = \sum_{j=1}^{n} \widehat{x}^j_{k|k} \mu^j_k$$

Note that the updated state estimate from the new model based on $\widehat{\alpha}_k$ replaces the old one assuming $\alpha_k = 0$ (total failure).

4.4.3 IM^{3}L Algorithm for Multiple Failures

In practice, multiple failures exist in which a second failure occurs during the first failure. In this case, no model in the model set matches or is close to the truth. To maintain good performance, a variable structure MM is again applied to generate the "matched" model based on the IM^{3}L algorithm. Here only two different failures are considered as an example since the proposed approach can be extended to cases with more failures straightforwardly.

The key here is to generate new models based on diagnosis of the first failure by IM^{3}L: once the first failure is detected and α_1 (denoted for the first failure effectiveness) is estimated, the corresponding row of H (or column of G) matrix in all other *failure* sensor (actuator, respectively) models are changed to that of the identified failure model. As such, they are either expected identically to detect the second failure or jump back to the nominal operation since the normal model remains unchanged. If the second failure is detected, a new model is created upon $\hat{\alpha}_2$ to update the state estimation. Clearly the detection and estimation for the second failure are highly dependent on detection and $\hat{\alpha}_1$ of the first failure as well as initialization of the new filters. Their initial conditions could be set in two ways as follows.

- Set to those of the corresponding filter for the first failure at the previous time. It is reasonable since the second failure occurs after the first failure.

- Set to mixing estimate/covariance of the corresponding filter for the first failure at the current time. The IMM algorithm provides these estimates automatically. Simulation results show that the performance by two different initializations is close.

Although the HIMM scheme could also be extended to multiple failures, as proposed in [65, 72] where two levels of the AMM estimators are used for dual failures' detection only, it is more complicated and less effective to provide an integrated way for failure detection, identification and estimation. To achieve this goal, the HIMM has to use more models to cover possible failures, unlike the IM^{3}L which adaptively changes the model set based on data. Therefore, only the IM^{3}L scheme was implemented for multiple failures in our simulations reported later.

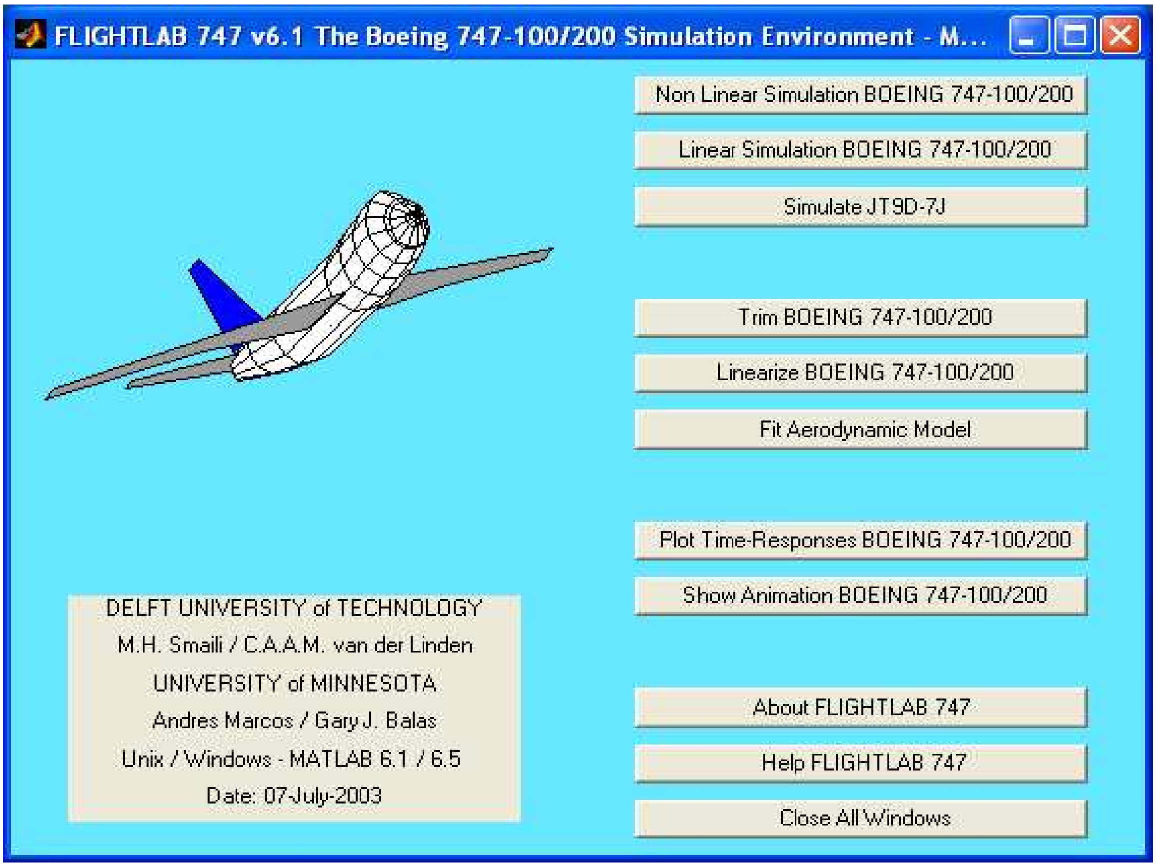
4.5 Boeing 747 Aircraft Simulator

A B747 simulator developed in [64] provides a tool to evaluate the performance of our algorithms more practically. It is an enhanced version of two previous programs: Delft University Aircraft Simulation and Analysis Tool (DASMAT) and Flight Lab747 (FTLAB747), which were originally developed at Delft University of Technology. A Boeing 747 (B747) is an intercontinental wide-body transport with four fan jet engines designed to operate from international airports. Due to its wide array of characteristics as a commercial airplane, the B747 is an ideal benchmark to design and test fault detection and identification algorithms [64]. In this section main menu functions are summarized first. Then how to generate actuator failures is briefly addressed. This simulator provides a practical base to evaluate performance of different approaches.

4.5.1 Main Functions of the Simulator

There are four main functions of the simulator: trimming, linearization, simulation and simulation analysis (see Fig. 4.3).

- Trim the aircraft. This provides an equilibrium point for the aircraft based on a user-defined initial guess for the aircraft control and states. There are six possible trim modes: straight-and-level trim, pushover-pullup, level turn, thrust-stabilizer turn, beta trim and specific power turn.



FLIGHTLAB 747 v6.1 The Boeing 747-100/200 Simulation Environment - M...

DELFT UNIVERSITY of TECHNOLOGY
M.H. Smaili / C.A.A.M. van der Linden
UNIVERSITY of MINNESOTA
Andres Marcos / Gary J. Balas
Unix / Windows - MATLAB 6.1 / 6.5
Date: 07-July-2003

Non Linear Simulation BOEING 747-100/200
Linear Simulation BOEING 747-100/200
Simulate JT9D-7J
Trim BOEING 747-100/200
Linearize BOEING 747-100/200
Fit Aerodynamic Model
Plot Time-Responses BOEING 747-100/200
Show Animation BOEING 747-100/200
About FLIGHTLAB 747
Help FLIGHTLAB 747
Close All Windows

Figure 4.3: Boeing747 simulator

- Linearization. The aircraft can be linearized at the equilibrium point to analyze the aircraft's modal characteristics. The linearized model is given in the state-space form.

- Nonlinear or linear simulation. It provides open loop or closed loop functions to simulate the nonlinear or linear aircraft models given initial conditions. The linear simulation provides a tool to validate a linearized model by a reliable comparison with nonlinear simulation results.

- Analyze simulation results. It provides output plots of the simulation and visualization of the flight from different perspectives.

Table 4.1: States for linear matrices

	$\mathbf{p}_b$	$\mathbf{q}_b$	$\mathbf{r}_b$	$\mathbf{V}$	α	β	ϕ	θ	ψ	$\mathbf{h}_e$	$\mathbf{x}_e$	$\mathbf{y}_e$
Long.		X		X	X			X		X	X	
Lat./Dir.	X		X			X	X		X			X
Total	X	X	X	X	X	X	X	X	X	X	X	X

4.5.2 Linearization of the Boeing 747 model

The linearized model of Boeing 747 can be described by

$$\dot{x}(t) = Ax(t) + Bu(t) + \xi(t) \tag{4.8}$$

$$z(t) = Cx(t) + v(t) \tag{4.9}$$

where is x is the state, u is the control input, and A, B, C are system matrices.

In the simulator, there are three types of models based on three different motions available for users to linearize the aircraft Boeing 747. The symmetric model (6 states) describes longitudinal motion while the asymmetric model (6 states) describes lateral/directional motion. The total model including 12 states is the complete nonlinear model. The determination of models defines the number of rows for the A and B matrices in the state-space model (1). Table 4.1 shows the states used for each type of motion, where X indicates the state is used.

In addition, the simulator allows the possibility of obtaining linearized models with respect to both types of inputs: pilot and control inputs. There is a transformation between the two inputs. Once the input signal is selected, the user defines detailed input information, i.e., specifies input components with non-zero signals. Table 4.2 shows possible inputs for

Table 4.2: Inputs for linear matrices

	Pilot Inputs	Control Inputs
Long	$\delta_{column}, \delta_{stab}, Tn1, Tn2, Tn3, Tn4$	$\delta_{eil}, \delta_{ih}, Tn1, Tn2, Tn3, Tn4$
Lat./Dir.	$\delta_{wheel}, \delta_{pedal}, Tn1 \sim Tn4$	$\delta_{ail}, \delta_{ru}, Tn1 \sim Tn4$
Total	$\delta_{column}, \delta_{wheel}, \delta_{pedal}, \delta_{stab}, Tn1 \sim Tn4$	$\delta_{eil}, \delta_{ail}, \delta_{ru}, \delta_{ih}, Tn1 \sim Tn4$

these matrices based on the different types of the control inputs and aircraft motion.

After selecting input signals and type of motions, the user defines observations. There are several observation groups available and users can choose interesting ones for their applications. The determination of the outputs defines the number of rows for the C and D matrices in the state-space model (2).

4.5.3 Boeing 747 Aircraft Model

The longitudinal model of the B747 aircraft was chosen for simulations. It was obtained by linearizing the longitudinal dynamics of the B747 at a straight and level flight condition at 7000 m altitude and 241 m/s velocity [85, 95]. The straight and level flight refers to zero flight path angle (FPA=0 deg).

States are pitch rate q (rad/s), total velocity V (m/s), angle of attack α (rad), pitch angle θ (rad) and altitude h_e (m). Measurements are pitch rate q, total velocity V, angle of attack α, pitch angle θ, acceleration $\dot{V}$ (m/s^2), and flight path angle γ (rad). Control inputs are elevator deflection δ_e (rad) and stabilizer deflection δ_{st} (rad). The model matrices are

$$A = \begin{bmatrix} -0.7284 & -0.0005 & -1.2025 & 0 & 0 \\ -0.0839 & -0.0055 & 6.0078 & -9.7850 & 0.0001 \\ 1.0019 & -0.0004 & -0.5151 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -240.996 & 240.996 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -0.0839 & -0.0055 & 6.0078 & -9.7850 & 0.0001 \\ 0 & 0 & -1 & 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} -1.871 & -4.6099 \\ 0 & 0 \\ -0.036 & -0.0944 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

4.5.4 VTOL aircraft model

In our study, we also use a longitudinal vertical takeoff and landing (VTOL) aircraft model [74] for performance evaluation. The state $x = [V_h,\, V_v,\, q,\, \theta]'$ consists of horizontal velocity V_h (m/s), vertical velocity V_v (m/s), pitch rate q (rad/s), and pitch angle θ (rad). The control inputs $u = [\delta_c,\, \delta_l]'$ are collective pitch control δ_c and longitudinal cyclic pitch control δ_l. The system matrices are

$$A = \begin{bmatrix} -0.0366 & 0.0271 & 0.0188 & -0.4555 \\ 0.0482 & -1.01 & 0.0024 & -4.0208 \\ 0.1002 & 0.3681 & -0.707 & 1.420 \\ 0.0 & 0.0 & 1.0 & 0.0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.4422 & 0.1761 \\ 3.5446 & -7.5922 \\ -5.52 & 4.49 \\ 0.0 & 0.0 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

For two flight models, discretizing (4.8) and (4.9) yields

$$x_{k+1} = Fx_k + Gu_k + w_k \tag{4.10}$$

$$z_k = Hx_k + v_k \tag{4.11}$$

where $F = e^{AT}$, $G = (\int_0^T e^{AT} d\tau)B,\ \ H = C,$ and T is the sampling period.

4.6 Performance Evaluation

In this section, two types of aircraft were used to demonstrate the performance of our proposed FDI schemes. A longitudinal vertical takeoff and landing (VTOL) aircraft model is used for detection and diagnosis of sensor failures while a Boeing 747-100/200 is used for actuator failures. The sampling time $T = 0.1$s.

4.6.1 Performance Indices

Different approaches are evaluated in terms of detection and estimation performance. The following detection performance measures are used in this work: false alarm (FA), missed detection (MD), percentages of correct detection and identification (CDI), incorrect fault identification (IFI), no mode detection (NM) [104]. A CDI is obtained if the model that is closest to the system mode (normal or fault mode) in effect at the given time has a probability that is largest and exceeds the preset threshold μ_T. An IFI is obtained if the model with a probability over μ_T is not the one closest to the actual false mode at the given time. An FA is obtained if the model with a probability larger than μ_T is not the normal mode in effect at the given time. An MD is obtained if the normal model has the highest probability larger than μ_T while the system has a fault. It is indecisive (NM) if no model has a probability larger than μ_T. Clearly, for a particular approach, the larger CDI is, the better the approach performs. Fig 4.4 uses three models to illustrate relationships among these performance indices.

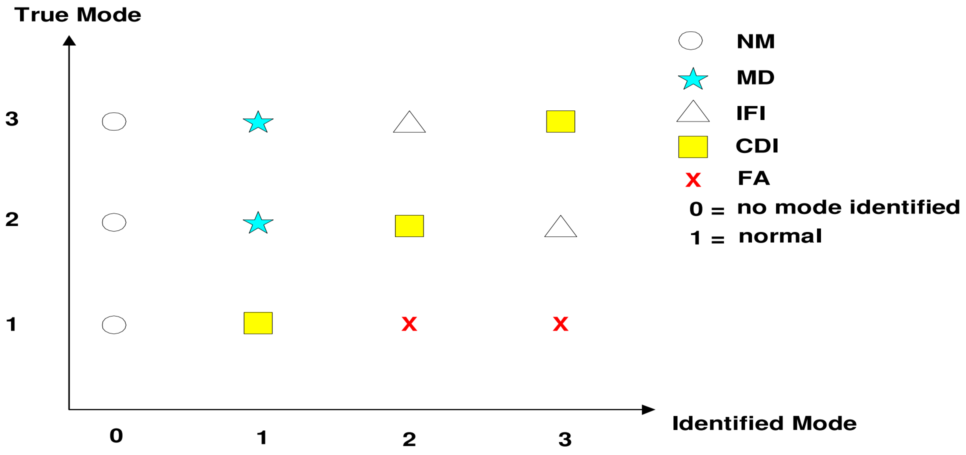


Figure 4.4: Relationship of FDI performance indices

The computational complexity of different algorithms is evaluated by the ratio of CPU processing time per iteration with respect to the HAMM. The root-mean-square error (RMSE) of the state and effectiveness factor at time k are used to evaluate estimation performance, respectively

$$\text{RMSE}(\widehat{x}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_k^i - \widehat{x}_k^i)'(x_k^i - \widehat{x}_k^i)} \tag{4.12}$$

$$\text{RMSE}(\widehat{\alpha}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\alpha_k^i - \widehat{\alpha}_k^i)^2} \tag{4.13}$$

where N is the number of Monte Carlo runs, and the superscript i stands for quantities on run i.

4.6.2 GLRT Detector for FDI

Conventional innovation-based approaches include the generalized likelihood ratio (GLR) test, the simple chi-square test and the sequential probability ratio test. Among these, the GLR is the most popular and has been successfully applied to failure detection, especially for actuator failures modeled by additive changes [99, 100]. Based on a single Kalman filter under normal operation used for residual generation, the GLR calculates the likelihood ratio of each failure hypothesis over no failure, which is usually unknown due to unknown failure size and change time. The key to the GLRT technique is to replace an unknown likelihood with its most probable likelihood. A full implementation of the GLR requires a linearly increasing number of parallel filters, and thus is infeasible for real applications. Instead, a finite window based GLR is widely used [100]. The GLR declares a failure if the test statistic exceeds a preset threshold determined by the desired decision error rates.

Here for the GLR detector, failures are modeled the same way as described in Sec 4.2. As a result, for actuator failures the batch (window-based) residuals from a normal filter can be deemed a linear measurement of α, therefore the MLE of $\hat{\alpha}$ can be obtained as a least squares solution under the linear Gaussian assumption [100]. But for sensor failures, there is no explicit MLE solution available because of the nonlinear relation. We obtain $\hat{\alpha}$ based on the marginal likelihood function as derived in Appendix B for IM^{3}L instead of numerical algorithms.

After a failure has been detected by the GLR detector, the filter estimate and covariance are updated [100]. Once the update is performed, the GLR can be used to detect further failures, i.e., allowing the detection of sequential failures. Since the GLR is based on a single model, it is difficult to deal with multiple failures. We will compare our approaches with the GLR only for sequential sensor or actuator failures.

4.6.3 Simulation Results

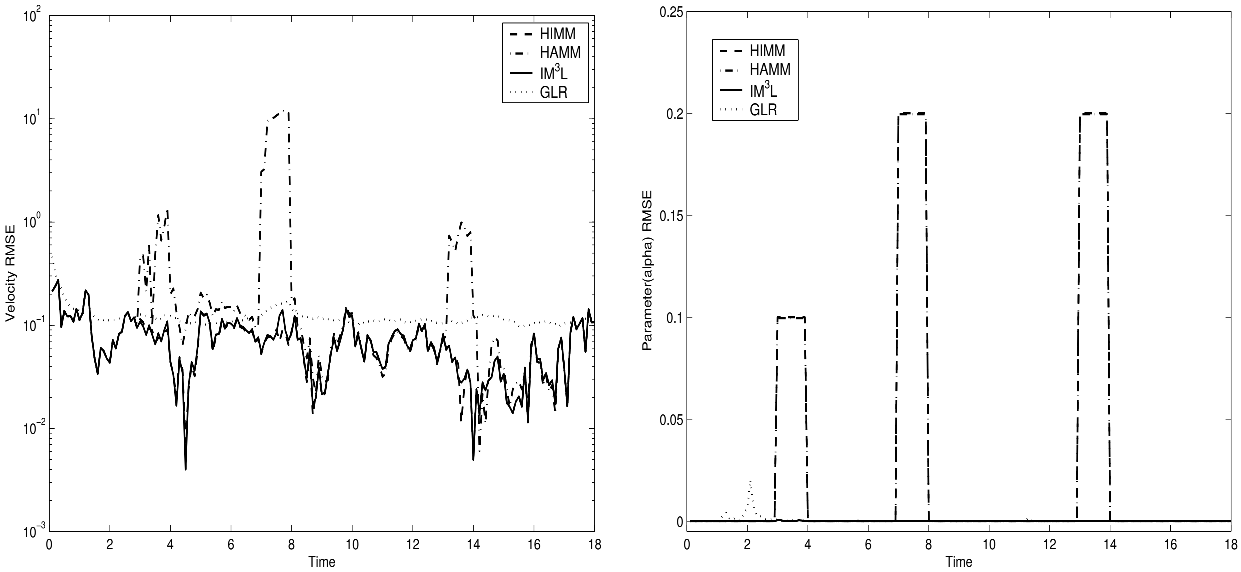
All results presented here are averages over 100 Monte Carlo runs. HIMM uses a hierarchical with a numerically robust IMM implementation [56]. HAMM represents a hierarchical autonomous MM algorithm with a 10^{-3} lower bound for each model probability. The transition probability matrices were designed for different scenarios. The detection threshold used for MM-based approaches is 0.9 while the threshold for the GLR corresponds to a 5% false alarm rate. The window size was chosen as 5.

Scenario 1. Sequential partial sensor failures of VTOL aircraft

For the HIMM and HAMM, a total of nine models were used (one for the normal, four for total failures plus four for half failures). For the IM^{3}L, a total of five models were used (four for total failures plus one for the normal). The following parameters were used: noise covariances $Q = (0.01)^2 I$ and $R = (0.2)^2 I$, initial state $x_0 = [250,\ 50,\ 10,\ 8]'$, control input

$u = [100,\ 100]'$.

Case 1 tested severe failures (with a small true effectiveness $\alpha \leq 20\%$). It included a 90% horizontal velocity V_h failure during $[30, 39]$ (i.e., 90% V_h failure means $\alpha = 10\%$ between $k = 30$ and $k = 39$), a 80% vertical velocity V_v failure during $[70, 79]$, and a 80% pitch rate q failure during $[130, 139]$. Fig. 4.5 shows the RMS velocity errors (i.e., the RMS errors of $[V_h,V_v]'$) and RMS errors of $\widehat{\alpha}$. In this case all algorithms identified faults correctly, and the difference for the RMS velocity error is not large. The HIMM and HAMM have the same RMS α errors, which are larger than those of the GLR and IM^{3}L. The IM^{3}L outperformed other approaches in terms of both RMS errors. Clearly the α errors in the GLR were caused by the false alarm. Since the same technique is used to estimate α, it is not surprising that the GLR achieved almost the same estimation accuracy as the IM^{3}L. Similar comparison results were observed for pitch rate failure, which are not shown for saving space. Such RMS failure errors will be omitted later if there are no abnormal results obtained.

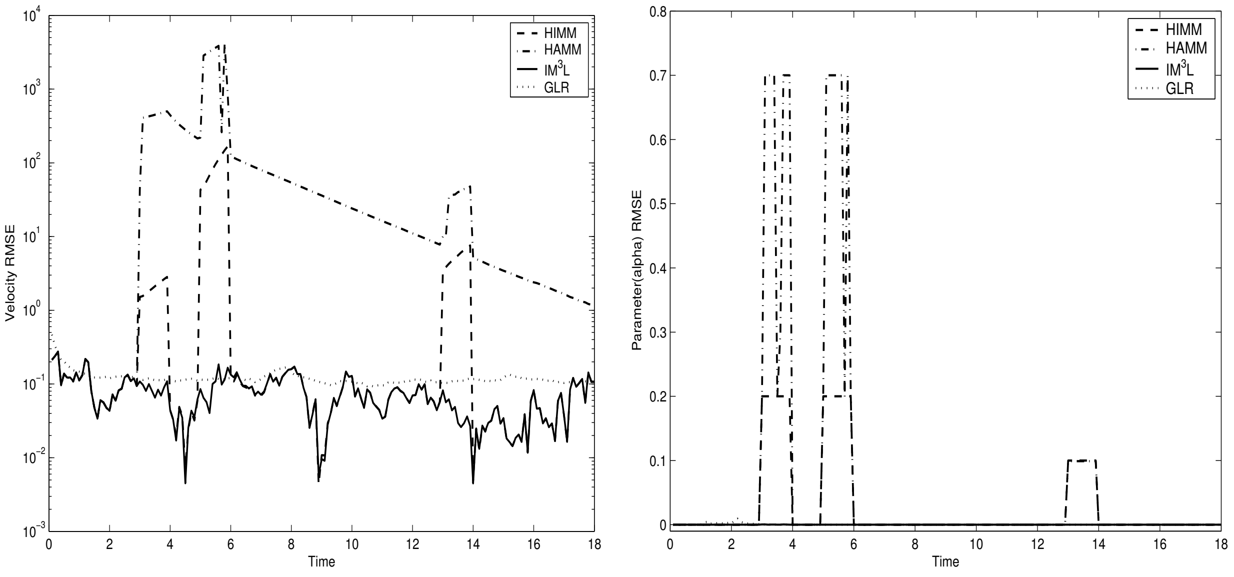


(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

Figure 4.5: Case 1 – sequential sensor failures (severe, total failure models)

Case 2 included a 70% horizontal velocity failure during $[30, 39]$, a 70% vertical velocity failure during $[50, 59]$, and a 60% pitch rate failure during $[130, 139]$. Fig. 4.6 shows the RMS velocity errors and $\widehat{\alpha}$ errors. Fig. 4.5 and Fig. 4.6 shows that as α increases, the performance of the HIMM and especially the HAMM deteriorates.



(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

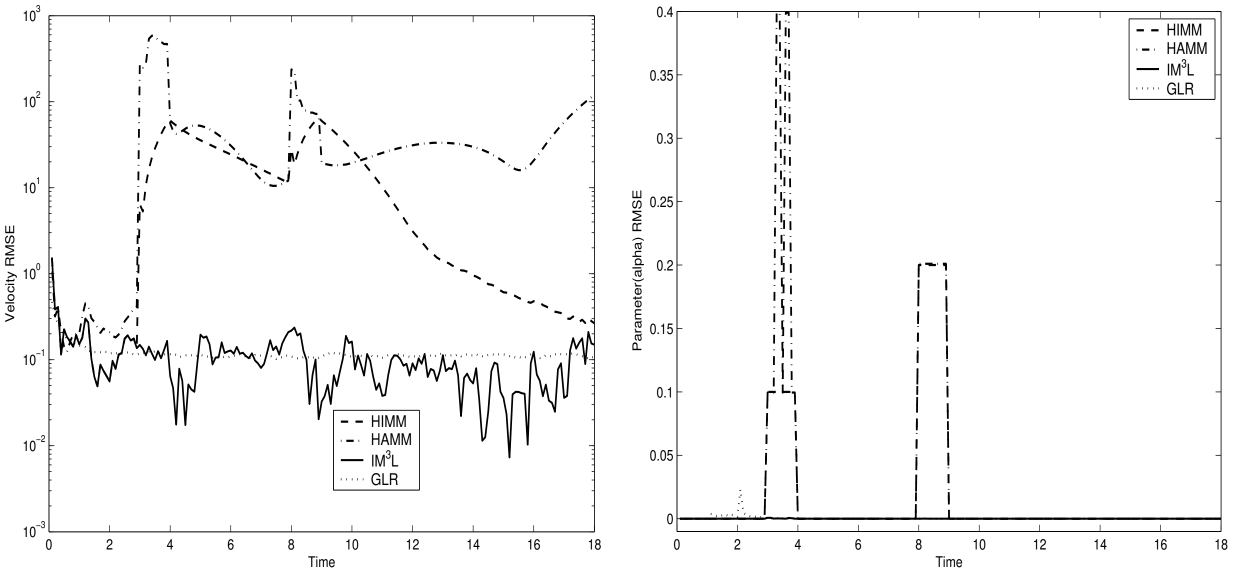
Figure 4.6: Case 2 – sequential sensor failures (mild, total failure models)

Both Case 1 and 2 used total failure and normal models at the higher level to detect sensor failures (although the IM^{3}L has only one level). However, it was observed that the total failure model has a limited detection range (i.e., $\alpha < 0.5$), outside of which the fault cannot be detected. Therefore, Case 3 was designed to test how the model set could affect the detection and estimation performance, where total failure models at the higher level were replaced with half failure models under a 90% V_h failure during $[30, 39]$ and a 30% q failure during $[80, 89]$. Fig. 4.7 shows the performance comparison results. Table 4.3 presents a comparison of the detection range for the HIMM with different model set design, where HIMM1 represents the HIMM estimator with total failure models at the higher level and

Table 4.3: Detection ranges of HIMM with different model set design for sensor failures

α	Total Failure	Half Failure	No Failure
HIMM1	$[0, 0.3)$	$[0.3, 0.5)$	$[0.5, 1]$
HIMM2	$[0, 0.25]$	$(0.25, 0.75)$	$[0.75, 1]$

HIMM2 represents one with half failure models instead. Clearly using half failure model has a greater detection range but the HIMM and especially HAMM perform worse in terms of the RMS velocity errors, as shown in Fig. 4.7. In this case, the IM^{3}L still has excellent performance even with different model set designs. Similar detection results were observed for the HAMM estimators. Since the HIMM and IM^{3}L have the same mechanism for fault detection, they have the same detection range. A single model based GLR algorithm does not have such a design issue.

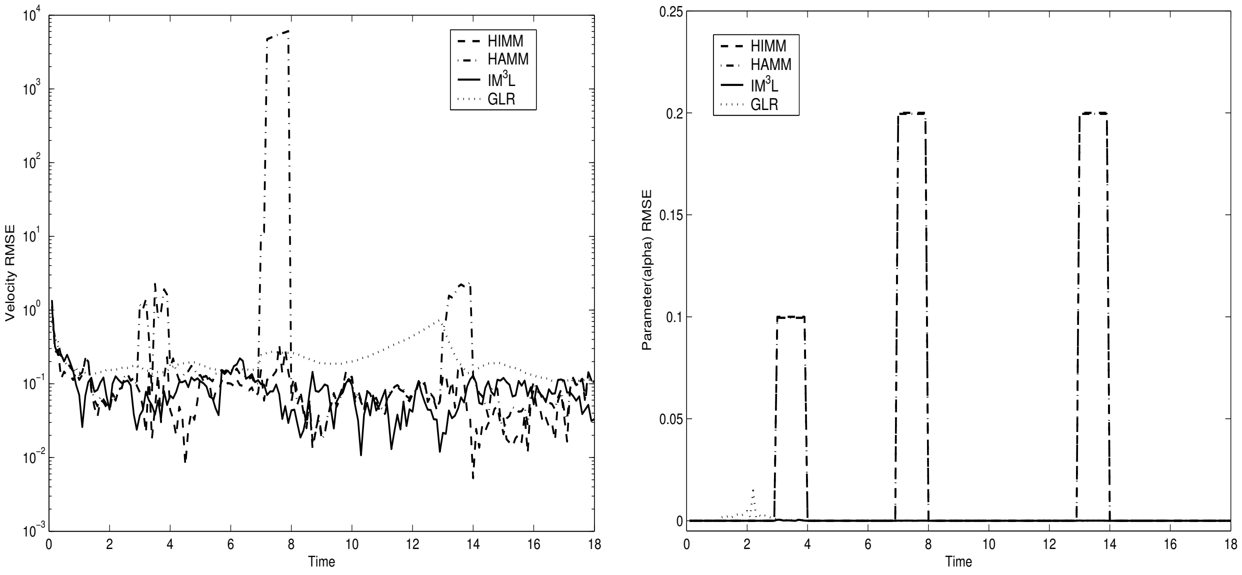


(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

Figure 4.7: Case 3 – sequential sensor failures (half failure models)

Case 4 was tested to evaluate the performance impact due to the uncertainties in noise statistics. It differs from Case 1 in that the noise matrices Q and R used by filters are 20 times the true ones. Fig. 4.8 shows the RMS velocity error and $\widehat{\alpha}$ errors. Cases with other uncertainties in noise statistics were also simulated. It shows that the HIMM and IM^{3}L maintain their estimation accuracy while the performance of the GLR and HAMM deteriorates. Table 4.4 shows the FDI results for all tested cases based on the partial sensor failures.



(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

Figure 4.8: Case 4 – sequential sensor failures (robustness)

Scenario 2. Multiple sensor failures of VTOL aircraft

Multiple failures were investigated only by the IM^{3}L, as explained in Sec. 4.2. Fig. 4.9 shows the performance comparison over four tested cases for the RMS velocity errors and $\hat{\alpha}_2$ errors. The FDI results are shown in Table 4.5. Case 1 tested severe failures. It included 90% horizontal velocity failure during $[30, 40]$ and $[80, 90]$, an 80% vertical velocity failure

Table 4.4: FDI results for sequential sensor failures (VTOL)

	%	MD	FA	CDI	NMD	IFI
Case 1	GLR	0	1.03	98.97	0	0
	HIMM	0	0	100	0	0
	HAMM	0	0	100	0	0
	IM^{3}L	0	0	100	0	0
Case 2	GLR	0	1.13	98.87	0	0
	HIMM	0	0	100	0	0
	HAMM	8.33	0	90.56	0	1.11
	IM^{3}L	0	0	100	0	0
Case 3	GLR	0	1.16	98.84	0	0
	HIMM	0	0	99.44	0.56	0
	HAMM	0	0.12	98.17	0.04	2.67
	IM^{3}L	0	0	99.44	0.56	0
Case 4	GLR	0	1.70	98.30	0	0
	HIMM	0	0	99.92	0.08	0
	HAMM	0	0	95.56	0	4.44
	IM^{3}L	0	0	99.92	0.08	0

MD=missed detection, FA=False alarm, CDI=correct detection and identification

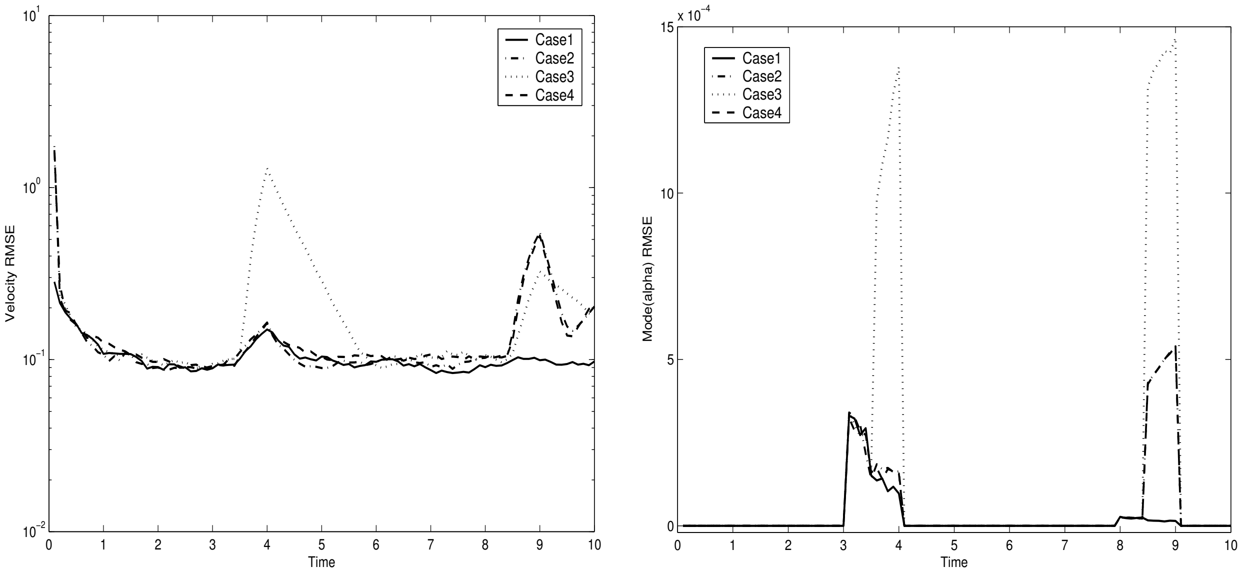
NMD=no mode detection, IFI=incorrect detection and identification

during $[35, 40]$, and a 60% pitch angle failure during $[85, 90]$. Case 2 differs from Case 1 in that total failure models were replaced by half failure models. Both cases were tested for relatively severe failures. It shows that both model sets have close performance except that state estimation errors with half failure models are slightly greater during the second failure.

Table 4.5: FDI results for multiple sensor failures (IM^{3}L, VTOL)

%	MD	FA	CDI	NMD	IFI
Case1	0	0	100	0	0
Case2	0	0.01	99	0.99	0
Case3	0	0	99	1	0
Case4	0	0.01	98.03	1.96	0

Case 3 tested mild multiple failures with half failure models, a 60% horizontal velocity failure during $[30, 40]$ and $[80, 90]$, a 30% vertical velocity failure during $[35, 40]$, and a 40% pitch rate failure during $[85, 90]$. Case 4 was simulated to evaluate the robustness of the algorithms as for Case 2. It shows that estimation errors including $\hat{x}$ and $\hat{\alpha}$ increase even though the detection rate is still high.



(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

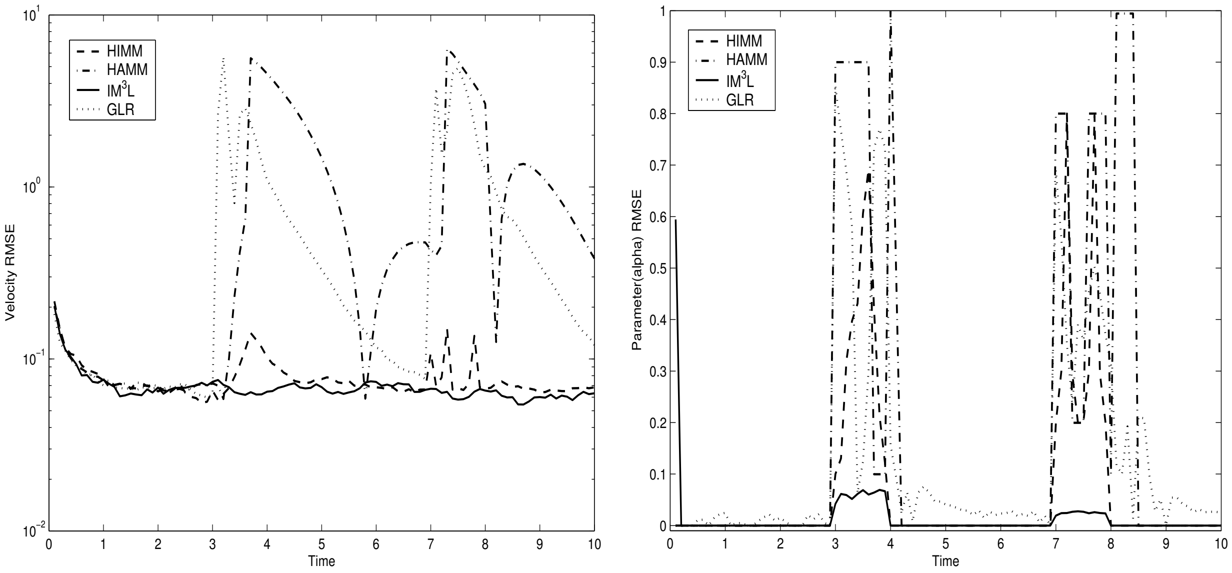
Figure 4.9: Multiple sensor failures

As observed in the sequential failures, detection ranges for multiple failures become greater when total failure models are replaced by half failure models, for example, detection ranges of tested first and second failures both change from $[0, 0.5)$ to $[0, 0.75)$.

Scenario 3. Sequential partial actuator failures of B747 aircraft

There are two hypothesized actuator failures for the selected B747 model (elevator or stabilizer failure). The HIMM and HAMM used a total of five models, respectively (one for the normal, two for total failures and two for half failures), and the IM^{3}L used three models (two total failure models plus one for the normal). The following parameters were used: control input $u = [25\ \ 25]'$, $Q = (0.01)^2I$, $R = (0.2)^2I$, and initial state $x_0 = [0.01\ \ 242\ \ 0.03\ 0.03\ 7001]'$.

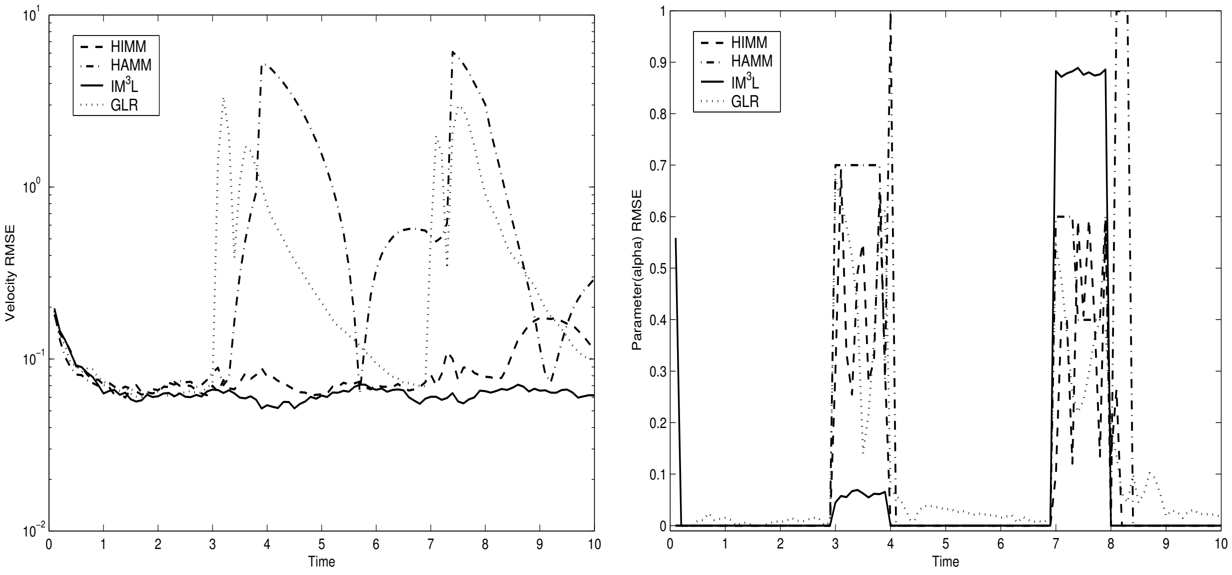
Case 1 was designed to test serious failures, which included a 90% elevator deflection δ_e failure during $[30, 39]$, and an 80% stabilizer deflection δ_{st} failure during $[70, 79]$. Fig. 4.10 shows the RMS velocity errors and the RMS errors of $\widehat{\alpha}$. Case 2 was tested for mild failures, which included a 70% δ_e failure during $[30, 39]$ and a 60% δ_{st} failure during $[70, 79]$. Its estimation results are shown in Fig. 4.11. Case 1 tells that even when actuator failures are severe, the GLR, HIMM and HAMM failed to detect faults overall correctly while IM^{3}L did. For the mild failures in Case 2, only the IM^{3}L correctly and timely detected the first failure, and all the other three detectors incorrectly identified the second δ_{st} failure as δ_e failure. Therefore, the RMS α errors for second δ_{st} failure are misleading since α was estimated based on an incorrect failure model due to the wrong decision. The high false alarm rate in the GLR was caused by inaccurate initial state condition for sequential failures. In both cases, the HAMM has the worst estimation accuracy for the state and α while the IM^{3}L is the best. The HIMM outperformed the GLR for state estimation. Note the tremendous improvement of the IM^{3}L in the $\widehat{\alpha}$ errors over the other three algorithms.



(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

Figure 4.10: Case 1 – sequential actuator failures (severe, total failure models)



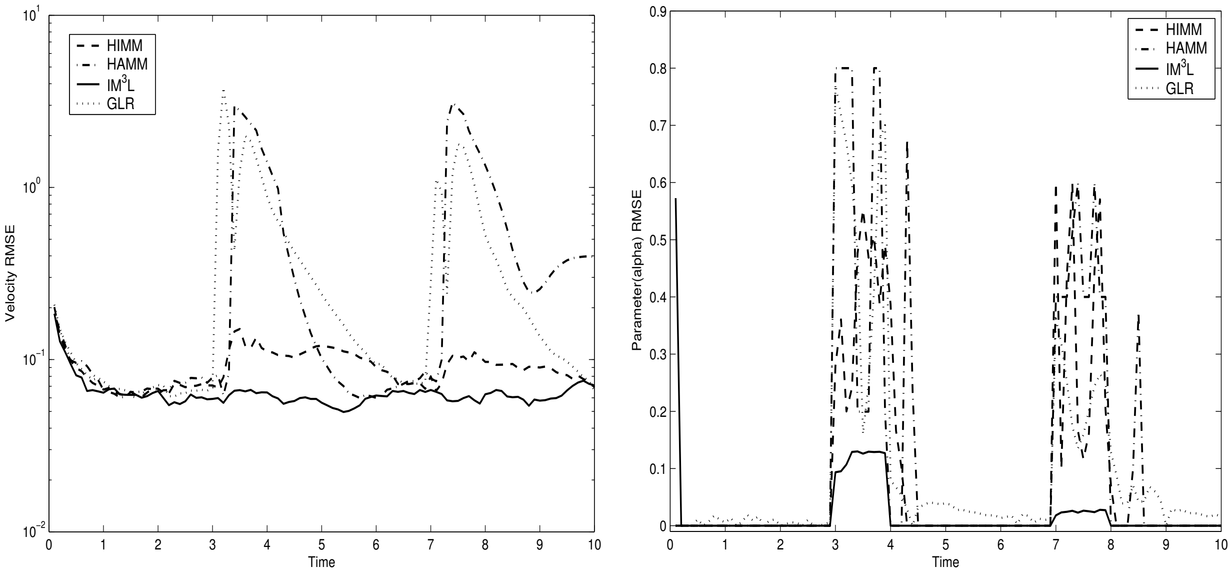
(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

Figure 4.11: Case 2 – sequential actuator failures (mild, total failure models)

More simulation results show that for the MM based algorithms with total failure models at the higher level, the detection range of actuator failures was much smaller than that of sensor failures. Further, unlike sensor failures, which have basically the same detection ranges, different actuator failures have different detection ranges. For example, the detection range of elevator failure by the IM^{3}L algorithm is $[0, 0.5)$ while that of the stabilizer failure is $[0, 0.3)$. For the B747 aircraft, stabilizer failures are more difficult to detect than elevator failures.

Case 3 was to evaluate the performance when total failure models are replaced by half failure models at the higher level. It included an 80% δ_e failure during $[30, 39]$ and a 40% δ_{st} failure during $[70, 79]$. Compared to Case 1 and 2, it shows that there is little improvement for detection ranges of the HIMM and HAMM. However, the IM^{3}L had a greater detection range for both failures ($\delta_e \in [0, 0.7)$, $\delta_{st} \in [0, 0.5)$) with actuate estimation, shown in Fig. 4.12. Clearly the IM^{3}L outperforms the other three algorithms significantly.

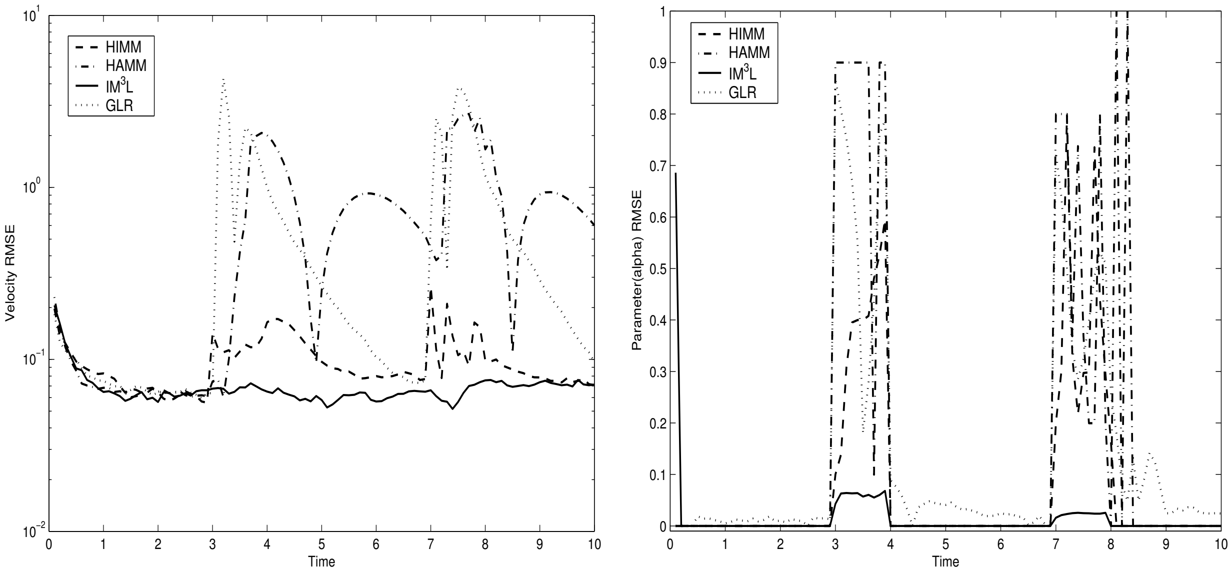


(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

Figure 4.12: Case 3 – sequential actuator failures (half failure models)

Case 4 was simulated to evaluate the robustness of the proposed algorithms, which differs from Case 1 in that the noise matrices Q and R used for the filter are 20 times the true ones. Fig. 4.13 presents the RMS velocity errors and $\widehat{\alpha}$ errors. The results show that the IM^{3}L approach is more robust than the other three algorithms to uncertainties in the noise statistics in terms of estimation accuracy. Table 4.6 shows the FDI results for all four cases of partial actuator failures. As the simulation results demonstrated, in general actuator failures are more difficult to detect than sensor failures, since sensor failures directly affect measurements while the impact of actuator failures takes time to appear in measurements.



(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

Figure 4.13: Case 4 – sequential actuator failures (robustness)

Scenario 4. Multiple actuator failures of B747 aircraft

Case 1 was tested for an 80% elevator failure during $[30, 40]$ and a 70% stabilizer failure during $[35, 40]$ with total failure models. Case 2 differs from Case 1 in that total failure models were replaced by half failure models. It is interesting to see that, compared with Case 1, state estimation accuracy in Case 2 deteriorates slightly but α estimation improves. This is because the half failure δ_{st} model of the B747 aircraft has a greater detection range

Table 4.6: FDI results for sequential actuator failures (B747)

	%	MD	FA	CDI	NMD	IFI
Case 1	GLR	0	17.19	75.57	0	7.24
	HIMM	5.05	0	93.44	1.51	0
	HAMM	14	5.97	76.02	1.01	3
	IM^{3}L	0	0	99	1	0
Case 2	GLR	0	18.5	74.49	0	7.01
	HIMM	11.4	0.04	83.94	1.55	3.07
	HAMM	14.07	3	76.91	1.02	5
	IM^{3}L	0	0	88.99	1.01	10
Case 3	GLR	0	17.74	75.36	0	6.9
	HIMM	8.63	0.21	87.35	1.47	2.34
	HAMM	13.48	0.72	81.82	1.46	2.52
	IM^{3}L	0	0	98.44	1.02	0.54
Case 4	GLR	0.6	20.07	72.49	0	6.84
	HIMM	5.01	0	93.36	1.63	0
	HAMM	13.93	2	78	1.23	4.84
	IM^{3}L	0	0	98.95	1.05	0

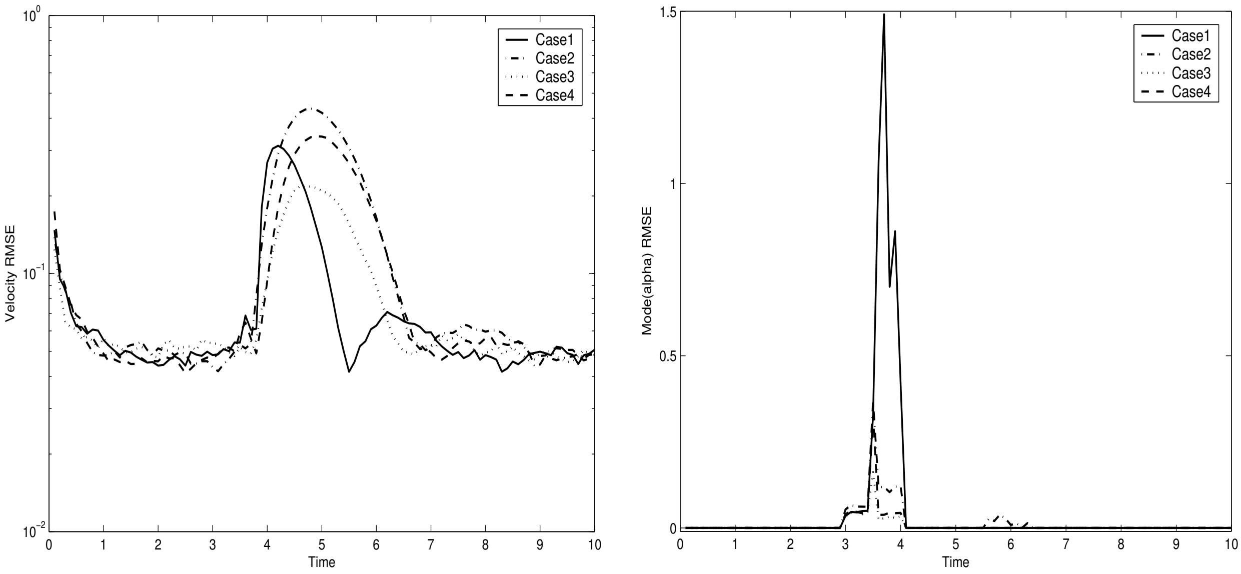
than the total failure model, which improves the accuracy of $\hat{\alpha}$. However, due to its greater detection range, the half failure model responds slowly, which leads to slightly larger state estimation errors. Case 3 with half failure models was simulated for mild failures, which included a 40% δ_e failure during $[30, 40]$ and a 60% δ_{st} failure during $[35, 40]$. It shows that the IM^{3}L still has good performance. Case 4 was designed to test the robustness relative to for Case 2. Fig. 4.14 show the performance comparison. The FDI results are shown in

Table 4.7: FDI results for multiple actuator failures (IM^{3}L,B747)

%	MD	FA	CDI	NMD	IFI
Case1	1	0.05	96.14	1.02	1.79
Case2	0	0.04	98.74	1.11	0.11
Case3	1	0	97	1	1
Case4	0	0	97.27	2.73	0

Table 4.7.

More cases were simulated for multiple failures of the VTOL and B747 aircraft. The simulation results show that in general the IM^{3}L sacrifices the estimation accuracy more in multiple actuator failures than in sensor failures. The IM^{3}L consistently beats the GLR, HIMM and HAMM in terms of estimation accuracy and robustness.



(a) RMS velocity errors

(b) RMS $\widehat{\alpha}$ errors

Figure 4.14: Multiple actuator failures

Table 4.8: Computational complexity of different algorithms

$CPUratio$	HAMM	GLR	HIMM	IM^{3}L
sensor failures	1	0.54	1.30	1.32
actuator failures	1	3.7	1.32	1.42

The computational loads of the algorithms evaluated in terms of relative CPU processing time ratios per iteration with respect to the HAMM are summarized in Table 4.8.

4.6.4 Discussion

Overall the IM^{3}L outperforms significantly the GLR, HIMM and HAMM in all tested scenarios. Through online adaptation, the IM^{3}L provides accurate estimates of the extent of failure and robust state estimates even during failures. The reason is that the new model based on $\widehat{\alpha}$ is statistically close to the truth, which is verified by the histogram of measurement residuals (not shown due to space limitation). For the HIMM or HAMM, when the designed failure models do not match the truth well, they cannot provide good estimates of the effectiveness factor and thus good state estimates. The detection performance of the GLR depends on the accuracy of the MLE $\widehat{\alpha}$. As a result, many design issues arise such as window size, MLE estimator and false alarm rate. The HIMM has a lighter computational load than the IM^{3}L, and their computational load is slightly more than that of the HAMM. The standard window-based GLR detector for actuator failures is computationally intensive while GLR for sensor failures uses marginal likelihood information and thus saves computation.

To obtain satisfactory performance, the HAMM has to be tuned carefully for the higher level models to take advantage of the information from the lower level. The IMM-based approach (HIMM and IM^{3}L) does not need such heuristic adjustments. Compared with the GLR whose performance depends highly on detection threshold, the performance of no MM based algorithms is sensitive to the choice of threshold. Similar to other hypothesis tests, the GLR is suitable for single failure detection since it is based on a single model under normal operation. Once a fault occurs, in order to detect sequential failures, the hypotheses must change or a state update procedure need to be performed in the GLR. This may result in a large false alarm due to inaccurate initialization after detection, as observed in actuator failures. The superior interactive structure of the IMM-FDI estimators does not have difficulty for the more general fault situations, including sequential or multiple failures.

As mentioned before, soft sensor failures can also be modeled by changing the mean of the measurement noise or R matrix. For failures caused by noise mean shift, some scenarios were tested on sensor failures of the VTOL. The results show that the IM^{3}L can detect and estimate increased noise-mean sensor failures when the noise-mean increases within some range. Model set design for this case is more difficult due to less prior information of the mean changes. In addition, the proposed IM^{3}L approach is also applicable to the detection of faults caused by a change in the F matrix (i.e., change in physical components). However, the application of the IM^{3}L to such failures by changes in the R or F matrix is more complicated and not addressed in this thesis.

In the HIMM approach, while the higher level is running consecutively, the lower level is opened to update estimates of α and state only when a fault is detected at the higher level. An alternative was also implemented: Once the lower level is opened, it remains open until the normal model has the highest model probability over the threshold, then the lower level is closed and estimation goes back to the higher level. The simulations showed that the performance difference between the two ways of implementation depends on the scenario and model set design.

The IM^{3}L successfully estimates the failure severity as well as the state when partial failures occur sequentially or concurrently. This is based on the assumption that abrupt system failures occur infrequently. Otherwise the MLE $\widehat{\alpha}$ may not be accurate. It should be emphasized that the idea of the IM^{3}L algorithm could also be applied to a nonlinear system except that the analytical solution for the unknown parameter may not be available and instead we may have to resort to numerical solutions such as the EM algorithm. Furthermore, the research here based on the IM^{3}L focuses on sensor failures or actuator failures separately. In fact it is applicable to simultaneous sensor and actuator failures by designing a complete model set containing all failure hypotheses of interest.

The IM^{3}L is particularly good for problems such as FDI, where the system can be modeled by one or more models and the structure of each model is different but known. It may not fit other problems well where the space of the true mode is continuous, for example, the application of target tracking. We implemented IM^{3}L for the same tracking problem formulated in [53], we found that the IM^{3}L performed worse than the EMA in terms of the tracking accuracy. As stated in [53], the EMA algorithm proposed is good for applications with a continuous mode space. In target tracking, there is no structure change between different models due to similar maneuver structure for target motions. As a result, in practice we need to analyze the problem well in order to determine the way of augmenting models adaptively.

There is a trade-off between the detection range and estimation accuracy obtained by using different models for model set design. According to our simulation results, generally speaking, half failure models used in the model set have a greater detection range but inferior estimation accuracy. In general, for severe sequential and multiple failures, the IM^{3}L algorithm with total failure models obtains better performance than that with half failure models, as shown in the above simulations. This makes sense since total failure models are closer to the truth than half failure models in this case. However, it also should be noticed that the impact of different model sets on detection and estimation is system dependent.

Compared with sequential failures, multiple failures are more difficult to detect in that the performance depends on the difference of quantities between multiple failures as well as diagnosis of the first failure (in dual failure cases). This is particularly true for cases in which half failure models are in effect. If the first failure is severe and the second is very mild, there may be longer detection delay, miss detection or larger estimation error, since the first severe failure may mask the second mild failure. In short, good performance can be achieved if different failures have comparable severeness based on the designed model sets. Future work includes the FDI of incipient failures and reconfigurable control.

4.7 Summary

In this chapter we proposed two FDI schemes based on variable-structure multiple model estimation, where the model set is made adaptively online. This makes it possible to cover all possible faults with a relatively small number of models at a given accuracy level. In particular, combining IMM and MLE estimation techniques, the proposed IM^{3}L provides an integrated way to detect and estimate partial as well as multiple failures. An efficient IMM-FDI scheme has been presented, and implementation issues based on hierarchical structure have also been discussed. The simulation results over various scenarios demonstrate that the IM^{3}L approach is a powerful technique for FDI. It outperforms the other two hierarchical MM based approaches (HIMM and HAMM) as well as the conventional GLR approach in terms of correct detection, accurate estimation of the extent of failure, and robust state estimation in the presence of partial failures (unmodeled failures) and uncertainties in the noise statistics.

Chapter 5

Sequential Detection of Change Points

5.1 Introduction

Over past decades, there has been a significant increase of research in a variety of applications including fault detection and diagnosis, quality control, safety of complex systems (aircraft, rockets, chemical technological processes, etc.), segmentation of signals, biomedical signal processing, finance, and channel monitoring for mobile wireless communication systems. The common feature across the above problems is to detect an abrupt change in a system based on stochastic observations of the system. [10] provides a good overview of the existing methodology in this area. There have been various tests for different problems. They can be largely classified into two categories: offline and online. The offline tests mean that the data are collected first and then a decision is made based on the analysis of all data. The online test makes the decision based on data sequentially obtained. Our work focuses more on online detection.

To make the problem more precise, assume that there is only one change at each time

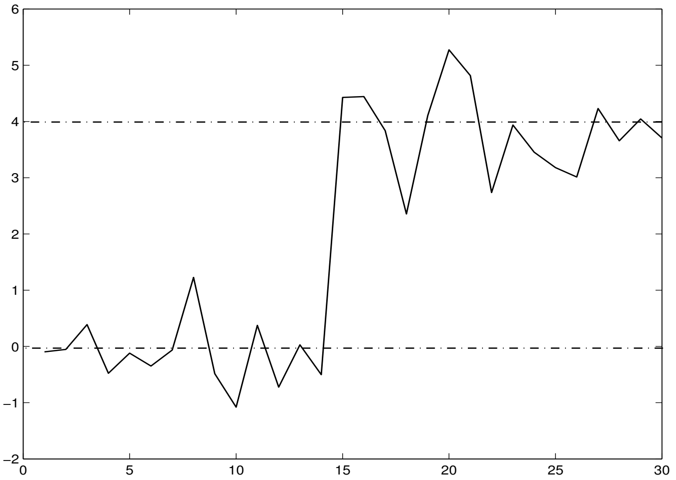


Figure 5.1: A change in the mean of a Gaussian process

instant. Let $\langle z_k \rangle$ be a measurement sequence. Before the unknown change time n, z_k ($k = 1, ..., n{-}1$) are independent and distributed identically with probability density function $f_0(z_k)$. After the change, z_k ($k = n, ..., N$) are independent and distributed identically with probability density function $f_1(z_k)$. For example, a change occurs in the mean of a Gaussian process in Fig. 5.1. This problem can be formulated as a binary hypothesis testing

$$\begin{array}{ll} \bar{H}_0(\text{no change}): & z_k \sim f_0(z_k) \quad \text{for} \quad k = 1, \ldots, N \\ \bar{H}_1(\text{there is a change}): & \begin{cases} z_k \sim f_0(z_k) \quad \text{for} \quad k = 1, \ldots, n-1 \\ z_k \sim f_1(z_k) \quad \text{for} \quad k = n, \ldots, N \end{cases} \end{array}$$

where n is unknown and referred to as the change point. This is known as *change point detection* in the statistical literature. The goal is to detect the occurrence of the change as soon as possible under certain constraints, for example, a fixed false alarm rate before n. The detection is performed by a *stopping rule*, which usually has the form

$$\hat{n} = \min\{k : g_k(z_1, ..., z_k) \geq \lambda\}$$

where λ is the preset threshold, $(g_k)_{k\geq 1}$ is a function of measurements, and $\hat{n}$ is the estimated time at which the change is detected.

The design of the quickest change detection procedure usually involves optimizing the tradeoff between two kinds of performance measures: detection delay and false alarm error rate (i.e., the probability of deciding $\bar{H}_1$ when $\bar{H}_0$ is true). They are a conflicting pair. It is difficult, if not impossible, to shorten the detection time and reduce the false alarm rate at the same time. The common practice is to maximize the detection probability (i.e., the probability of deciding $\bar{H}_1$ when $\bar{H}_1$ is true) subject to a fixed false alarm error rate.

There has been a long history of work on change point detection in statistics and engineering. According to the sample size, there are two approaches to detect changes. One is the fixed-size batch detection, which is widely used in quality control. Typical algorithms include Schewhart's chart, geometric moving average, and finite moving average chart [10]. Basically the test statistics are calculated based on fixed-size sample and compared to a threshold. The test will terminate once $\bar{H}_1$ is decided, otherwise, the sampling and test continue. The other is sequential detection, which monitors the system variables successively. Sequential change point detection is designed to detect a change as soon as possible after its occurrence. It is usually preferable to consider decisions in a sequential setting since measurements are received sequentially by sensors in most practical systems. Moreover, a sequential test does not need to determine the sample size in advance, unlike the fixed sample size tests. In has been shown that sequential tests outperform non-sequential tests based on the same decision error rates [44]. The use of sequential tests for binary hypotheses has been well studied in the literature. Next we briefly present several sequential test procedures.

5.2 Sequential Probability Ratio Test (SPRT)

Let H_0, H_1 denote two simple hypotheses: H_0 : a measurement sequence $\langle z_k \rangle$ with a probability density $f_0(z)$, and H_1 : a measurement sequence $\langle z_k \rangle$ with a probability density function $f_1(z)$. It is well known that for binary simple hypotheses testing, Wald's SPRT (SPRT) is optimal in the sense that it makes the quickest detection for both tests (H_0, H_1) given any fixed false alarm and missed detection probabilities. Let

$$L^k = \log \frac{f(z^k|H_1)}{f(z^k|H_0)} = \sum_{\kappa=1}^{k} \log \frac{f\left(z_\kappa|H_1, z^{\kappa-1}\right)}{f\left(z_\kappa|H_0, z^{\kappa-1}\right)} \tag{5.1}$$

denote the log-likelihood ratio of two hypotheses H_1 and H_0 based on the measurements up to k, where $f(z^k|H_i)$ is the likelihood of the hypotheses, $f\left(z_\kappa|H_i, z^{\kappa-1}\right)$ $(i = 0, 1)$ is the marginal likelihood. let A and B be two thresholds which are set by $A = \frac{\beta}{1-\alpha}$ and $B = \frac{1-\beta}{\alpha}$ for given false alarm rate α and miss detection probability β, where

$$P(\text{Choose } H_1|H_0) \leq \alpha,\ P(\text{Choose } H_0|H_1) \leq \beta,\ \ 0 < \alpha, \beta < 1 \tag{5.2}$$

Then the SPRT decision rule is

$$\begin{array}{ll} \text{Accept } H_1 & \text{if } L^k \geq \log B \\ \text{Accept } H_0 & \text{if } L^k \leq \log A \\ \text{Continue } (k \mapsto k+1) & \text{otherwise} \end{array} \tag{5.3}$$

However, the problem formulation of the standard SPRT does not fit well to the change point detection. In particular, the SPRT assumes all data relates to one of the two hypotheses so that it simply chooses that there is a change or that there is no change. Whenever a decision (H_0 or H_1) is made, the test will terminate. However, this is not the goal of change point detection. In fact, we mainly want to know "*when the change occurs*". Therefore, the test should continue to the next cycle with more measurements if H_0 is deemed true.

There are several formulations in the literature regarding change point detection. A non-Bayesian approach, first proposed by Page in 1954, Cumulative Sums (CUSUM), minimizes the detection delay given the decision error probabilities. It is one of the most popular algorithms used to detect a possible change from a given process to another given process. Shiryayev sequential probability ratio test (SSPRT), a Bayesian approach proposed by Shiryayev in the early 1960s, minimizes a risk function at each time step with the assumption that the change point has a geometric prior distribution. It has been proven to provide the quickest detection of a change in a sequence of conditionally independent measurements under the given decision error rates. One note to emphasize is that both CUSUM and SSPRT hold the optimality for *simple* hypotheses, that is, distributions are completely known before and after the change, which is not the case for many applications where change magnitude is usually unknown (e.g., H_1 is a *composite* hypothesis). In order to apply these sequential tests to such situations, they have to be modified, which is discussed in the next chapter.

5.3 Repeated SPRT-Based Detector: CUSUM

The standard CUSUM algorithm, also called Page's test, was proposed as a means to detect sequential changes in distributions of discrete-time random processes. It guarantees the quickest decision given the decision error rate for simple hypotheses. Page's test can be interpreted as a *repeated* SPRT with the lower threshold $logA$ equal to 0 and the upper threshold equal to λ decided by error probabilities. The key idea is to restart the SPRT algorithm as long as H_0 is being accepted, which makes it naturally fit to change point detection problems.

The CUSUM algorithm [10] can be written in a recursive manner:

$$L^k = \max \left\{ L^{k-1} + \log \frac{f\left(z_k|H_1, z^{k-1}\right)}{f\left(z_k|H_0, z^{k-1}\right)},\ 0 \right\}, \quad L^0 = 0 \tag{5.4}$$

and the decision rule is

1) Accept H_1 (declare a change) if $L^k \geq \lambda$. Then stopping time $\hat{n} = \min\{k : L^k \geq \lambda\}$ is the time that a change is detected.

2) Continue the test $(k \mapsto k+1)$ if $L^k < \lambda$.

Asymptotic optimality of the CUSUM procedure was proved by Lorden (1971) under a min–max criterion. It shows that the CUSUM asymptotically minimizes the "worst case" detection delay subject to a lower bound on the mean time between false alarms. Many modifications and extensions of the CUSUM-type test have been developed for the online change-point detection in stochastic control and signal processing, such as fault detection in complex systems and signal segmentation [35, 34, 76].

5.4 SSPRT-Based Detector

The Shiryayev sequential probability ratio test (SSPRT) focuses on the detection in a series of conditionally independent measurements by noting the change in the probability density function of the measurements. Some results have recently been reported for fault detection using SSPRT [63, 93].

The optimality of SSPRT is to minimize an expected cost at each time step. This cost includes the measurement cost and the cost due to a terminal decision error by false alarm or miss detection. Since a Bayesian framework is used, it is necessary to define prior information. This includes the a priori probabilities of H_i $(i = 0, 1)$ and the transition probabilities of H_0 to H_1 from $k-1$ to k (which can be time variant or invariant). When a binary hypothesis test is performed, the transition probabilities degenerate into a single number, usually assumed to be time-invariant for simplicity.

The decision rule of the SSPRT is obtained by defining the posterior probability ratio P_k. Let $p_k = P\{n \leq k|z^k\}$ denote the posterior probability that a change occurs (at unknown time n) by time k given the available measurements z^k, and

$$P_k \triangleq \frac{p_k}{1-p_k}, \quad P_T \triangleq \frac{p_T}{1-p_T} \tag{5.5}$$

where the choice of a preset threshold p_T is related to the desirable decision error rate. The SSPRT becomes

1) Accept H_1 (declare a change) if $P_k \geq P_T$. The stopping time is $\hat{n} = \min\{k : P_k \geq P_T\}$.

2) Continue the test $(k \mapsto k+1)$ if $P_k < P_T$.

Calculating p_k is the key to SSPRT. Fortunately, it can be done recursively. Let p_0^i denote the prior probability of hypothesis H_i being true, π the transition probability from H_0 to H_1, and $\phi_k^i \triangleq P(n \leq k+1|z^k)$ $(i = 0, 1)$. The recursive form of the posterior probability of H_i is given by

$$p_k^1 = \frac{\phi_{k-1}^1 f\left(z_k|H_1, z^{k-1}\right)}{\sum_{i=0}^1 \phi_{k-1}^i f\left(z_k|H_i, z^{k-1}\right)}, \; p_k^0 = 1 - p_k^1$$

$$\phi_{k-1}^1 = p_{k-1}^1 + \pi\left(1 - p_{k-1}^1\right), \; \phi_{k-1}^0 = 1 - \phi_{k-1}^1$$

Then, the test statistic of the SSPRT is

$$P_k \triangleq \frac{p_k^1}{p_k^0} = \frac{f(z_k|H_1, z^{k-1})}{f(z_k|H_0, z^{k-1})} \frac{P_{k-1} + \pi}{1 - \pi}, \; P_0 \triangleq \frac{p_0^1}{p_0^0}$$

Note that no reset mechanism is necessary for the SSPRT due to the nature of its problem formulation that determines when a disruption of H_1 is true. Further, if π is zero, the SSPRT becomes the SPRT. This makes sense in that SPRT assumes all data relate to one of the two hypotheses.

Note that the above SPRT-based procedures assume that measurements are independent. However, measurements are correlated in many practical problems, such as the target tracking problem. In this case the above test still works provided the sequence $\langle l_k \rangle$ of marginal likelihood ratios is independent, where

$$l_\kappa = \frac{f\left(\tilde{z}_\kappa|H_1, z^{\kappa-1}\right)}{f\left(\tilde{z}_\kappa|H_0, z^{\kappa-1}\right)}$$

Fortunately, this is approximately the case since the measurement residual sequence is approximately Gaussian distributed and weakly coupled under some conditions [44]. Thus measurement residuals, instead of measurements, should be used to compute likelihood ratios.

Chapter 6

Sequential Detection of Target Maneuvers

6.1 Introduction and Related Research

Maneuvering target tracking (MTT) is an important problem complicated by the fact that accelerations are generally unknown and that structural variations may also exist as the target moves into and out of the maneuvering mode. Neither accelerations nor possible structural changes are available directly through measurements in practice. In general, the MTT problem is a hybrid estimation problem since it involves discrete mode or parameter estimation as well as continuous state estimation. The decision-based techniques for MTT, which appeared after the decision free adaptive Kalman filter techniques based on a single model, have become quite popular and have been studied extensively in the literature [6, 8, 12, 50]. In decision-based approaches, the state estimation is based on a hard decision on the target motion model which is made by the maneuver detector. Therefore making reliable and timely decisions is key to these approaches for satisfactory state estimation. Many such algorithms and techniques have been developed to detect maneuvers [50].

Target maneuvers consist of maneuver onset and termination, which are observable as changes appearing in the measurements. The problem of detecting maneuvers thus can be classified as maneuver onset detection and termination detection. Onset detection algorithms can be categorized based on which test they utilize: the chi-square based test and the likelihood ratio based test. Techniques based on the chi-square test include measurement residual based and input estimate based detectors. Those based on likelihood ratio tests include the generalized likelihood ratio (GLR) based and marginalized likelihood ratio based (MLR) detectors. These algorithms are widely used in MTT applications but there are no comprehensive references available for performance comparison to our knowledge. Thus, this chapter will first focus on the onset detection performance comparison for six existing maneuvering detection algorithms in various scenarios:

- Measurement residual based chi-square detector (MR)
- Input estimate based chi-square detector (IE)
- Input estimate based Gaussian significance detector (IEG)
- Generalize likelihood ratio test detector (GLR)
- Marginalized likelihood ratio detector (MLR)
- Cumulative sum based detector (CUSUM)

Maneuver termination is in general much harder to detect than maneuver onset detection, but is also less important [50]. Up to date maneuver termination is rarely defined by any rigorous problem formulation and few methods have been presented in the literature.

Besides aforementioned traditional detection algorithms that are based on batch processing, another class of statistical tests that can be applied to maneuver detection is the sequential tests for change point detection, as pointed out in [50, 82]. Sequential detection procedures have been successfully applied to fault detection (e.g., [63, 93]) but not to target maneuver detection, to our knowledge, except for a quickest detector in [97]. Some min-max based solutions have been proposed in [75, 76, 77] with applications to navigation system integrity monitoring. It optimizes the worst case situation with given decision error rates.

In this chapter, we consider detecting a target's maneuver as a binary hypothesis testing problem (H_0: no maneuver; H_1: with maneuver). Once a target starts maneuvering, it should be detected as quickly as possible under certain constraints such as decision errors. Two target maneuver onset detectors based on CUSUM and SSPRT tests are developed by using a likelihood marginalization technique to cope with the difficulty that target maneuver accelerations are unknown. The proposed approach essentially utilizes a priori information about the maneuver accelerations in typical tracking engagements and thus allows improvement of the detection performance, especially for normal accelerations, as compared with two widely used maneuver detectors.

This chapter is organized as follows. In Section 2, the problem of maneuvering target detection is formulated as binary composite hypothesis testing. Traditional maneuver detectors are briefly discussed in Section 3 and are compared in Section 4 over various scenarios. Motivation of sequential detection of target maneuvers is presented in Section 5. Sequential procedures based on CUSUM and SSPRT are derived by means of likelihood marginalization using two typical prior models of maneuvers in Section 6. The performance of the proposed maneuver detectors is evaluated by simulations and compared with that of two widely used detectors in Section 7. A summary is provided in Section 8.

6.2 Problem Formulation

The target-measurement model is given by

$$x_{k+1} = F_k x_k + u_k + w_k \tag{6.1}$$

$$z_k = H_k x_k + v_k, \quad k = 1, 2, \ \ldots \tag{6.2}$$

where x_k is the target state, u_k is the maneuver control input, and z_k is the measurement. $w_k \sim \mathcal{N}(0, Q_k)$ and $v_k \sim \mathcal{N}(0, R_k)$ are independent process and measurement noises, respectively, and the initial state $x_0 \sim \mathcal{N}(0, P_0)$ is independent of w_k and v_k.

It is assumed that $u_k = 0$ when the target is not maneuvering at time k and $u_k \neq 0$ when the target is maneuvering. If the target begins a maneuver at an unknown time $n \leq k$ then

$$\langle u_k \rangle = \{\ldots, 0, \ldots, 0, u_n, u_{n+1}, \ldots, u_k\}$$

In general, it is not necessary for u to remain constant during the maneuver. The focus of the maneuver *onset detection* is to decide on a maneuver and estimate the onset time n, which can be formulated as a binary hypothesis testing problem:

$$\begin{aligned} H_0: & \quad u_m = 0 \quad \text{for} \quad m = 1, \ldots, k \\ H_1: & \quad \begin{cases} u_m = 0 & \text{for} \quad m = 1, \ldots, n-1 \\ u_m \neq 0 & \text{for} \quad m = n, \ldots, k \end{cases} \end{aligned}$$

where u and n are unknown parameters, and n is referred to as the change point. This is known as *change point detection* in the statistical literature. Since both u and n are usually unknown in practice, hypothesis H_1 is clearly composite. This makes the maneuver detection difficult since in general there is no existing optimal non-Bayesian solution for composite hypotheses testing problems.

There are two common approaches to solving this problem of unknown u. The first one is based on the generalized maximum likelihood ratio (GLR) principle, where the unknown u (treated as nonrandom) is replaced with the maximum likelihood estimate $\hat{u}$ (see [32, 50, 99]). The second technique is based on Bayesian framework where u is treated as random, and likelihood under H_1 is determined by an appropriate marginalization—averaging out all possible values of the unknown u as a nuisance parameter. The a priori probability distribution $f(u)$ is needed for this technique. In target tracking applications, prior distributions of some target maneuver motions are available [30], which are usually dependent on class or type of target. In other cases, a non-informative prior may be assumed. In this chapter, we will focus on the development of the sequential maneuver detection algorithms with the proper utilization of prior information.

6.3 Existing Algorithms for Maneuver Onset Detection

6.3.1 Measurement Residual Based Chi-Square Detector (MR)

The measurement residual based chi-square test is a very simple tool and has been used as a standard component in many MTT applications [4, 8, 12, 20, 50]. Under the linear-Gaussian assumption and H_0, measurement residuals of a Kalman filter are zero mean, Gaussian and white; i.e., $\tilde{z}_k \sim \mathcal{N}(0, S_k)$ where $\tilde{z}_k = z_k - \hat{z}_{k|k-1}$ and $S_k = \mathtt{cov}\,(\tilde{z}_k)$. As such, $\epsilon_k = \tilde{z}_k^T S_k^{-1} \tilde{z}_k$ is a $\chi^2_{n_z}$ distributed variable with $n_z = \dim(\tilde{z}_k)$. This property allows us to check if $\tilde{z}_k$ has the assumed distribution under H_0. This leads to a simple test for a maneuver:

$$\epsilon_k > \chi^2_{n_z}(\alpha) \tag{6.3}$$

where $1-\alpha$ is the confidence level of the test. This means H_0 will be rejected with confidence $1 - a$ if ϵ_k exceeds the corresponding threshold. In practice, the moving sum ϵ_k^s for a sliding window of length s is often used, as is the exponentially decaying average ϵ_k^ρ,

$$\epsilon_k^s = \sum_{j=k-s+1}^{k} \epsilon_k \tag{6.4}$$

$$\epsilon_k^\rho = \sum_{j=1}^{k} \rho^{k-j}\epsilon_j = \rho\epsilon_{k-1}^\rho + \epsilon_k \qquad 0 < \rho < 1 \tag{6.5}$$

where ϵ_k^s is chi-square distributed as $\epsilon_k^s \sim \chi^2_{sn_z}$, and ϵ_k^ρ is only approximately distributed as $\frac{1}{1+\rho}\chi^2_{n_\rho}$ with $n_\rho = n_z\frac{1+\rho}{1-\rho}$. Its effective window length is $\frac{1}{1-\rho}$. For example, $\rho = 0.8$ corresponds to $s = 5$.

6.3.2 Input Estimate Based Chi-Square Detector (IE)

The basic idea of the IE detector is to explicitly estimate the unknown input $\hat{u}_k$ and then make a decision based on chi-square test of $\hat{u}_k$. Due to its conceptual simplicity, an IE detector is used in many algorithms in target tracking [8, 17, 39, 50, 94]. Theoretically under H_0 and the linear-Gaussian assumption, u_k is zero and thus the linear unbiased estimate $\hat{u}_k$ is zero mean and Gaussian $\hat{u}_k \sim \mathcal{N}(0, \Sigma_k)$ with $\Sigma_k = \texttt{cov}(\hat{u}_k)$. Similarly to the measurement residual detector, $\epsilon_k^u = \hat{u}_k' \Sigma_k^{-1} \hat{u}_k$ is $\chi^2_{n_u}$ distributed with $n_u = \dim(\hat{u}_k)$, which provides a justification for the chi-square test for maneuver onset detection. Clearly, the key to this approach is obtaining accurate input estimates.

Under the linear-Gaussian assumption, u can be explicitly estimated using a least-squares framework [8]. Assume u is constant in the interval $[k-s, ..., k-1]$, i.e., $u_i = u$, $i = k-s, ...k-1,$. It can be shown that $\hat{u}_k$ follows from the linear model

$$\tilde{z} = \Psi u + v \tag{6.6}$$

where $\tilde{z} = [\tilde{z}_{k-s+1}, ..., \tilde{z}_k]', \Psi = [\Psi_{k-s+1}, ..., \Psi_k]', \Psi_i = H \sum_{j=k-s}^{i} [\prod_{m=0}^{i-j-1} \Phi_{i-m}]G$, $\Phi_i = F[I - K_i H]$. Then the input estimate is given using least-squares batch estimation:

$$\hat{u} = (\Psi' S^{-1} \Psi)^{-1} \Psi' S^{-1} \tilde{z} \tag{6.7}$$

$$\Sigma = \texttt{cov}(\hat{u}) = (\Psi' S^{-1} \Psi)^{-1} \tag{6.8}$$

$$S = \texttt{diag}(S_{i+1}), \quad S_{i+1} = \texttt{cov}(\tilde{z}_{i+1}) \tag{6.9}$$

The input estimate detector decides H_1 if the test statistics $\epsilon_k^u > \lambda$, where λ is decided by the confidence level.

If a maneuver is detected, the state estimates are corrected. The estimated input contributes to correct the predicted state as well as its associated covariance.

6.3.3 Input Estimate Based Gaussian Significance Detector (IEG)

This detector is also based on an input estimate, but uses a different test statistic [11, 13, 40]. A maneuver will be declared if a component $\hat{u}_k^i$ of $\hat{u}_k$ is statistically significant; that is,

$$\max_i \left(\frac{|\hat{u}_k^i|}{\Sigma_i^{1/2}} \right) > \lambda,\ i = 1, ..., n_u \tag{6.10}$$

where $\Sigma_i^{1/2} = \texttt{var}(\hat{u}_k^i)$, $n_u = \dim(\hat{u}_k)$, and λ is determined from the standard Gaussian distribution. The absolute value of $\hat{u}_k^i$ should be used since we are testing the significance of $\hat{u}_k^i$ rather than its direction.

The above detectors are used to detect H_0 only, regardless of the maneuver model. For example, input estimate based techniques directly estimate the unknown target maneuver u from available measurements. The next three detectors are based on the likelihood ratio test, which is a standard method of hypothesis testing. Let

$$L_k\left(u, n\right) = \log \frac{f(z_s^k|H_1(u,n))}{f(z_s^k|H_0)} \tag{6.11}$$

denote the log-likelihood ratio of two hypotheses, H_0 and H_1, based on a set of measurements $z_s^k \triangleq \{z_{k-s+1}, ..., z_k\}$, where $f(z_s^k|H_i)$ $(i = 0, 1)$ is the likelihood of the hypothesis.

Since maneuvers are not known, which makes the likelihood ratio in (6.11) unknown, maneuver onset detection becomes a composite hypothesis testing problem in this case. In the literature, two types of detectors are available, differing in the assumption of unknown u.

6.3.4 Generalized Likelihood Ratio (GLR) Detector

The generalized likelihood ratio (GLR) detector is quite popular and has many forms. The basic idea is to replace the unknown likelihood with the most probable likelihood. A full implementation of GLR, requiring a linearly increasing number of parallel filters, is infeasible for real MTT applications. Instead, a GLR algorithm based upon a finite window is widely used [32, 90, 50, 99, 100]. Assuming that u is a constant in the interval $[k-s,...,k-1]$, $(\hat{u},\hat{n})$ is used in test statistics given by

$$L_k(\hat{u},\hat{n}) = \log \frac{f\left(\tilde{z}_s^k|H_1\left(\hat{u},\hat{n}\right)\right)}{f\left(\tilde{z}_s^k|H_0\right)} = \sum_{i=k-s+1}^{k} \log \frac{f(\tilde{z}_i|H_1(\hat{u},\hat{n}))}{f(\tilde{z}_i|H_0)} \tag{6.12}$$

where the joint maximum $(\hat{u},\hat{n}) = \arg\max_{u,n} f\left(\tilde{z}_s^k|H_1\left(u,n\right)\right)$ can be found in two steps iteratively.

In the first step, $\hat{u}\left(n\right)$ is estimated. Under the linear-Gaussian assumption, the LS estimate $\hat{u}$ is equivalent to the MLE, and thus $\hat{u}\left(n\right) = \hat{u}^{IE}$. In the second step, $\hat{n}$ is computed as the maneuver starting time that maximizes the likelihood ratio given $\hat{u}\left(n\right)$,

$$\hat{n} = \arg\max_n \sum_i^k \log \frac{f\left(\tilde{z}_i|H_1\left(\hat{u}\left(n\right),n\right)\right)}{f\left(\tilde{z}_i|H_0\right)}, \; i = k-s+1,...,k-1. \tag{6.13}$$

The GLR detector declares a maneuver if the test statistic $L_k\left(\hat{u},\hat{n}\right)$ exceeds a preset threshold λ, where λ is determined by the desired decision error rates.

6.3.5 Marginalized Likelihood Ratio (MLR) Detector

The marginalized likelihood ratio test is a Bayesian approach which assumes u is a random variable with a certain prior density. The MLR detector replaces the unknown likelihood with the average marginalized likelihood by averaging over all possible values of u :

$$L_k(n) = \log \frac{f\left(\tilde{z}_i^k | H_1\left(n\right)\right)}{f\left(\tilde{z}_i^k | H_0\right)} = \log \frac{E\left[f\left(\tilde{z}_i^k | H_1\left(u, n\right)\right)\right]}{f\left(\tilde{z}_i^k | H_0\right)} = \log \frac{\int f\left(\tilde{z}_i^k | H_1\left(u, n\right)\right) f\left(u\right) du}{f\left(\tilde{z}_i^k | H_0\right)} \tag{6.14}$$

The MLR detector declares a maneuver if $L_k(n) > 0$. This test aims to obtain the marginal ML estimate n that has the maximum likelihood for an average u, i.e., $\hat{n} = \arg\max_n L_k(n)$.

A major difficulty for the MLR detector is how to calculate the integral in (6.14). An approximate MLR detector was recently proposed in [27], where a Kalman filter is used in both the forward filtering and the backward smoothing steps required in likelihood estimation for computing test statistics. A non-informative prior of the jump magnitude was used in this approximate MLR detector. The paper [27] further illustrates how to correctly marginalize noise covariances of the model being used, where it is considered a nuisance parameter.

6.3.6 CUSUM Based Detector (CUSUM)

For simple hypotheses, the CUSUM algorithm has been proven to minimize the worst mean delay for detection when the mean time between false alarms goes to infinity [10]. However, since maneuver u is unknown in our problem, then $\hat{u}$ can be used in the likelihood computation ($\hat{u}$ can be obtained using the standard input estimation approach). The CUSUM detector decides H_1 if $L_k - \min_{k-s+1 \leq j \leq k} L_j \geq \lambda$, where λ is determined by the desired decision error rates,

$$L_k = \sum_{i=k-s+1}^{k} \log \frac{f(\tilde{z}_i | H_1(\hat{u}, n))}{f(\tilde{z}_i | H_0)} \tag{6.15}$$

$$\hat{n} = \min\{k : L_k - \min_{k-s+1 \leq j \leq k} L_j \geq \lambda\} \tag{6.16}$$

Since $\hat{u}$ is used, it is no longer possible to justify the optimality of the CUSUM detector. Other possible test statistics for the CUSUM test were discussed in the literature, including a CUSUM maneuver detector in [28].

6.4 Comparison of Existing Detection Algorithms

Four scenarios were designed to highlight different aspects of the detection algorithms. The first scenario (SM1) represents a relatively simple detection problem with a large sudden maneuver. The second (SM2) introduces a much smaller acceleration making it more difficult to detect. The third (CT) uses a mismatched maneuver model to evaluate the robustness of the detection algorithms in the presence of structural mismatch. The fourth (RM) is a random scenario, which allows an overall comparison of the algorithms under many different maneuvering scenarios.

The MR, IEG, GLR and CUSUM represent the detectors discussed in (6.5), (6.10), (6.12) and (6.15), respectively; the IE represents the detector defined in Sec. 6.3.2; and the MLR represents an approximate two-filter MLR detector developed in [27]. Unless otherwise stated, all simulation results are based on 100 Monte Carlo runs using false alarm rate $P_{fa} = 1\%$ or 5%.

6.4.1 Target Motion Model

The target motion and its observation are described by (6.1)–(6.2) with $F_k = F_{CV} = \text{diag}[F_2, F_2]$ or $F_k{=}F_{CT}$, and $G_k = G_{CV} = \text{diag}[G_2, G_2]$, where w_k and v_k are independent and

$$E[w_k] = 0,\ E[v_k] = 0,\ cov(w_k) = (0.008)^2 I,\ cov(v_k) = (35)^2 I$$

$$F_2 = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \quad G_2 = \begin{bmatrix} T^2/2 \\ T \end{bmatrix} \quad H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$F_{CT} = \begin{bmatrix} 1 & \frac{\sin \omega_k T}{\omega_k} & 0 & \frac{1-\cos \omega_k T}{\omega_k} \\ 0 & \cos \omega_k T & 0 & -\sin \omega_k T \\ 0 & \frac{1-\cos \omega_k T}{\omega_k} & 1 & \frac{\sin \omega_k T}{\omega_k} \\ 0 & \sin \omega_k T & 0 & \cos \omega_k T \end{bmatrix}$$

Initially the target moves at a nearly constant velocity with initial state $x_0 = [x, \dot{x}, y, \dot{y}]' = [0, 0, 0, 120m/s]'$, where (x, y) are the Cartesian coordinates. Two-point differencing is used to initialize the Kalman filter.

6.4.2 Four Scenarios

1. Simple Detection Scenario (SM1)

The target makes a maneuver during the period $k = [20, 28]$ with a magnitude of $u_k = [25, 4]'$.

2. Difficult Detection Scenario (SM2)

The target makes a maneuver during $k = [20, 25]$ with $u_k = [1, 1]'$ and then increases to $u_k = [2, 1.5]'$ during $k = [26, 35]$.

3. Constant–Turn Scenario (CT)

In practice it is also possible that an actual change takes place in the structure of the motion model. This situation is explored by changing the state transition matrix from *CV* to CT for $k = [20, 28]$ with a turn rate of $\omega_k = 3^o/\sec$.

4. Random Maneuver Scenario (RM)

In order to test the detectors on multiple maneuver scenarios simultaneously, a random scenario was developed, where $\langle u_k \rangle$ is assumed to be a semi-Markov random sequence described in polar coordinates as $u_k = a_k \angle \theta_k$ with magnitude a_k and angle θ_k. The scenario always starts from a nonmaneuvering state ($u_0 = 0$). The discrete sojourn time d_n that $\langle u_k \rangle$ stays in state $u_n = a_n \angle \theta_n$ until it jumps to another state $u_{n+1} = a_{n+1} \angle \theta_{n+1}$ ($u_n \neq u_{n+1}$) has a binomial distribution with $N = 50$ and $p = 0.5$:

$$p\{d_n = x\} = \binom{N}{x} p^x (1-p)^{N-x}, \quad x = 0, 1, 2, ..., 50$$

The transition from u_n to u_{n+1} is governed by

$$\begin{aligned} f(a_{n+1}|a_n) &= 0.6\delta(a_{n+1}) + 0.1\delta(a_{n+1} - 40) + \frac{0.3}{40} U(a_{n+1}; (0, 40)) \\ f(\theta_{n+1}|a_n = 0) &= \frac{1}{2\pi} U(\theta_{n+1}; (-\pi, \pi]) \\ f(\theta_{n+1}|a_n \neq 0) &= \mathcal{N}\left(\theta_{n+1}; \theta_n, \left(\frac{\pi}{12}\right)^2\right) \end{aligned}$$

where δ is the delta function, U is the uniform distribution.

Performance Indices

The performance of the detectors is compared in terms of average onset detection delay ($\hat{n} - n$), receiver operating characteristics (ROC) curves, and computational load. Since maneuver onset time in the random scenarios is not predefined and thus is different for each realization, it is meaningless when computing ROC curves to compare the detection probability at a specific time point as other deterministic scenarios did.

6.4.3 Simulation Results

Average Delay of Maneuver Onset Detection

Fig. 6.1 shows the average onset detection delay $(\hat{n}-n)$ under four scenarios for $P_{fa} = 5\%$. It was found that all detectors performed considerably worse on the SM2 scenario than on any other scenarios due to the really small accelerations applied. It is immediately noticeable that MR and MLR perform worse than all the other detectors in almost all scenarios regardless of window size. It appears difficult to generalize the relation between detection delay and window size from the plots. There is a small trend that a larger window size detects a maneuver more slowly. Intuitively, a larger window requires more data to properly compute the test statistic. Therefore it shows that larger windows do not necessarily confer better performance in terms of onset detection delay.

In our specific scenarios, the IE detector had the smallest detection delay when $s = 4$ in SM1 and CT scenarios. Likewise for GLR in scenarios SM2 and RM. In all scenarios considered here, the IE and GLR detectors ultimately performed better than the other techniques. Also, it was noted that detectors using input estimation with a window length of 4 to 6 generally gave a smaller detection delay.

Scenarios with $P_{fa} = 1\%$ were simulated and results are presented in Fig. 6.2. Longer detection delays were observed for all the detectors under all the scenarios since a smaller false alarm rate causes a larger detection threshold, and thus a lower probability of detection.

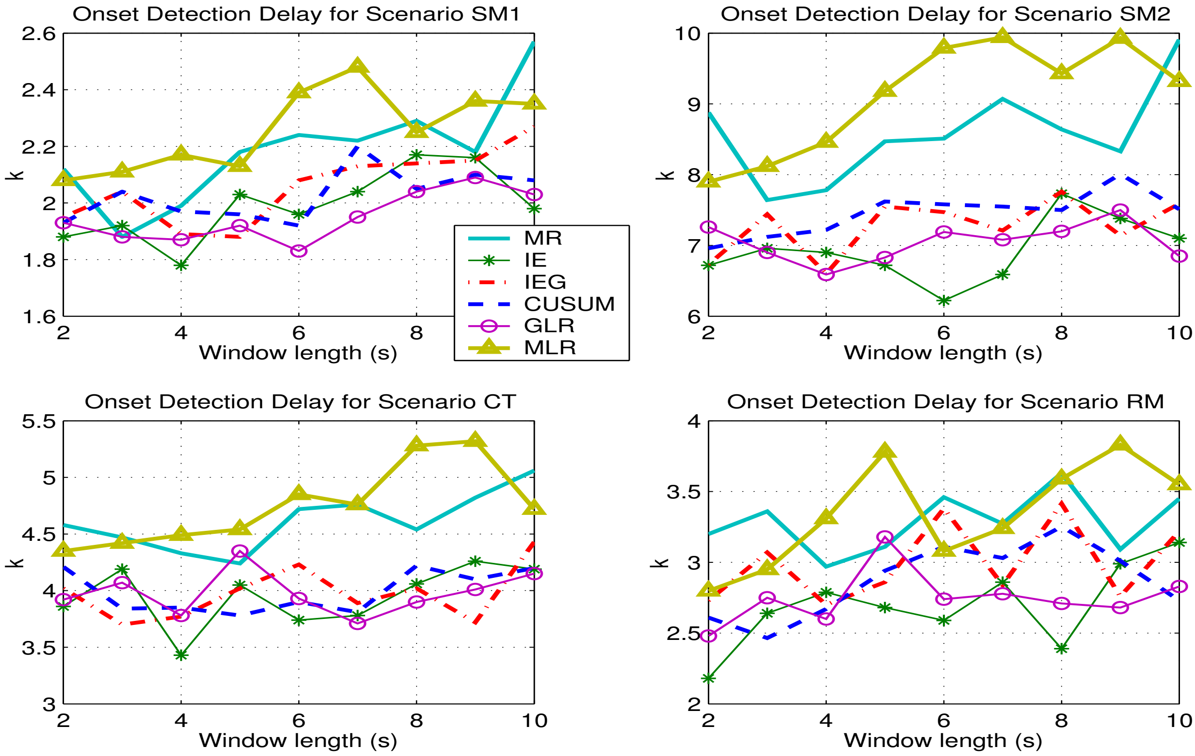


Figure 6.1: Onset detection delay for Pfa=5%

ROC Curves

The ROC curves for SM1 at $k = 20, ..., 23$ with $P_{fa} = 5\%$ are overlaid in Fig. 6.3. The ROC curves were generated by exhaustively computing the P_d at the time of interest for different P_{fa}'s using 250 Monte Carlo runs, each running up to time $k = 23$. The window size was chosen to be 5 for the detectors.

It is interesting to note that the MR detector performed comparably with the other four detectors using input estimates at maneuver onset time ($k = 20$) and became worse as time progressed. This verifies the contribution of input estimate on the detection performance of the IE based and GLR detectors: the accuracy of the input estimate improves as more maneuver data become available. At $k = 21$, GLR outperformed other detectors slightly for

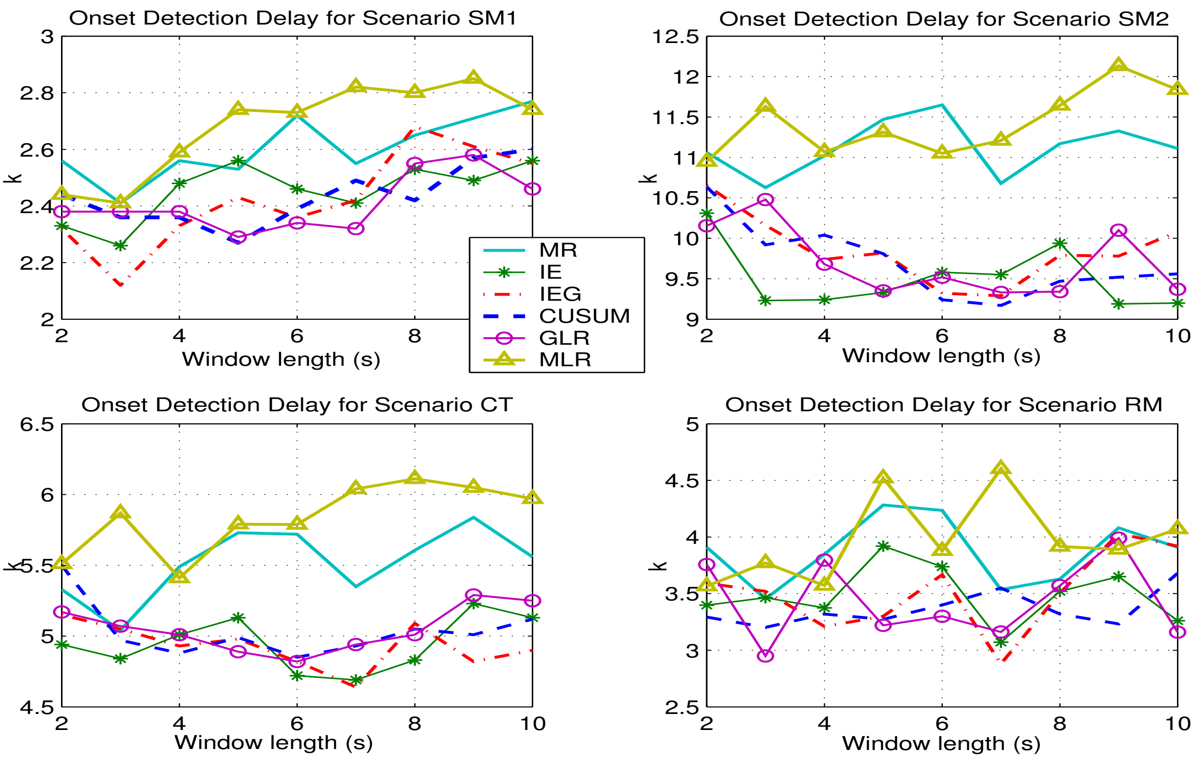


Figure 6.2: Onset detection delay for Pfa=1%

small P_{fa} ($P_{fa} < 0.25$); however, at this early time, no single detector was clearly superior. At $k = 22$, the CUSUM and GLR had similar performance and outperformed the other detectors.

At $k = 23$ each detector had a detection probability very close to 100%. This result is consistent with the average onset detection delay results (less than 3 sec for all detectors shown in Fig. 6.1). We noticed that MLR generally performed worse than other detectors for all samples under consideration.

The ROC curves for each maneuver onset detector at $k = 22, 24$ for SM2 with $P_{fa} = 5\%$ were overlaid in Fig. 6.4. It shows that for hard-to-detect maneuvers, MLR and MR detectors performed more poorly than the other IE-based detectors.

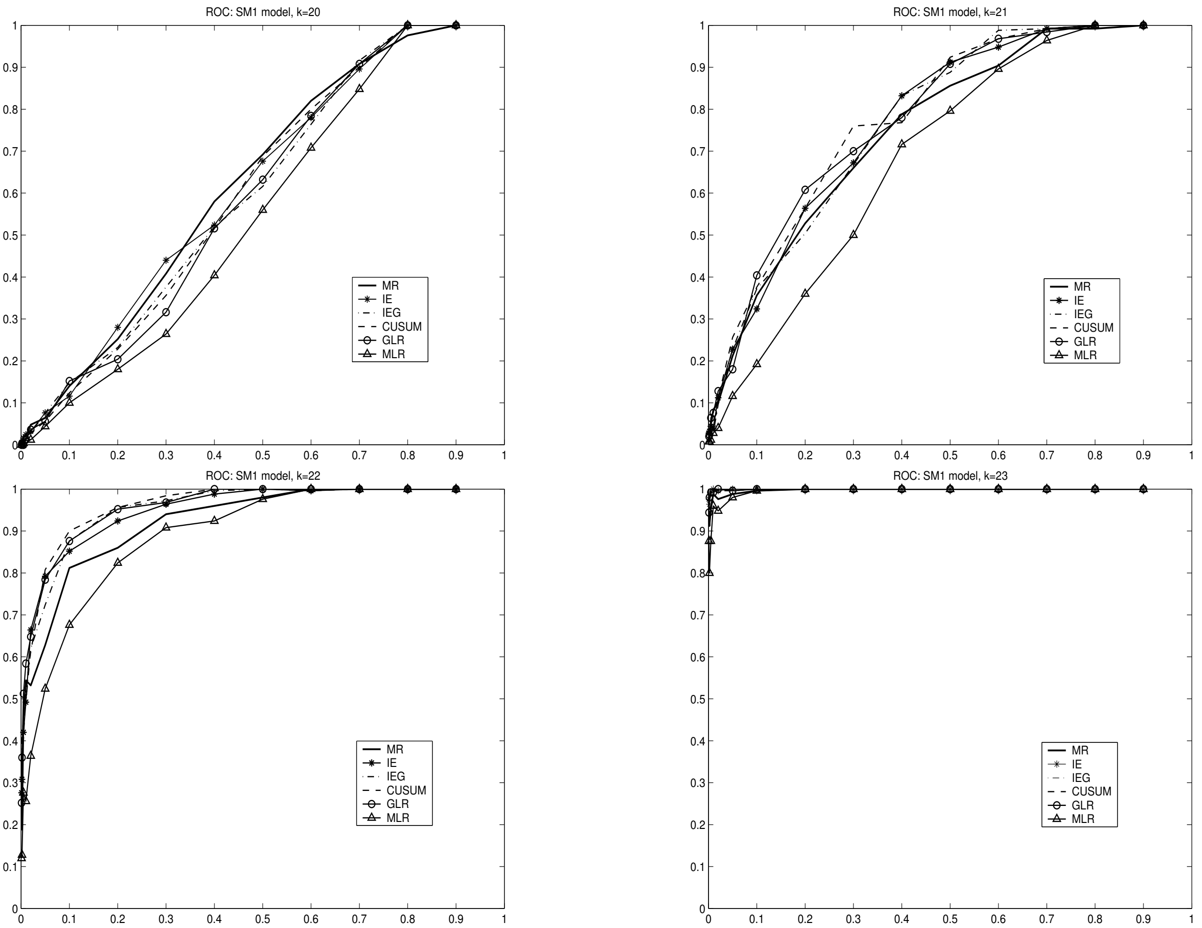


Figure 6.3: ROC curves for different detectors in SM1 (simple case)

Computational Complexity

The comparison of the computational complexity is shown in Fig. 6.5. Since scenario specifics and false alarm rate are irrelevant to the computational time required for detections, only one plot was shown and the results could be applied to all scenarios and false alarm rates.

As expected, the MR detector is least computationally intensive due to its simplicity. The MLR detector has the heaviest computational load due to the forward-backward filtering required at each time point [27]. The IE and IEG detectors required approximately the same processing time, as did the GLR and CUSUM detectors. This is because the detectors in

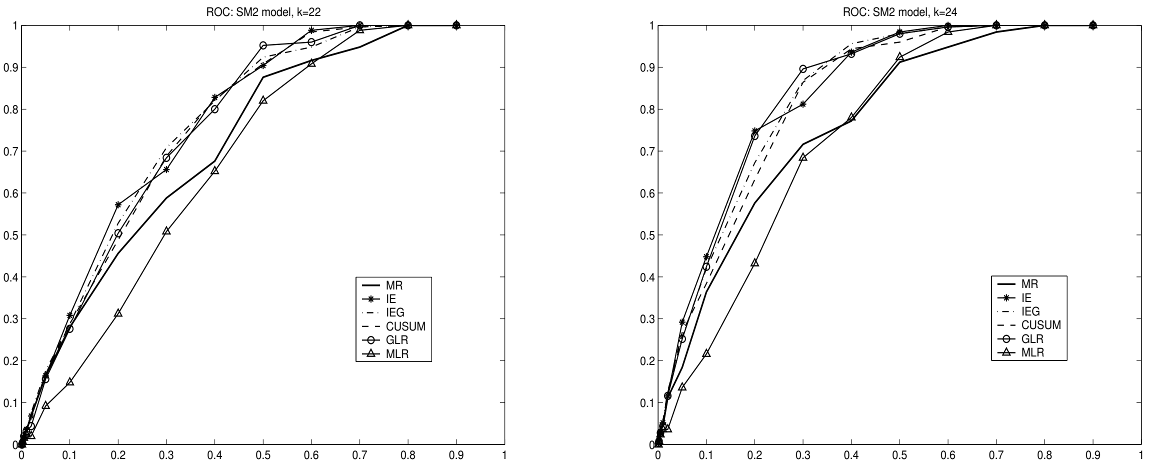


Figure 6.4: ROC curves for different detectors in SM2 (hard case)

each pair have similar decision functions, while the majority of the processing required goes into estimating the input. The processing time of all the detectors except MR increases as the window length s increases.

6.4.4 Discussions of Existing Maneuver Detection Algorithms

In general, the IE, IEG, GLR and CUSUM detectors outperform the MR and MLR detectors. IE, IEG, GLR, CUSUM have comparable performance, which is, however, scenario dependent.

As pointed out in [50], even though IE and GLR focus on estimation and detection respectively, the input-estimation component is essentially the same since ML and LS estimators are the same under the linear-Gaussian assumption. For this reason, they perform about the same, since only decision functions are different.

The MLR detector performed poorly, which might be due to the fact that the scenarios used did not exploit the strength of the MLR detector. With a better prior knowledge or unknown noise covariance, the marginalization technique for the MLR detector may improve

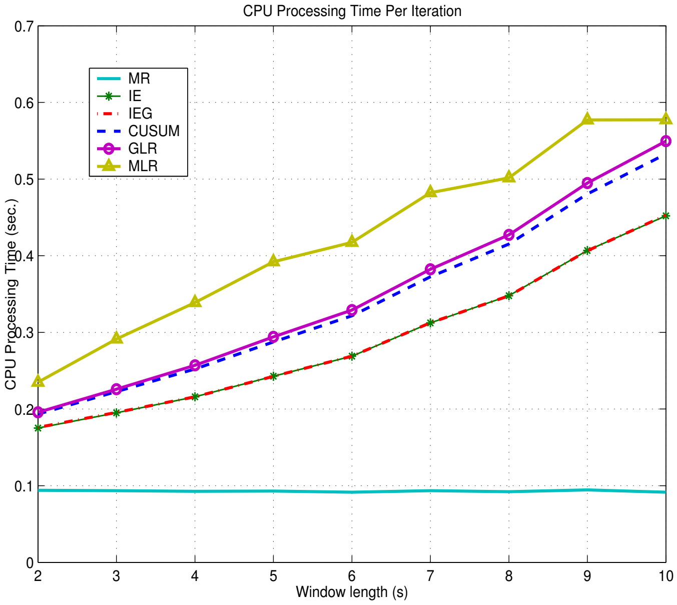


Figure 6.5: Comparison of CPU time

the performance. Furthermore, there is no need to be limited to the automatic threshold produced by the MLR, since the threshold was specified in the detection problem based on the desired false alarm rates.

It can be seen that onset detection delay is not strongly related to window length for all the detectors under consideration. As we have previously noted, if the window is too small, less data will cause a poor input estimate, resulting in a poor decision. Conversely, if the window length is longer, $\hat{u}$ is more accurate, resulting in a better decision. However, this increases the detection delay until enough data points have filled the sliding window. The best choice of the window length is no doubt scenario dependent.

Besides deterministic scenarios, a random scenario is important in order to verify that a particular scenario under consideration does not unfairly highlight a specific artifact of the detection algorithm that may or may not actually be important in real-life detection performance.

6.5 Motivation of Sequential Detection of Target Maneuvers

Based on our study, we feel that the existing detectors described above based on simplistic models are non-optimal and ad hoc. All detectors require some batch (sliding window) processing and are computationally demanding, especially for input estimate and likelihood ratio based tests. In addition, no detectors utilize any prior information of target motions, which is available in some cases. These limitations motivate us to investigate sequential processing of measurements, a technique which has been widely studied in change point detection. A sequential test involves a stopping rule and a final decision to achieve a trade-off between sample size and decision accuracy. For maneuver detection, sequential testing procedures are actually preferable because measurements are available sequentially. Moreover, a sequential test does not need to determine the sample size in advance, unlike nonsequential-based tests.

We consider detecting a target's maneuver as a binary hypothesis testing problem. Once a target starts maneuvering, it should be detected as quickly as possible under certain constraints such as the rates of false alarms and missed detections. The Wald's SPRT is well known for the problem of binary simple hypothesis testing. It makes the decision by comparing the likelihood ratio between hypotheses (H_0, H_1) with two thresholds. The SPRT is optimal in the sense that it makes the quickest detection under both hypotheses (H_0, H_1) given any decision error rates. However, the standard SPRT does not fit the sequential change point detection since it assume all data relates to one of the two hypotheses, i.e., it simply decides that there is a maneuver or that there is no maneuver. The CUSUM and SSPRT are very popular tests for sequential change point detection based on different assumptions. Unfortunately neither test can be blindly used for the detection of onset maneuvers since they are optimal only for simple hypothesis, which is not the case for maneuver detection where maneuver magnitudes are usually unknown. In the following sections, we will modify and evaluate them for the problem of maneuver onset detection.

6.6 Sequential Maneuver Detection Algorithm Development

6.6.1 Test Statistics of Two Sequential Detection Algorithms

In this section, test statistics are developed for the CUSUM and SSPRT detectors based on the marginalization technique and appropriate prior PDFs of the maneuvers of interest.

Denote by $\tilde{z}_k$ and S_k the measurement residual and its covariance, provided at time k by a nonmaneuvering Kalman filter for the system (6.1)-(6.2). As shown in Secs. 5.3 and 5.4, the test statistics of the CUSUM detector is

$$L^k = \max\left\{L^{k-1} + \log\frac{f\left(\tilde{z}_k|H_1, z^{\kappa-1}\right)}{f\left(\tilde{z}_k|H_0, z^{\kappa-1}\right)},\ 0\right\}, \quad L^0 = 0$$

and the test statistics of the SSPRT detector is

$$P_k \triangleq \frac{p_k^1}{p_k^0} = \frac{f(\tilde{z}_k|H_1, z^{k-1})}{f(\tilde{z}_k|H_0, z^{k-1})}\frac{P_{k-1}+\pi}{1-\pi},\ P_0 \triangleq \frac{p_0^1}{p_0^0}$$

Clearly, the key to computing test statistics of these detectors is to obtain marginal likelihood functions $f\left(\tilde{z}_k|H_i, z^{k-1}\right)$ under each hypothesis ($i = 0, 1$).

The marginal likelihood of H_0 is then determined as (under the linear Gaussian assumptions)

$$f\left(\tilde{z}_k|H_0, z^{k-1}\right) = \mathcal{N}(\tilde{z}_k; 0, S_k) = \frac{1}{\sqrt{|2\pi S_k|}}e^{-\frac{1}{2}\tilde{z}_k' S_k^{-1}\tilde{z}_k} \tag{6.17}$$

and the marginal likelihood of H_1 can be shown to be

$$f\left(\tilde{z}_k|H_1, z^{k-1}\right) = \mathcal{N}(\tilde{z}_k; HGu, S_k) = \frac{1}{\sqrt{|2\pi S_k|}} e^{-\frac{1}{2}(\tilde{z}_k - HGu)' S_k^{-1}(\tilde{z}_k - HGu)} \tag{6.18}$$

where the unknown u makes $f\left(\tilde{z}_k|H_1, z^{k-1}\right)$ unknown.

As mentioned in Sec. 6.3.3, there are two common approaches to solving this problem. The window-based GLR procedure is still limited with the difficulty of choosing the window size and the threshold. In target tracking applications, prior distributions of some target maneuver motions are available. Thus, a Bayesian-based approach can be applied, where u is treated as random and $f\left(\tilde{z}_k|H_1, z^{k-1}\right)$ is determined by an appropriate marginalization—averaging out all possible values of the unknown u as a nuisance parameter [28]. Specifically,

$$\begin{aligned} f\left(\tilde{z}_k|H_1, z^{k-1}\right) &= E\left[f\left(\tilde{z}_k|H_1(u), z^{k-1}\right)\right] \\ &= \int f\left(\tilde{z}_k|H_1\left(u\right), z^{k-1}\right) f\left(u\right) du \end{aligned} \tag{6.19}$$

where $f\left(u\right)$ is the *a priori* probability density function (PDF) of u.

Clearly, this approach provides a good means of utilizing the *a priori* information if it is available. A major difficulty for its application is how to calculate the integral in (6.19). If prior $f\left(u\right)$ is Gaussian or uniformly distributed, an exact formula can be obtained to calculate $f\left(\tilde{z}_k|H_1, z^{k-1}\right)$.

Lemma 1. If $f\left(u\right)$ is a Gaussian distribution $f(u) = \mathcal{N}(u; \bar{u}, \Lambda)$, marginal likelihood $f(\tilde{z}_k|H_1, z^{k-1})$ is determined to be (see Appendix D)

$$\begin{aligned} f(\tilde{z}_k|H_1, z^{k-1}) &= \int \mathcal{N}(\tilde{z}_k; HGu,\ S_k)\mathcal{N}(u; \bar{u}, \Lambda) du \\ &= \mathcal{N}(\tilde{z}_k; HG\bar{u},\ S_k + HG\Lambda(HG)') \end{aligned} \tag{6.20}$$

Lemma 2. If $f\,(u)$ is a uniform distribution $f(u) = \mathcal{U}(u; -u_{\max}, u_{\max})$, marginal likelihood $f(\tilde{z}_k|H_1, z^{k-1})$ is determined as (see Appendix E)

$$\begin{aligned} f\left(\tilde{z}_k|H_1, z^{k-1}\right) &= \int \mathcal{N}(\tilde{z}_k; HGu,\ S_k)\, \mathcal{U}(u; -u_{\max}, u_{\max}) du \\ &= \frac{1}{4u_{\max}\,|HG|}[erf(\frac{1}{\sqrt{2S}}(HGu_{\max} - \tilde{z}_k)) + erf(\frac{1}{\sqrt{2S}}(HGu_{\max} + \tilde{z}_k))] \end{aligned} \tag{6.21}$$

If $f\,(u)$ is not Gaussian or uniformly distributed, it is rarely possible to analytically calculate the integral in (6.19). To overcome this difficulty, we propose a fairly general way to approximately evaluate this expectation integral based on a Gaussian sum approximation of the prior PDFs. That is, we approximate

$$f(u) \approx \sum_{j=1}^{N} \lambda_j \mathcal{N}(u; \bar{u}_j, \Lambda_j) \tag{6.22}$$

where N is the number of Gaussian components, λ_j are their weights ($0 < \lambda_j < 1,\ \sum_{i=1}^{N} \lambda_j = 1$) and $\bar{u}_j$, Λ_j are their means and variances, respectively. The determination of these parameters is a part of the design and does not require online approximation. This approximation can be done by using standard Matlab functions for nonlinear multidimensional optimization to obtain a locally best fit. Other expectation maximization (EM) based mixture estimation techniques can also be used. Higher accuracy can be obtained with more components of the sum.

In general, under some regularity conditions, a non-Gaussian density function can be approximated to any desired accuracy by a weighted sum of Gaussians provided the number of components is sufficiently large [92]. This makes the proposed approach generally applicable when a marginalization is hard to obtain in exact analytical form.

Lemma 3. If $f_1(u)$ is a Gaussian-sum distribution $f(u) = \sum_{j=1}^{N} \lambda_j \mathcal{N}(u; \bar{u}_j, \Lambda_j)$, marginal likelihood $f(\tilde{z}_k|H_1, z^{k-1})$ is determined as

$$f\left(\tilde{z}_k|H_1, z^{k-1}\right) = \sum_{i=1}^{N} \lambda_i \mathcal{I}_i(\tilde{z}_k) \tag{6.23}$$

where $\mathcal{I}_i(\tilde{z}_k)$ is explicitly determined by (6.20):

$$\mathcal{I}_i(\tilde{z}_k) = \int \mathcal{N}(\tilde{z}_k; HGu,\ S_k)\mathcal{N}(u; \bar{u}_i, \Lambda_i)du = \mathcal{N}(\tilde{z}_k; HG\bar{u}_i,\ S_k + HG\Lambda_i(HG)') \tag{6.24}$$

Based on (6.17) and (6.18), test statistics of CUSUM and SSPRT detectors can be obtained.

6.6.2 Test Statistics of Sequential Detection for a Typical 2D Target

Next test statistics of maneuver detection algorithms are developed specifically for a typical 2D target with curvilinear motions. This curvilinear motion model accounts for possibly nonzero normal (cross-track) and tangential (along-track) target maneuver accelerations simultaneously.

Maneuver Model

A 2D target maneuver motion can be described as

$$x_{k+1} = Fx_k + \Gamma(x_k)a_k + w_k \tag{6.25}$$

where the state in the Cartesian coordinates is $x = [\mathsf{x}, \dot{\mathsf{x}}, \mathsf{y}, \dot{\mathsf{y}}]'$, and the acceleration is $a = [a_t, a_n]'$ with decomposed tangential and normal components a_t and a_n, respectively. The system matrices are

$$F = \operatorname{diag}\{F_2, F_2\}, \ \Gamma(x) = G\Upsilon(x), \ G = \operatorname{diag}\{G_2, G_2\}$$

$$F_2 = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, G_2 = \begin{bmatrix} T^2/2 \\ T \end{bmatrix}, \Upsilon(x) = \begin{bmatrix} c(x) & -s(x) \\ s(x) & c(x) \end{bmatrix}$$

where $\Upsilon(x)$ is the rotation matrix with

$$c(x) = \cos\phi = \frac{\dot{\mathsf{x}}}{\sqrt{\dot{\mathsf{x}}^2 + \dot{\mathsf{y}}^2}}, \ s(x) = \sin\phi = \frac{\dot{\mathsf{y}}}{\sqrt{\dot{\mathsf{x}}^2 + \dot{\mathsf{y}}^2}}$$

that maps a_k to the Cartesian coordinates, and $\phi = \arctan(\dot{\mathsf{x}}/\dot{\mathsf{y}})$ is the target heading angle. A target maneuver is described in the model through the control input $u_k = \Gamma(x_k)a_k$. Clearly the system mode described in (6.25) is a nonlinear model due to the dependency of a_k on x_k. But it is linear given x_k. This makes it possible for us to design the filter within a linear framework.

Acceleration Models

Different models of the acceleration a_k can be used [51] depending on the maneuver capabilities of the targets of interest in the tracking application. To be more specific. we develop maneuver detectors for *manned* maneuvering aircraft in mind.

The normal acceleration is induced by the lift forces and is usually the dominant one during the maneuver. It was proposed in [30] that the normal acceleration $a_n(t)$ be modeled as an asymmetric function $a_n(t) = \alpha + \beta e^{\gamma b(t)}$ where α, β, γ are design parameters depending on the particular target type and $b(t)$ is a zero-mean first-order Gauss-Markov process. The marginal probability density function, essential for further development of the maneuver detector, can be easily derived to be

$$f(a_n) = \frac{1}{\sqrt{2\pi}\,|\gamma(a_n - \alpha)|} e^{-\frac{1}{2\gamma^2}(\ln\frac{a_n - \alpha}{\beta})^2} \tag{6.26}$$

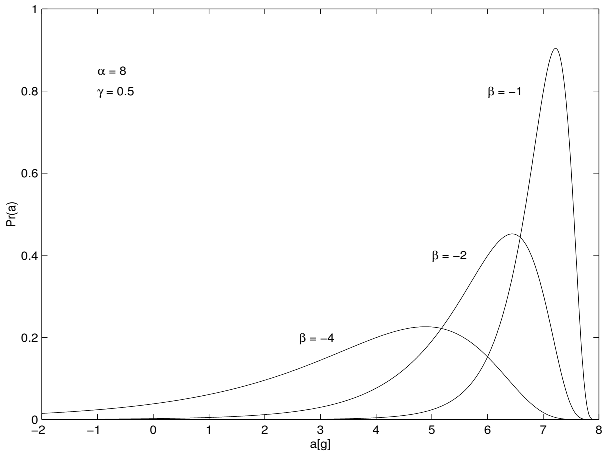


Figure 6.6: Asymmetric PDFs of normal acceleration.

For three typical choices of α, β, γ, this highly asymmetrical density is shown in Fig. 6.6, including one $[(\alpha, \beta, \gamma) = (8, -4, 0.5)]$ that is considered typical of modern piloted aircraft in evasive maneuvers. This model is more accurate than the usual symmetrical models (e.g., the Singer model) at the cost of choosing these parameters, which requires knowledge of the target type, obtained either a priori or a posteriori (e.g., with the help of an image-sensor-aided target classification) [51].

The tangential acceleration is determined by the *thrust-minus-drag* force. It is generally smaller in magnitude and shorter than the normal acceleration forse. This provides us a guideline for scenario designs. Various random process models for its magnitude are discussed and analyzed in detail in [51]. Most important for our maneuver detection approach is the choice of the marginal PDF of the process. Purely *a priori* symmetrical models (such as the ternary-uniform density in the Singer model) or *a posteriori*, adaptive (such as the conditional Rayleigh density in the mean-adaptive "current" model) are possible candidates.

Here we simply use a Gaussian marginal PDF model

$$f(a_t) = \mathcal{N}(a_t; \bar{a}_t, \sigma_t^2) \triangleq \frac{1}{\sqrt{2\pi}\sigma_t} e^{-\frac{(a_t-\bar{a}_t)^2}{2\sigma_t^2}} \tag{6.27}$$

with parameters $\bar{a}_t, \sigma_t^2$ either specified *a priori* (e.g., with some *nominal* values for the targets of interest) or determined *a posteriori* through filter-based estimates $\bar{a}_t = \hat{a}_{t_{k|k-1}}, \sigma_t^2 = \hat{\sigma}^2_{t_{k|k-1}}$. Both cases were investigated.

Under the assumption of independence between tangential and normal maneuvers, the PDF of the total acceleration vector is $f(a) = f(a_t, a_n) = f(a_t)f(a_n)$. For particular implementation of maneuver detectors in the sequel it is assumed also that direct position measurements of the target are available, i.e., $H = \text{diag}\left\{ \begin{bmatrix} 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \end{bmatrix} \right\}$ in (6.2).

Test Statistics

The key to computing test statistics of the two detectors is to obtain likelihood functions $f(\tilde{z}_k|H_i, z^{k-1})$ $(i = 0,\ 1)$. The marginal likelihood of H_0 is available in (6.17).

$$f(\tilde{z}_k|H_0, z^{k-1}) = \mathcal{N}(\tilde{z}_k; 0, S_k) = \frac{1}{|2\pi S_k|^{\frac{1}{2}}} e^{-\frac{1}{2}\tilde{z}_k{}' S_k^{-1} \tilde{z}_k}$$

To obtain $f(\tilde{z}_k|H_1, z^{k-1})$, we implement the marginalization approach (6.19) with the acceleration PDF models (6.26)-(6.27). However, (6.25) is a nonlinear system. In order to facilitate the calculation of test statistics under H_1, we write $f(\tilde{z}_k|H_1, z^{k-1}) = f(\tilde{z}_k|H_1, \hat{x}_{k-1|k-1})$ to make the nonlinear maneuver model (under H_1) *contionally linear* again. Then, according to (6.19) and (6.25), we have[1]

$$f(\tilde{z}|H_1, \hat{x}) = E[f(\tilde{z}|H_1, \hat{x}, a)] = \int \mathcal{N}(\tilde{z}; H\Gamma(\hat{x})a, S) f(a) da$$

$$= \int \int \mathcal{N}(\tilde{z}; H\Gamma(\hat{x})[a_t, a_n], S) f(a_t) f(a_n) da_t da_n \tag{6.28}$$

where $f(a_t)$ and $f(a_n)$ are given by (6.26) and (6.27), respectively.

Due to the complex prior density $f(a_n)$, which makes the integral in (6.28) no analytical solution available, we employ the Gaussian sum approximation techniques

$$f(a_n) \approx \sum_{i=1}^{N} \lambda_i \mathcal{N}(a_n; \bar{a}_n^{(i)}, \sigma_n^{(i)2}) \tag{6.29}$$

For our implementation of maneuver detection we obtained a fairly accurate approximation of the asymmetric $f(a_n)$ of (6.26) with a sum of only two components, which is illustrated in Fig. 6.7. Note that the Gaussian sum approximation technique can be also applied to $f(a_t)$ if a non-Gaussian model is adopted. As such,

$$f(a) = \sum_{i=1}^{N} \lambda_i \mathcal{N}(a; \bar{a}^{(i)}, \Lambda^{(i)})$$

where $\bar{a}^{(i)} = [\bar{a}_t, \bar{a}_n^{(i)}]'$ and $\Lambda^{(i)} = \text{diag}\{\sigma_t^2, \sigma_n^{(i)2}\}$. Then evaluating the likelihood (6.28) reduces to

$$f(\tilde{z}|H_1, \hat{x}) = \sum_{i=1}^{N} \lambda_i \mathcal{I}_i(\tilde{z}; \hat{x}) \tag{6.30}$$

where according to (6.23),

$$\mathcal{I}_i(\tilde{z}; \hat{x}) = \int \int \mathcal{N}(\tilde{z}; H\Gamma(\hat{x})a,\ S) \mathcal{N}(a; \bar{a}^{(i)}, \Lambda^{(i)}) da \tag{6.31}$$

$$= \mathcal{N}(\tilde{z}; H\Gamma(\hat{x})\bar{a}^{(i)},\ S + H\Gamma(\hat{x})\Lambda^{(i)}(H\Gamma(\hat{x}))') \tag{6.32}$$

[1] We drop the time index to simplify the notation.

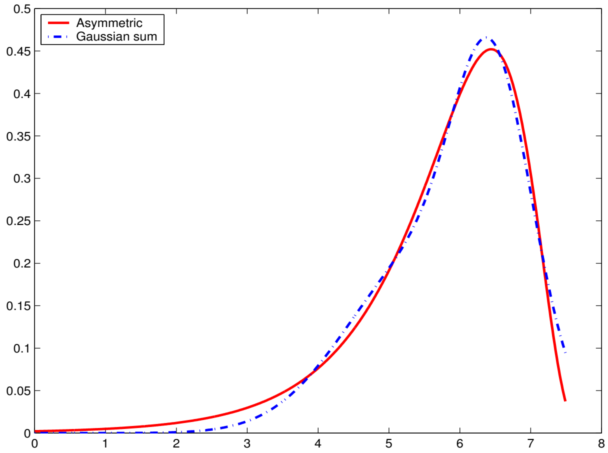


Figure 6.7: Gaussian sum approximation of the asymmetric PDF

6.7 Performance Evaluation

Performance of the proposed algorithms was evaluated through simulations. Three scenarios were designed to compare different aspects of the detection algorithms using ground truth trajectories generated by a curvilinear-motion model. All results are averages over 100 Monte Carlo runs.

6.7.1 Ground-Truth Model

The standard 2D curvilinear-motion model

$$\dot{x}(t) = v(t)\cos\phi(t) + w_x(t) \tag{6.33}$$

$$\dot{y}(t) = v(t)\sin\phi(t) + w_y(t) \tag{6.34}$$

$$\dot{v}(t) = a_t(t) + w_v(t) \tag{6.35}$$

$$\dot{\phi}(t) = a_n(t)/v(t) + w_\phi(t) \tag{6.36}$$

was used to generate the ground truth trajectories, where (x, y), v, ϕ denote the target position, speed and heading. This model is fairly general since it accounts for along- and cross-track accelerations.

The initial states used for each scenario were chosen to be

$$\begin{aligned}
x_0^i &\sim \mathcal{N}(\bar{x}_0,\sigma_{x_0}^2) & \bar{x}_0 &= 120km, & \sigma_{x_0} &= 5km \\
y_0^i &\sim \mathcal{N}(\bar{y}_0,\sigma_{y_0}^2) & \bar{y}_0 &= 150km, & \sigma_{y_0} &= 5km \\
v_0^i &\sim \mathcal{N}(\bar{v}_0,\sigma_{v_0}^2) & \bar{v}_0 &= 300m/s, & \sigma_{v_0} &= 5m/s \\
\phi_0^i &\sim \mathcal{N}(\bar{\phi}_0,\sigma_{\phi_0}^2) & \bar{\phi}_0 &= 30\deg, & \sigma_{\phi_0} &= 1\deg
\end{aligned}$$

with

$$\sigma_{w_x} = 5m, \sigma_{w_y} = 5m, \sigma_{w_v} = 1m/s, \sigma_{w_\phi} = 0.01\deg$$

The scenarios included both deterministic and random ones. The target track length starts from $k = 30$ and continues for another 70sec. The sampling time $T = 1s$.

(DN) *Deterministic normal acceleration scenario.* The target makes a normal maneuver during the period $k = [80, 100]$ with a magnitude of 20 m/s^2.

(DT) *Deterministic tangential acceleration scenario.* The target makes a tangential maneuver during the period $k = [80, 100]$ with a magnitude of 20 m/s^2.

(RN) *Random normal acceleration scenario.* In order to test detectors on multiple maneuvering scenarios simultaneously, a random maneuver scenario was developed, where the magnitude of the normal acceleration was fixed during $k = [80, 100]$ but chosen randomly over runs: $a_n \sim f(a_n)$ as given by (6.26) with $(\alpha, \beta, \gamma) = (4, -2, 0.5)$.

6.7.2 Simulation Results

There were some parameters to be designed. The threshold for each detector was determined by simulations with given false alarm rate $P_{fa} = 1\%$. Two-point differencing was used to initialize the Kalman filter. The corresponding noise covariances were $Q = (0.5)^2 I$ and $R = (50)^2 I$.

For the SSPRT detector, the prior probability for H_1 being true was 0.01 and transition probability π from H_0 to H_1 was 0.005. In addition, prior densities for testing normal accelerations were used by the CUSUM and SSPRT detectors: $f(a_t) = \mathcal{N}(a_t; 0, (0.5g)^2)$ and $f(a_n) \approx 0.44\mathcal{N}(a_n; 0.65g,\ (1.3g)^2) + 0.56\mathcal{N}(a_n; 2.2g, (0.6g)^2)$ which were determined by a local fit. Such design is obviously better to detect normal maneuvers since prior $f(a_t)$ will not affect normal accelerations much. It is possible that we could design a model with prior densities particularly good for tangential accelerations. In this simulation, we design $f(a_t) = \mathcal{N}(a_t; 0, (3.5g)^2)$ and $f(a_n)$ with zero mean and very small variance to test tangential accelerations. We could have a different design if we have better knowledge of prior information of tangential accelerations. For targets with both maneuvers, we consider combinations of two or more models, which is under investigation [89].

The CUSUM and SSPRT represent the two proposed detectors. The MR represents a detector based on measurement residuals. The IE represents one based on input estimation technique. Details of MR and IE can be found in Sec 6.3. The size of data window used in IE and MR detectors was 5. The performance of these four detectors was compared in terms of average onset detection delay $(\hat{n} - n)$, ROC curves and computational load.

Average Delay of Maneuver Onset Detection

The average onset detection delay under the four scenarios is shown in Table 6.1. It is clear that SSPRT detectors have smaller detection delay in both scenarios with normal acceleration as well as tangential acceleration. For the cases of tangential acceleration, the performance of the CUSUM and SSPRT detectors could be improved if we have the proper prior knowledge for the tangential acceleration.

Table 6.1: Average delay of maneuver onset detection

$\hat{n} - n$	DN	RN	DT
CUSUM	5.33	5.57	5.75
SSPRT	4.68	4.69	5.10
MR	5.19	5.12	5.15
IE	5.04	5.03	5.11

Furthermore, the SSPRT detector also provides the posterior probability of a maneuver. The posterior hypothesis probability for scenario DN is shown in Fig. 6.8. We note that posterior probability of H_1 quickly increases after the onset time. Similar results were observed for scenario DT.

ROC Curves

The ROC curves for deterministic normal acceleration scenarios were generated by computing the P_d at the time of interest with different P_{fa} using 100 Monte Carlo runs. The ROC curves for each maneuver onset detector at $k = 83$ and $k = 84$ for scenario DN are given

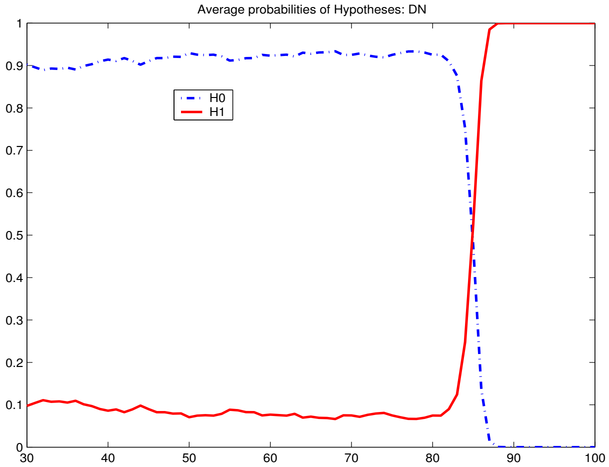


Figure 6.8: Posterior probability: normal accelerations change at $k = 80$

in Fig. 6.9 and Fig. 6.10, respectively. It is apparent from the curves that SSPRT has the best performance for the tested scenarios. The CUSUM detector outperforms IE and MR detectors for scenario DN with $P_{fa} > 0.2$. Note that IE performed better as time goes on due to the improved accuracy of the input estimate as more maneuver data become available. For ROC curve at $k = 83$ is shown in Fig. 6.11 for the scenario DT. The SSPRT and IE have compatible performance with $0.1 < P_{fa} < 0.2$, and the SSPRT outperforms other detectors with $P_{fa} > 0.2$. The ROC curves verify what we observed for maneuver onset detection delay.

Table 6.2: Computational complexity of different algorithms

CPU(s)	MR	CUSUM	SSPRT	IE
	1	2.97	3.27	7.61

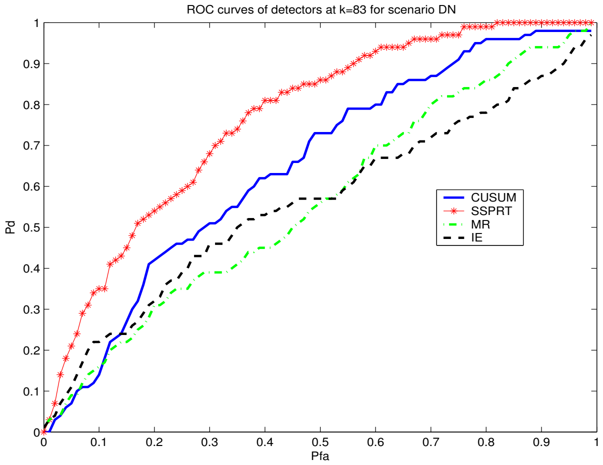


Figure 6.9: ROC curves of all detectors for scenario DN

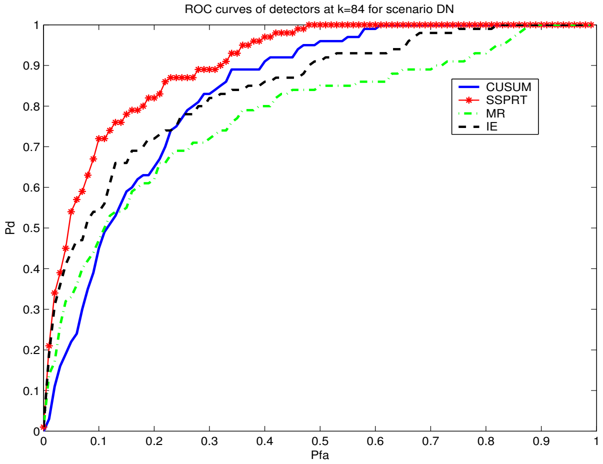


Figure 6.10: ROC curves of all detectors for scenario DN

Computational Complexity

The computational complexities of different algorithms were compared by using the ratio of the CPU processing time per iteration in Table 6.2. All other algorithms (CUSUM and SSPRT) were compared with the MR detector, which is the simplest. It shows that the proposed detection algorithms have a much less computational load than the IE-based algorithms.

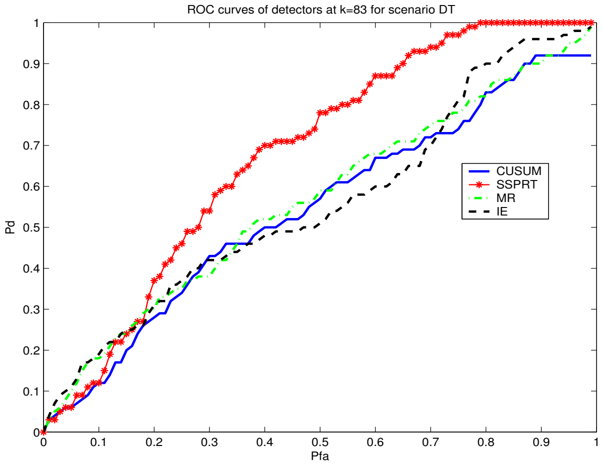


Figure 6.11: ROC curves of all detectors for scenario DT

6.8 Summary

In this chapter, existing target maneuver detection algorithms are first compared in a number of typical maneuvering target tracking scenarios. Then two sequential detectors for maneuver onset were developed and evaluated over various scenarios designed for different aspects of maneuvers. The proposed detection algorithms are based on sequential testing procedures, CUSUM and SSPRT, and use properly marginalized likelihoods to cope with the unknown magnitude of maneuver accelerations – a key difficulty in maneuver onset detection. Both the CUSUM and SSPRT detection algorithms developed are explicit, recursive and general. Utilizing prior distributions of the accelerations is essential in the application of the proposed scheme. In particular the detection procedure is developed for a typical 2D target with curvilinear motions, which uses conditioning to make the system model under H_1 within a linear framework. Performance evaluation and comparison have demonstrated that the proposed detectors are more effective than two popular detectors when appropriate prior distributions of accelerations are known.

As we realize, under a similar mathematical problem formulation, there exists a duality between actuator failure detection and target maneuver detection [31]. This motivates us to cross-apply the proposed sequential testing procedures of maneuver detection for fault detection. Fault detection has been considered as a change-point detection problem in the statistical literature. Some sequential tests have been recently developed for fault detection in dynamic systems [63, 76, 93]. In [88], two sequential detectors for actuator failure detection in a Boeing 747 aircraft have been developed in a general setting and evaluated through modeled data as well as real data. The proposed detection algorithms are based on the proposed sequential testing procedures (CUSUM and SSPRT) in Sec. 6.6 with the use of properly marginalized likelihoods to cope with the unknown magnitude of failures – a major difficulty in fault detection. Performance evaluation and comparison have demonstrated that the proposed detectors are more cost effective than other popular detectors when appropriate prior distributions of failures are known.

Chapter 7

Conclusion and Future Research

The main topics of this dissertation can be divided into two areas: including hybrid estimation and sequential change point detection with applications. For hybrid estimation, the multiple model approach, particularly variable-structure MM estimation, is studied theoretically as well as practically. Efficient and generally applicable VSMM algorithms are developed and successfully applied to fault detection and diagnosis as well as target tracking. Related work has been published in [85, 86, 54, 53, 102, 87, 101]. Based on our preliminary results, the multiple model method and especially VSMM are promising techniques to solve challenging problems such as hybrid estimation. The dissertation formulates target maneuver onset detection as a change point detection problem. Based on advanced sequential statistical tests, explicit and general detectors are developed and evaluated via various tracking scenarios. The algorithms can be applied to other applications under similar formulations such as fault detection. Related work has been published in [82, 84, 88, 89].

My current research will naturally lead to further exploration of adaptive estimation and change detection. This is a multi-discipline research field with a broad range of applications including air traffic control system, fault detection, industrial process control, communications, medical informatics, robotic systems, etc. There are still many challenging problems open in the area of the multiple model estimations, such as model-set design, smart mode generation, and cooperation strategy design. Incipient failure detection is also an interesting topics for fault detection, which is a key issue for automated system maintenance since early warning is important and required to avoid more serious consequences.

Currently I am working on target maneuver detection with range rate measurements, a continuation of my previous work. Due to the high nonlinearity between range rate and target states, it is not easy to implement within a conventional Kalman filtering framework, which is simple and available for a set of existing target maneuver detectors. Thus we propose a simple measurement conversion technique to treat range rate as a linear measurement in Cartesian coordinates so that a standard Kalman filter can be used for target tracking. Some issues need further study including tracker design based on the proposed detectors, the gain of using range rate for detection and estimation, and development of new statistics based on range rate measurements. Initial research result was published in [83].

Fault detection and estimation actually involve joint detection and estimation since the objective is to not only detect the presence of fault and identify its type but also estimate its severity and state. The existing solutions to this problem are mainly decision-based where the estimation performance depends on the detector. My future research work is to consider the framework of joint decision and estimation, to propose solutions, and to implement them for real problems. This structure has very promising potentials to applications in fault detection and diagnosis, and ground target tracking.

I am also interested in the research topic of sensor fusion. An efficient multi-sensor fusion procedure could efficiently extract true information, which is hidden in distorted multi-sensor data. Information from multiple sensors can reduce overall uncertainty and thus increase the accuracy of the measurement, which finally improves the detection performance. Sensor fusion has great potential for applications in system identification, pattern recognition, fault detection, image processing, and target classification and tracking.

Appendix A

Convex Combination of Estimates: EMA approach

Here we explain and justify the proposed EMA approach, as we did in [53]. Denote by $\hat{x}_1$ and $\hat{x}_2$ any two distinct unbiased estimators of x. Define a new estimator as a convex combination of $\hat{x}_1$ and $\hat{x}_2$:

$$\hat{x} = \alpha_1 \hat{x}_1 + \alpha_2 \hat{x}_2$$

where $\alpha_1 + \alpha_2 = 1$, $0 < \alpha_i < 1$. We would like to determine the condition under which $\hat{x}$ is better than $\hat{x}_1$. From

$$\tilde{x} = x - \hat{x} = \alpha_1 \tilde{x}_1 + \alpha_2 \tilde{x}_2, \quad \tilde{x}_i = x - \hat{x}_i$$

it follows that the mean-square error (mse) of $\hat{x}$ is

$$\mathcal{E} = E[\tilde{x}'\tilde{x}] = E[(\alpha_1 \tilde{x}_1 + \alpha_2 \tilde{x}_2)'(\alpha_1 \tilde{x}_1 + \alpha_2 \tilde{x}_2)]$$

$$= \alpha_1^2 E[\tilde{x}_1'\tilde{x}_1] + \alpha_2^2 E[\tilde{x}_2'\tilde{x}_2] + 2\alpha_1\alpha_2 E[\tilde{x}_1'\tilde{x}_2]$$

Since $\hat{x}_1$ and $\hat{x}_2$ are unbiased, if their estimation errors are uncorrelated (i.e., orthogonal), we have

$$\mathcal{E} = \alpha_1^2 \mathcal{E}_1 + \alpha_2^2 \mathcal{E}_2$$

where $\mathcal{E}_i$ is the mse of $\hat{x}_i$. Since $\alpha_1 = 1 - \alpha_2$, we have $\mathcal{E} < \mathcal{E}_1$ if and only if

$$(1 - 2\alpha_2 + \alpha_2^2)\mathcal{E}_1 + \alpha_2^2 \mathcal{E}_2 < \mathcal{E}_1$$

or equivalently

$$\mathcal{E}_2 < \frac{2 - \alpha_2}{\alpha_2} \mathcal{E}_1$$

That is, the convex combination $\hat{x}$ is better than $\hat{x}_1$ if and only if the mse$(\hat{x}_2) < \frac{2-\alpha_2}{\alpha_2}mse(\hat{x}_1)$ or equivalently

$$\alpha_2 < \frac{2\mathcal{E}_1}{\mathcal{E}_1 + \mathcal{E}_2}$$

This inequality is always satisfied if $\mathcal{E}_2 < \mathcal{E}_1$, that is, if $\hat{x}_2$ is better than $\hat{x}_1$, as illustrated in Figure A.1(a). Even if $\hat{x}_2$ is worse than $\hat{x}_1$, this inequality shows that $\hat{x}$ is still better than $\hat{x}_1$ provided α_2 satisfies the above inequality, as illustrated in Figure A.1(b). In Figure A.1, the distance measure is the standard error (square-root of mse) $||\tilde{x}|| = (E[\tilde{x}'\tilde{x}])^{1/2} = (\mathcal{E})^{1/2}$; the line that connects $\tilde{x}_1$ and $\tilde{x}_2$ represents all possible points of $\tilde{x}$ and its solid line portion represents the points at which $\hat{x}$ is better than $\hat{x}_1$.

The above result relies on the assumption that $\tilde{x}_1$ and $\tilde{x}_2$ are orthogonal. As illustrated in Figures A.1(c) and A.1(d), if they are not orthogonal, it is possible for $\hat{x}$ to be better than $\hat{x}_1$ (i.e., $\hat{x}$ with the optimal α_1 and α_2 is better than $\hat{x}_1$) if and only if

$$E[\tilde{x}_1'\tilde{x}_2] < E[\tilde{x}_1'\tilde{x}_1]$$

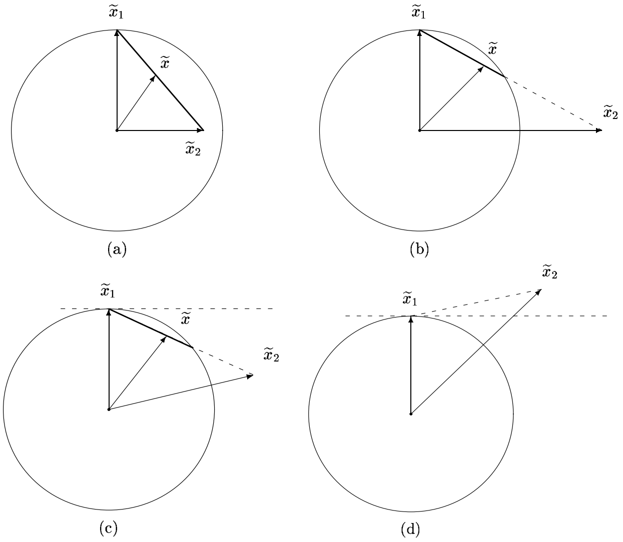


Figure A.1: Conditions for Estimate Improvement

that is, the projection of $\tilde{x}_2$ on $\tilde{x}_1$ either has a smaller magnitude than $\tilde{x}_1$, if they are in the same direction, or has a direction that is opposite to $\tilde{x}_1$. In other words, use of $\hat{x}_2$ may still be beneficial even if $\hat{x}_2$ is worse than $\hat{x}_1$ and their estimation errors are correlated. In fact, it can be easily shown that the optimal $\alpha_2 = \frac{\mathcal{E}_1}{\mathcal{E}_1+\mathcal{E}_2}$ is the bisecting point of the solid line portion (Figure A.1) at which $\tilde{x}$ is orthogonal to $\tilde{x}_2 - \tilde{x}_1$.

Appendix B

MLE of α for Sensor Partial Failures

The log-marginal likelihood function is

$$\begin{aligned}
&\ln f(z_k|z^{k-1}, a_k) \\
&= -\frac{1}{2}\ln(|2\pi S_k|) - \frac{1}{2}(z_k - H_k\widehat{x}_{_{k|k-1}})'S_k^{-1}(z_k - H_k\widehat{x}_{_{k|k-1}})
\end{aligned}$$

Taking the derivative and setting it to zero with $\tilde{z}_{k|k-1} = z_k - H_k\widehat{x}_{_{k|k-1}}$, $A_k = S_k^{-1}$, we have

$$\frac{d\ln f(z_k|z^k, \alpha_k)}{d\alpha_k} = 0 \Longrightarrow \frac{d\tilde{z}'_{k|k-1}A_k\tilde{z}_{k|k-1}}{d\alpha_k} = 0 \tag{B.1}$$

Since

$$\begin{aligned}
\frac{d\tilde{z}'_{k|k-1}A_k\tilde{z}_{k|k-1}}{d\alpha_k} &= \tilde{z}'_{k|k-1}(A_k' + A_k)\frac{d\tilde{z}_{k|k-1}}{d\alpha_k} \\
\frac{d\tilde{z}_{k|k-1}}{d\alpha_k} = \frac{d(z_k - H_k\widehat{x}_{_{k|k-1}})}{d\alpha_k} &= -(\frac{dH_k}{d\alpha_k})\widehat{x}_{_{k|k-1}}
\end{aligned}$$

(B.1) becomes

$$\tilde{z}'_{k|k-1}(A_k' + A_k)(\frac{dH_k}{d\alpha_k})\widehat{x}_{_{k|k-1}} = 0$$

Since $\frac{dH_k}{d\alpha_k}$ is easy to obtain by the formula of matrix derivation with respect to a scalar, and other terms are given by the Kalman filter directly, the MLE of α can be obtained. The actuator failure effectiveness factor can be estimated similarly.

Appendix C

The PDF of Normal Acceleration

It is well known that the PDF $f_Y(y)$ of $Y = g(x)$ can be determined from the PDF $f(x)$ of X by

$$f_Y(y) = \sum\nolimits_i \frac{f_X(x_i)}{|g'(x_i)|}$$

where $g^{'}(x) = \frac{d}{dx}g(x)$ and x_i's are the real roots if the equation $y = g(x)$ in terms of y: $y = g(x_i)$. Let $g(b) = \alpha + \beta e^{\gamma b}$, then $g'(b) = \beta\gamma e^{\gamma b}$. The root of $a_n = \alpha + \beta e^{\gamma b}$ is $b_1 = \frac{1}{\gamma}\log\frac{a_n-\alpha}{\beta}$. Thus we have

$$\begin{aligned} f\left(a_n\right) = \frac{f(b_1)}{|g^{'}(b_1)|} &= \frac{\mathcal{N}(\frac{1}{\gamma}\log\frac{a_n-\alpha}{\beta};0,1)}{|\gamma(a_n-\alpha)|} \\ &= \frac{1}{\sqrt{2\pi}\,|\gamma(a_n-\alpha)|}e^{-\frac{1}{2\gamma^2}(\log\frac{a_n-\alpha}{\beta})^2} \end{aligned}$$

Appendix D

Marginal Likelihood Function under H 1 with Gaussian Prior

If the *a priori* probability density function $f\left(u\right)$ is Gaussian distributed,

$$f(u) = \mathcal{N}(u; \bar{u}, \Lambda)$$

then

$$\begin{aligned}
f\left(\tilde{z}_k|H_1, z^{k-1}\right) &= \int f\left(\tilde{z}|H_1\left(u\right), z^{k-1}\right) f\left(u\right) du \\
&= \int \mathcal{N}(\tilde{z}; HGu,\ S)\mathcal{N}(u; \bar{u}, \Lambda) du \\
&= \int \frac{1}{\sqrt{|2\pi S|}} e^{-\frac{1}{2}(\tilde{z}_k - HGu)' S_k^{-1}(\tilde{z} - HGu)} \frac{1}{\sqrt{|2\pi\Lambda|}} e^{-\frac{1}{2}(u-\bar{u})'\Lambda^{-1}(u-\bar{u})} du
\end{aligned}$$

We drop the time index for simplicity. Let $m = HGu$, $\Lambda_m = HG\Lambda(HG)'$, and $\bar{m} = HG\bar{u}$. Then $dm = |HG|\, du$, and

$$
\begin{aligned}
f\left(\tilde{z}_k|H_1, z^{k-1}\right) &= \frac{1}{\sqrt{|(2\pi)^2 S\Lambda_m|}} \int e^{-\frac{1}{2}[(\tilde{z}-\bar{u})'S^{-1}(\tilde{z}-\bar{u})+(m-\bar{m})'\Lambda_m^{-1}(m-\bar{m})]} dm \\
&= \frac{1}{\sqrt{|(2\pi)^2 S\Lambda_m|}} \int e^{-\frac{1}{2}D} dm
\end{aligned}
$$

Define

$$
\begin{aligned}
&C_1 \triangleq S^{-1} + \Lambda_m^{-1},\ C_2 \triangleq S^{-1}\tilde{z} + \Lambda_m^{-1}\bar{m} \\
&c_3 \triangleq \tilde{z}'S^{-1}\tilde{z} + \bar{m}'\Lambda_m^{-1}\bar{m},\ D \triangleq m'C_1 m - 2m'C_2 + c_3
\end{aligned}
$$

and let $t = C_1^{1/2}(m - C_1^{-1}C_2)$, $dt = \left|C_1^{1/2}\right| dm$. Then

$$
\begin{aligned}
f\left(\tilde{z}_k|H_1, z^{k-1}\right) &= \frac{e^{-\frac{1}{2}(c_3 - C_2'C_1^{-1}C_2)}}{\sqrt{|2\pi S C_1 \Lambda_m|}} \frac{1}{\sqrt{2\pi}} \int e^{-\frac{1}{2}t't} dt \\
&= \frac{e^{-\frac{1}{2}(c_3 - C_2'C_1^{-1}C_2)}}{\sqrt{|2\pi S C_1 \Lambda_m|}}
\end{aligned}
$$

It can be directly verified that

$$
\begin{aligned}
c_3 - C_2'C_1^{-1}C_2 &= (\tilde{z} - \bar{m})'(S + \Lambda_m)^{-1}(\tilde{z} - \bar{m}) \\
SC_1\Lambda_m &= S + \Lambda_m
\end{aligned}
$$

Thus

$$
\begin{aligned}
f\left(\tilde{z}_k|H_1, z^{k-1}\right) &= \frac{1}{\sqrt{|2\pi S C_1 \Lambda_m|}} e^{-\frac{1}{2}(c_3 - C_2'C_1^{-1}C_2)} \\
&= \mathcal{N}(\bar{m}, S + \Lambda_m)
\end{aligned}
$$

Appendix E

Marginal Likelihood Function under H 1 with Uniform Prior

If the *a priori* probability density function $f(u)$ is uniformly distributed,

$$f(u) = \mathcal{U}(u; -u_{\max}, u_{\max})$$

we have

$$\begin{aligned} f\left(\tilde{z}_k | H_1, z^{k-1}\right) &= \int f\left(\tilde{z} | H_1(u), z^{k-1}\right) f(u)\, du \\ &= \int \mathcal{N}(\tilde{z}_k; HGu,\ S_k)\, \mathcal{U}(u; -u_{\max}, u_{\max}) du \\ &= \int_{-u_{\max}}^{u_{\max}} \frac{1}{\sqrt{|2\pi S|}} e^{-\frac{1}{2}(\tilde{z}_k - HGu)' S_k^{-1} (\tilde{z}_k - HGu)} \frac{1}{2u_{\max}} du \end{aligned}$$

Let $m = HGu$, and drop the time index for simplicity, then

$$f\left(\tilde{z}_k | H_1, z^{k-1}\right) = \frac{1}{2u_{\max}\,|HG|} \int_{-HGu_{\max}}^{HGu_{\max}} \frac{1}{\sqrt{|2\pi S|}} e^{-\frac{1}{2}(\tilde{z} - m)' S^{-1} (\tilde{z} - m)} dm$$

Let $C = \frac{1}{\sqrt{|2S|}}(m - \tilde{z})$, then $dC = \frac{1}{\sqrt{|2S|}} dm$, thus

$$\begin{aligned} f\left(\tilde{z}_k | H_1, z^{k-1}\right) &= \frac{1}{2u_{\max}\,|HG|} \int_{\frac{1}{\sqrt{|2S|}}(HGu_{\max}-\tilde{z})}^{\frac{1}{\sqrt{|2S|}}(HGu_{\max}-\tilde{z})} \frac{1}{\sqrt{|\pi|}} e^{-C'C} dC \\ &= \frac{1}{4u_{\max}\,|HG|}[erf(\frac{1}{\sqrt{|2S|}}(HGu_{\max} - \tilde{z})) + erf(\frac{1}{\sqrt{|2S|}}(HGu_{\max} + \tilde{z}))] \end{aligned}$$

where the erf function is defined as

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t} dt$$

Bibliography

[1] A. Averbuch, S. Itzikowitz, and T. Kapon. Radar Target Tracking—Viterbi versus IMM. *IEEE Trans. Aerospace and Electronic Systems*, AES-27(3):550–563, May 1991.

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

[3] Y. Bar-Shalom, editor. *Multitarget-Multisensor Tracking: Applications and Advances*, volume II. Artech House, Norwood, MA, 1992.

[4] Y. Bar-Shalom and K. Birmiwal. Variable Dimension Filter for Maneuvering Target Tracking. *IEEE Trans. Aerospace and Electronic Systems*, AES-18(5):621–629, Sept. 1982.

[5] Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association.* Academic Press, New York, 1988.

[6] Y. Bar-Shalom and X. R. Li. *Estimation and Tracking: Principles, Techniques, and Software.* Artech House, Boston, MA, 1993. (Reprinted by YBS Publishing, 1998).

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

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

[9] M. Basseville. Detecting Changes in Signals and Systems. *Automatica*, 24(3):309–326, May 1988.

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

[11] E. Bekir. Adaptive Kalman Filter for Tracking Maneuvering Targets. *AIAA Journal of Guidance*, 6(5):414–416, Sept.-Oct. 1983.

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

[13] S. S. Blackman and R. F. Popoli. *Design and Analysis of Modern Tracking Systems.* Artech House, Norwood, MA, 1999.

[14] H. A. P. Blom. A Sophisticated Tracking Algorithm for ATC Surveillance Data. In *Proc. International Radar Conf.*, Paris, France, May 1984.

[15] H. A. P. Blom. An Efficient Filter for Abruptly Changing Systems. In *Proc. 23rd IEEE Conf. Decision and Control*, Las Vegas, NV, Dec. 1984.

[16] H. A. P. Blom and Y. Bar-Shalom. The Interacting Multiple Model Algorithm for Systems with Markovian Switching Coefficients. *IEEE Trans. Automatic Control*, AC-33(8):780–783, Aug. 1988.

[17] P. L. Bogler. Tracking a Maneuvering Target Using Input Estimation. *IEEE Trans. Aerospace and Electronic Systems*, AES-23(3):298–310, May 1987.

[18] W. S. Chaer, R. H. Bishop, and J. Ghosh. Hierarchical Adaptive Kalman Filter for Interplanetary Orbit Determination. *IEEE Trans. Aerospace and Electronic Systems*, 34(3):883–895, July 1998.

[19] L. H. Chiang, E. L. Russell, and R. D. Braatz. *Fault Detection and Diagnosis in Industrial Systems.* Springer, 2001.

[20] J. R. Cloutier, C. F. Lin, and C. Yang. Maneuvering Target Tracking via Smoothing and Filtering Through Measurement Concatenationn. *AIAA Journal of Guidance, Control, and Dynamics*, 16(2):377–384, March-Apr. 1993.

[21] M. Efe and D. P. Atherton. The IMM Approach to the Fault Detection Problem. In *11th IFAC Symp. on System Identification*, Fukuoka, Japan, July 1997.

[22] K. A. Fisher and P. S. Maybeck. Multiple Model Adaptive Estimation with Filtering Spawning. *IEEE Trans. Aerospace and Electronic Systems*, AES-38(3):755–768, 2002.

[23] P. M. Frank. Fault Diagnosis in Dynamic Systems Using Analytical and Knowledge-Based Redundancy—A Survey and some New Results. *Automica*, 26:459–474, 1990.

[24] C. M. Fry and A. P. Sage. On Hierarchical Structure Adaptation and Systems Identification. *Int. J. Control*, 20(3):433–452, 1974.

[25] M. Gauvrit. Bayesian Adaptive Filter for Tracking with Measurements of Uncertain Origin. *Automatica*, 20:217–224, Mar. 1984.

[26] J. Gertler. Survey of Model-Based Failure Detection and Isolation in Complex Plants. *IEEE Control Systems Magazine*, 8(6):3–11, 1988.

[27] F. Gustafsson. The Marginalized Likelihood Test for Detecting Abrupt Changes. *IEEE Trans. Automatic Control*, AC-41:66–78, Jan. 1996.

[28] F. Gustafsson. *Adaptive Filtering and Change Detection.* Wiley, 2001.

[29] R. Isermann. Process Fault Detection Based on Modeling and Estimation Methods —A Survey. *Automatica*, 20(4):387–404, July 1984.

[30] J. D. Kendrick, P. S. Maybeck, and J. G. Reid. Estimation of Aircraft Target Motion Using Orientation Measurements. *IEEE Trans. Aerospace and Electronic Systems*, 17(2):254–260, Mar. 1981.

[31] T. H. Kerr. Duality Between Failure Detection and Radar/Optical Maneuver Detection. *IEEE Trans. Aerospace and Electronic Systems*, AES-25:520–528, July 1989.

[32] J. Korn, S. W. Gully, and A. S. Willsky. Application of the Generalized Likelihood Ratio Algorithm to Maneuver Detection and Estimation. In *Proc. 1982 American Control Conf.*, Arlington, VA, June 1982.

[33] T. L. Lai. Sequential Changepoint Detection in Quality Control and Dynamical Systems. *J. R. Statist. Soc.*, 57(4):613–658, 1995.

[34] T. L. Lai. Sequential Multiple Hypothesis Testing and Efficient Fault Detection-Isolation in Stochastic Systems. *IEEE Trans. on Information Theory*, 46(2):595–608, Mar. 2000.

[35] T. L. Lai and J. Z. Shan. Efficient Recursive Algorithms for Detection of Abrupt Changes in Signals and Controls Systems. *IEEE Trans. on Automatic Control*, 44(5):952–966, May 1999.

[36] D. G. Lainiotis. Optimal Adaptive Estimation: Structure and Parameter Adaptation. *IEEE Trans. Automatic Control*, 16(2):160–170, April 1971.

[37] D. G. Lainiotis. Partitioning: A Unifying Framework for Adaptive Systems, I: Estimation. *Proc. IEEE*, 64(8):1126–1143, Aug. 1976.

[38] J. Layne. Monopulse Radar Tracking Using an Adaptive Interacting Multiple Model Method with Extended Kalman Filters. In *Proc. 1998 SPIE Conf. on Signal and Data Processing of Small Targets,* vol. 3373, Orlando, FL, Apr. 1998.

[39] H. Lee and M.-J. Tahk. Generalized input-estimation technique for tracking maneuvering targets. *IEEE Transactions on Aerospace and Electronic Systems*, 35(4):1388–1402, 1999.

[40] S.-C. Lee and C.-Y. Liu. Trajectory Estimation of Reentry Vehicle by Use of on-Line Input Estimator. *Journal of Guidance, Control, and Dynamics*, 22(6):808–, 1999.

[41] X. R. Li. Multiple-Model Estimation with Variable Structure: Some Theoretical Considerations. In *Proc. 33rd IEEE Conf. on Decision and Control*, pages 1199–1204, Orlando, FL, Dec. 1994.

[42] X. R. Li. Hybrid Estimation Techniques. In C. T. Leondes, editor, *Control and Dynamic Systems: Advances in Theory and Applications*, volume 76, pages 213–287. Academic Press, New York, 1996.

[43] X. R. Li. Engineer's Guide to Variable-Structure Multiple-Model Estimation for Tracking. In Y. Bar-Shalom and W. D. Blair, editors, *Multitarget-Multisensor Tracking: Applications and Advances*, volume III, chapter 10, pages 499–567. Artech House, Boston, MA, 2000.

[44] X. R. Li. Multiple-Model Estimation with Variable Structure—Part II: Model-Set Adaptation. *IEEE Trans. Automatic Control*, AC-45(11):2047–2060, Nov. 2000.

[45] X. R. Li. Model-Set Design for Multiple-Model Estimation—Part I. In *Proc. 2002 International Conf. on Information Fusion*, pages 26–33, Annapolis, MD, USA, July 2002.

[46] X. R. Li and Y. Bar-Shalom. Mode-Set Adaptation in Multiple-Model Estimators for Hybrid Systems. In *Proc. 1992 American Control Conf.*, pages 1794–1799, Chicago, IL, June 1992.

[47] X. R. Li and Y. Bar-Shalom. A Recursive Multiple Model Approach to Noise Identification. *IEEE Trans. Aerospace and Electronic Systems*, AES-30(3):671–684, July 1994.

[48] X. R. Li and Y. Bar-Shalom. Multiple-Model Estimation with Variable Structure. *IEEE Trans. Automatic Control*, AC-41(4):478–493, Apr. 1996.

[49] X. R. Li and V. P. Jilkov. Expected-Mode Augmentation for Multiple-Model Estimation. In *Proc. 2001 International Conf. on Information Fusion*, pages WeB1.3–WeB1.10, Montreal, QC, Canada, Aug. 2001.

[50] X. R. Li and V. P. Jilkov. A Survey of Maneuvering Target Tracking—Part IV: Decision-Based Methods. In *Proc. 2002 SPIE Conf. on Signal and Data Processing of Small Targets,* vol. 4728, Orlando, Florida, USA, April 2002.

[51] X. R. Li and V. P. Jilkov. Survey of Maneuvering Target Tracking—Part I: Dynamic Models. *IEEE Trans. Aerospace and Electronic Systems*, AES-39(4):1333–1364, Oct. 2003.

[52] X. R. Li and V. P. Jilkov. A Survey of Maneuvering Target Tracking—Part V: Multiple-Model Methods. *IEEE Trans. Aerospace and Electronic Systems* (to appear), Jan. 2006. Also available at URL: "http://ece.engr.uno.edu/isl/Reprints/MTTSurveyPart5.pdf".

[53] X. R. Li, V. P. Jilkov, and J.-F. Ru. Multiple-Model Estimation with Variable Structure–part VI: Expected-Mode Augmentation. *to appear in IEEE Trans. Aerospace and Electronic Systems*, July 2005.

[54] X. R. Li, V. P. Jilkov, J.-F. Ru, and A. Bashi. Expected-Mode Augmentation Algorithms for Variable-Structure Multiple-Model Estimation. In *Proc. IFAC 15th World Congress*, Barcelona, Spain, July 2002. Paper no. 2816.

[55] X. R. Li and Y. M. Zhang. Multiple-Model Estimation with Variable Structure—Part V: Likely-Model Set Algorithm. *IEEE Trans. Aerospace and Electronic Systems*, AES-36(2):448–466, Apr. 2000.

[56] X. R. Li and Y. M. Zhang. Numerically Robust Implementation of Multiple-Model Algorithms. *IEEE Trans. Aerospace and Electronic Systems*, AES-36(1):266–278, Jan. 2000.

[57] X. R. Li, Y. M. Zhang, and X. R. Zhi. Multiple-Model Estimation with Variable Structure—Part IV: Design and Evaluation of Model-Group Switching Algorithm. *IEEE Trans. Aerospace and Electronic Systems*, AES-35(1):242–254, Jan. 1999.

[58] X. R. Li, Z.-L. Zhao, P. Zhang, and C. He. Model-Set Design for Multiple-Model Estimation—Part II: Examples. In *Proc. 2002 International Conf. on Information Fusion*, pages 1347–1354, Annapolis, MD, USA, July 2002.

[59] X. R. Li, X. R. Zhi, and Y. M. Zhang. Multiple-Model Estimation with Variable Structure—Part III: Model-Group Switching Algorithm. *IEEE Trans. Aerospace and Electronic Systems*, AES-35(1):225–241, Jan. 1999.

[60] H.-J. Lin and D. P. Atherton. An Investigation of the SFIMM Algorithm for Tracking Manoeuvring Targets. In *Proc. 32nd IEEE Conf. Decision and Control*, pages 930–935, San Antonio, TX, Dec. 1993.

[61] D. G. Luenberger. *Linear and Nonlinear Programming.* Addison-Wesley, Reading, Massachusetts, 2nd edition, 1984.

[62] D. T. Magill. Optimal Adaptive Estimation of Sampled Stochastic Processes. *IEEE Trans. Automatic Control*, AC-10:434–439, 1965.

[63] D. P. Malladi and J. L. Speyer. A Generalized Shiryaev Sequential Probability Ratio Test for Change Detection and Isolation. *IEEE Trans. Automatic Control*, AC-44(8):1522–1534, 1999.

[64] A. Marcos and G. J. Balas. A Boeing 747-100/200 Aircraft Fault Tolerant and Fault Diagnostic Benchmark. Technical Report AEM-UoM-2003-1, Dept. of Aerospace and Engineering Mechanics, University of Minnesota, June 2003.

[65] P. S. Maybeck. Application of Multiple Model Adaptive Algorithms to Reconfigurable Flight Control. *Control and Dynamic Systems*, 52:291–320, 1992.

[66] P. S. Maybeck and P. D. Hanlon. Performance Enhancement of a Multiple Model Adaptive Estimator. In *Proc. 32nd IEEE Conf. on Decision and Control*, pages 462–268, San Antonio, TX, Dec. 1993. Also in IEEE Trans. Aerospace and Electronic Systems, Oct. 1995.

[67] P. S. Maybeck and P. D. Hanlon. Performance Enhancement of a Multiple Model Adaptive Estimator. *IEEE Trans. Aerospace and Electronic Systems*, AES-31(4):1240–1254, Oct. 1995.

[68] P. S. Maybeck and K. P. Hentz. Investigation of Moving-Bank Multiple Model Adaptive Algorithms. *AIAA J. Guidance, Control, and Dynamics*, 10(1):90–96, Jan.-Feb. 1987.

[69] P. S. Maybeck and R. D. Stevens. Reconfigurable Flight Control Via Multiple Model Adaptive Control Methods. *IEEE Trans. Aerospace and Electronic Systems*, AES-27(3):470–480, May 1991.

[70] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan. Interacting Multiple Model Methods in Target Tracking: A Survey. *IEEE Trans. Aerospace and Electronic Systems*, AES-34(1):103–123, 1998.

[71] R. K. Mehra, C. Rago, and S. Seereeram. Failure Detection and Identification using a Nonlinear Interactive Multiple Model (IMM) Filtering Approach with Aerospace Applications. In *11th IFAC Symp. on System Identification*, Fukuoka, Japan, July 1997.

[72] T. E. Menke and P. S. Maybeck. Sensor/Actuator Failure Detection in the Vista F-16 by Multiple Model Adaptive Estimation. *IEEE Trans. Aerospace and Electronic Systems*, AES-31(4):1218–1229, Oct. 1995.

[73] A. Munir and D. P. Atherton. Adaptive Interacting Multiple Model Algorithm for Tracking a Manoeuvring Target. *IEE Proc.–Radar, Sonar, and Navigation*, 142(1):11–17, Feb. 1995.

[74] K. S. Narenda and S. S. Tripathi. Identification and Optimazation of Aircraft Dynamics. *Journal of Aircraft*, 10:193–199, Jan. 1973.

[75] I. Nikiforov. A Simple Recursive Algorithm for Diagnosis of Abrupt Changes in Random Signals. *IEEE Trans. Information Theory*, 2000.

[76] I. Nikiforov. Optimal Sequential Change Detection and Isolation. In *Proc. 15th IFAC Word Congress*, Barcelona, Spain, July 2002.

[77] I. Nikiforov. A Lower Bound for the Detection/Isolation Delay in a Class of Sequential Tests. *IEEE Trans. Information Theory*, 2003.

[78] R. J. Patton. Fault Detection and Diagnosis in Aerospace Systems Using Analytical Redundancy. *Computering and Control Engineering Journal*, pages 127–136, May 1991.

[79] R. J. Patton, P. M. Frank, and R. R. Clark. *Fault Diagnosis in Dynamic Systems, Theory and Applications.* Prentice-Hall, Englewood Cliffs, NJ, 1989.

[80] C. Rago and R. K. Mehra. Failure Detection and Identification: A Multiple Model Approach for the Multirate Case. In *Proceedings of the Workshop on Estimation, Tracking and Fusion: A Tribute to Yaakov Bar-Shalom*, pages 513–527, Monterey, CA, May 2001.

[81] C. Rago, R. Prasanth, R. K. Mehra, and R. Fortenbaugh. Failure Detection and Identification and Fault Tolerant Control Using the IMM-KF with Applications to the Eagle-Eye UAV. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 4208–4213, Tampa, FA, Dec. 1998.

[82] J.-F. Ru, A. Bashi, and X. R. Li. Performance Comparison of Target Maneuver Onset Detection Algorithms. In *Proc. 2004 SPIE Conf. on Signal and Data Processing of Small Targets,* vol. 5428, pages 419–428, 2004.

[83] J.-F. Ru, H. Chen, and X. R. Li. A Range-rate-based Detection Technique for Tracking a Maneuvering Target, to appear in the Proc. SPIE Conf. Signal and Data Processing of Small Targets. Aug. 2005.

[84] J.-F. Ru, V. P. Jilkov, X. R. Li, and A. Bashi. Sequential Detection of Target Maneuvers, to appear in the IEEE International Conference on Information Fusion. Jul. 2005.

[85] J.-F. Ru and X. R. Li. Application of FDI to a Boeing 747 Aircraft by Multiple Model Approach. Technical report, Dept. of Electrical Engineering, University of New Orleans, Nov. 2003.

[86] J.-F. Ru and X. R. Li. Interacting Multiple Model Algorithm with Maximum Likelihood Estimation for FDI. In *Proceedings of the 2003 IEEE International Symposium on Intelligent Control*, pages 661–666, Houston, TX, Oct. 2003.

[87] J.-F. Ru and X. R. Li. Variable-Structure Multiple-Model Approach to Fault Detection, Identification and Estimation. *submitted to AIAA Journal of Guidance, Control and Dynamics*, Mar. 2005.

[88] J.-F. Ru, X. R. Li, and V. P. Jilkov. Sequential Fault Detection of a Boeing 747 Aircraft. Technical report, Dept. of Electrical Engineering, University of New Orleans, Mar. 2004.

[89] J.-F. Ru, X. R. Li, and V. P. Jilkov. Multiple Model Detection of Target Maneuvers, to appear in the Proc. SPIE Conf. Signal and Data Processing of Small Targets. Aug. 2005.

[90] K. Schnepper. A Comparison of GLR and Multiple Model Filters for a Target Tracking Problem. In *Proceedings of the 25 Conference on Decision and Control*, pages 666–670, Athens, Greece, December 1986.

[91] M. L. Schwall, J. C. Gerdes, B. Baker, and T. Forchert. A probabilistic Vehicle Diagnostic System Using Multiple Models. In *Proc. 15th Innovative Applications of Artificial Intelligence Conference (IAAI-2003)*, Aug. 2003.

[92] H. W. Sorenson and D. L. Alspach. Recursive Bayesian Estimation Using Gaussian Sums. *Automatica*, 7:465–479, 1971.

[93] J. L. Speyer and J. E. White. Shiryaev Sequential Probability Ratio Test for Redundancy Management. *AIAA Journal of Guidance, Control, and Dynamics*, 7(5):588–595, 1984.

[94] T. K. Sung and J. G. Lee. A Decoupled Adaptive Tracking Filter for Real Applications. *IEEE Trans. Aerospace and Electronic Systems*, 33(3):1025–1030, 1997.

[95] I. Szaszi, S. Ganguli, A. Marcos, G. Balas, and J. Bokor. Application of FDI to a Nonlinear Boeing -747 Aircraft. In *Proceedings of Mediterranean Conference on Control and Automation*, Lisbon, Portugal, Jul. 2002.

[96] J. R. Vasquez and P. S. Maybeck. Enhanced Motion and Sizing of Bank in Moving-Bank MMAE. In *Proc. 1999 American Control Conf.*, pages 1555–1562, San Diego, CA, June 1999.

[97] T. C. Wang and P. K. Varshney. A Tracking Algorithm for Maneuvering Targets. *IEEE Trans. Aerospace and Electronic Systems*, AES-29(3):910–924, July 1993.

[98] K. Watanabe and S. G. Tzafestas. A Hierarchical Multiple Model Adaptive Control of Discrete-time Stochastic Systems for Sensor and Actuator Uncertainties. *Automatica*, 26(5):875–886, Sept. 1990.

[99] A. S. Willsky. A Survey of Design Methods for Failure Detection in Dynamic Systems. *Automatica*, 12(6):601–611, Nov. 1976.

[100] A. S. Willsky and H. L. Jones. A Generalized Likelihood Ratio Approach to the Estimation in Linear Systems Subject to Abrupt Changes. In *Proc. 1974 IEEE Conf. On Decision and Control*, Nov. 1974.

[101] M. Yang, J.-F. Ru, H. Chen, A. Bashi, X. R. Li, and N.S.V. Rao. Predicting Internet End-to-End Delay: A Statistical Study, to appear in *Annual Review of Communications*. vol. 58, 2005.

[102] M. Yang, J.-F. Ru, X. R. Li, H. Chen, and A. Bashi. Predicting Internet End-to-End Delay: A Multiple-Model Approach. In *Proceedings of the 2005 IEEE Global Internet*, Mar. 2005.

[103] Y. M. Zhang and J. Jiang. Integrated Active Fault-Tolerant Control Using IMM Approach. *IEEE Trans. Aerospace and Electronic Systems*, AES-37(4):1221–1235, Oct. 2001.

[104] Y. M. Zhang and X. R. Li. Detection and Diagnosis of Sensor and Actuator Failures Using IMM Estimator. *IEEE Trans. Aerospace and Electronic Systems*, AES-34(4):1293–1312, Oct. 1998.

Vita

Jifeng Ru received the B.S. and M.S. degree from Hefei University of Technology, China, in 1996 and 1999, respectively, both in Electrical Engineering. She worked as a research as well as teaching assistant during that time. Her research was mainly on intelligent control, fault diagnosis and artificial intelligence. From June, 1999 to Dec. 1999, She was a system engineer in Huawei Technologies Co. Ltd, China.

She has been a Ph.D. candidate and research assistant in the Department of Electrical Engineering at the University of New Orleans since 2001. Her current research includes signal processing, estimation and detection theory, fault detection and diagnosis, statistical inference, information fusion and target tracking. She has authored and coauthored 16 journal and conference proceedings papers.