MULTIPLE-MODEL TRACKING WITH FIXED-LAG SMOOTHING USING IMPRECISE INFORMATION

by

ZHUFANG YANG

A thesis submitted to the Faculty of Engineering
of the University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

School of Engineering (Electronic, Electrical & Computer engineering)
The University of Birmingham
September 2002

UNIVERSITY OF
BIRMINGHAM

University of Birmingham Research Archive

e-theses repository

This unpublished thesis/dissertation is copyright of the author and/or third parties. The intellectual property rights of the author or third parties in respect of this work are as defined by The Copyright Designs and Patents Act 1988 or as modified by any successor legislation.

Any use made of information contained in this thesis/dissertation must be in accordance with that legislation and must be properly acknowledged. Further distribution or reproduction in any format is prohibited without the permission of the copyright holder.

SYNOPSIS

A new multiple-model filter for target tracking has been developed and performed well in this thesis. The procedure of the new multiple model (MM) filter has no compromises between non-manoeuvre and manoeuvre, and between manoeuvres except in the ambiguous cases. The operation of the new MM filter is simple like the variable-dimension (VD) filter but does not require manoeuvre reconstruction. The new MM filter also considers all kinds of motions like interacting multiple model (IMM) filter but with a small number of models and significantly reduced computational load.

The scheme of the new multiple-model tracking filter consists of manoeuvre detection, construction of manoeuvre filters, construction of safeguard filter, and filter selection. The performance of the proposed tracking filter mainly relies on manoeuvre detection, construction of manoeuvre model filter, and construction of safeguard model filter. In order to improve the tracking, several manoeuvre detection methods have been developed. One of the manoeuvre detection methods is to test a statistic of normalised squared smoothed accelerations to give quicker manoeuvre detection. This thesis suggests that the manoeuvre be detected by testing the changes of the statistic of normalised squared innovations to give effective manoeuvre detection, based on Chen and Norton's (1986) detection by testing rapid parameter changes. The thesis also modifies Weston and Norton's (1997) change detection with the fixed-lag smoothing instead of the fixed-interval smoothing used by Weston and Norton's method, and obtains more accurate and quicker manoeuvre detection.

According to the features of target motion, the target manoeuvres are modelled as straight-line acceleration motion, cross-track acceleration motion, and curvilinear acceleration motion. Thus the manoeuvre model filters can be constructed by these three kinds of motions with a limited number of manoeuvre model filters and reduced computational load. To avoid the risk of the loss of track, a safeguard filter is used in the case of uncertain manoeuvre. The safeguard filter is constructed by combining Singer's (1970) filter and input estimation, to provide at least comparable performance to IMM filter.

Further improvement for multiple-model tracking is provided by using the fixed-lag smoothing technique. In comparison with the multiple-model filter alone, the fixed-lag smoothing multiple-model filter provides much better performance (even with fixed lag d=1), and can be implemented in a real time at the costs of a small delay and slight increase in computational load.

ACKNOWLEDGEMENTS

The author expresses gratitude to the ORS and the School of Electronic, Electrical & Computer Engineering of the University of Birmingham for their financial support.

Special thanks go to Professor John Norton of the University of Birmingham for his supervision and technical advice throughout this research.

Thanks to all the members of the Power and Control Group for their assistance, especially Professor M. Brdys, Thomas Meurers and T. Chang.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

Tracking has many military and civilian applications, ranging from undersea surveillance and space-age weapon systems to bubble-chamber experiments and image processing.

Some of the earliest examples involved radar and sonar systems, where manual tracing of "blips" on video displays by human operators evolved into computer-controlled tracking algorithms. Radar and sonar applications continue to abound. Military uses include land, air, sea, and space surveillance involving a large variety of sensors, targeting and control of individual weapons and weapon systems, and overall battle management. Civilian uses include air traffic control, collision avoidance, and navigation.

For tracking a manoeuvring target, although it has been more than 30 years since Singer published his paper on tracking a manoeuvring target (Singer, 1970), there still remains a great deal of debate surrounding this problem and more papers are published every year. Various tracking techniques have been developed over the past three decades. They are, however, scattered in the literature. So far, there is no comprehensive survey and systemic analysis for tracking. As a result, few people have a good knowledge of these techniques. This thesis is intended to fill some of the gaps by giving the survey and detailed analysis of tracking techniques to provide evidence for both practitioners and researchers in the tracking community.

No matter how many the tracking techniques, the manoeuvring target tracking techniques can mainly be divided into two categories:

1. Single model filter tracking.
2. Multiple-model filter tracking.

In the first category, the techniques use a single target motion model and different strategies are used to detect mismatch between the actual and filter motion models. Then model tuning techniques are used to make the filter motion model adaptive to the actual target motion model. The early work of Singer (1970) uses a single model filter modelling the target acceleration as a random process with known exponential autocorrelation through the whole process of tracking in the presence of the manoeuvre and in the absence of the manoeuvre. The results of this approach are that the filter is capable of tracking manoeuvring targets if the random process is properly constructed, but the quality of the estimate is degraded, compared with the Kalman filter based on a rectilinear motion model, when tracking a target moving at a constant velocity. McAulay and Denlinger (1973) performed a hypothesis test to decide when to switch between two trackers like Singer's but with different parameters. Similar early methods are developed by Hampton and Cooke (1973), and Thorp (1973). In the works of input estimation by Chan et al. (1979; 1982; 1993) and of variable-dimension filter by Bar-Shalom and Birmiwal (1982), and Park et al (1995), they use the manoeuvre detection strategies to make the filter adapt to the actual target motion. The problems in all single model filter only consider that the construction of manoeuvre model is suitable, but the assumed manoeuvre model may not adapt to the actual target motion, causing loss of the track.

The second category uses multiple-model techniques and it is assumed that the target can switch between any of the defined models. In a multiple-model bank, each model is driven by the same measurement sequence. The probability of each model being correct is evaluated using Bayes's formula. The combined state estimates are generated using the probabilities and state estimates associated with each model. Multiple model algorithms give a smooth transition, when the actual target model switches from one motion model to another, compared with the previously described technique where a single model adapts to a target model. In practice multiple model algorithms use a suboptimal Bayesian approach because the exact approach has an exponential growth with time in the number of hypotheses. The multiple-model filter of Magill (1965) runs several trackers in parallel, each with a different acceleration model. Each model is associated with a probability of currently being valid, computed from its innovation history. These probabilities weight the individual tracker estimates to produce the posterior mean state estimate. Gholson and Moose (1977) model the target manoeuvres as a semi-Markov process whereby N possible acceleration inputs are selected according to some a priori probabilities. Ricker and Williams (1978) describe a similar technique that requires N filters throughout. They compute a weighted average of the N possible inputs and use it to update the N filters. An efficient and popular formulation, the interacting multiple-model (IMM) tracker (Blom, 1984), has been used in conjunction with a nonlinear circular-turn model (Lerro and Bar-Shalom, 1993; Dufour and Mariton, 1992). More recently, researchers try to reduce the heavy computational load in IMM and develop a lot of techniques based on IMM filter, such as adaptive interacting multiple model tracking of Munir and Atherton (1994), and Layne and Piyasena (1997), a selected filter interacting multiple model (SFIMM) algorithm for tracking manoeuvring targets of Lin and Atherton (1993), a fuzzy interacting multiple

model (FIMM) algorithm of McGinnity and Irwin (1998), and so on. Their disadvantages are either heavy computation or increasing tracking error because of improper construction of IMM. Their performance is worse than a Kalman filter that assumes no manoeuvres in the absence of manoeuvre, and worse than an appropriate manoeuvring model filter during the manoeuvre.

This thesis is also intended to improve the tracking by proposing a new multiple-model tracking to reduce computational load and/or get better performance.

Optimal smoothing (retrospective state estimation) has a long history, but it receives less attention than optimal filtering because its recursive implementation is more complicated than that of filtering. However, optimal smoothing is potentially useful to help identify target type, resolve manoevure ambiguities and reconstruct tracks. Therefore, this thesis will use the fixed-lag smoothing to give further improvement for the new multiple-model tracking filter.

1.1 New Multiple-Model Tracking Filter With Fixed-Lag Smoothing

Figure 1.1 illustrates the procedure of new multiple-model tracking filter with fixed-lag smoothing. It mainly consists of the following steps:

- Manoeuvre detection
- Construction of manoeuvre filters
- Construction of safeguard filter
- Filter selection
- Fixed-lag smoothing

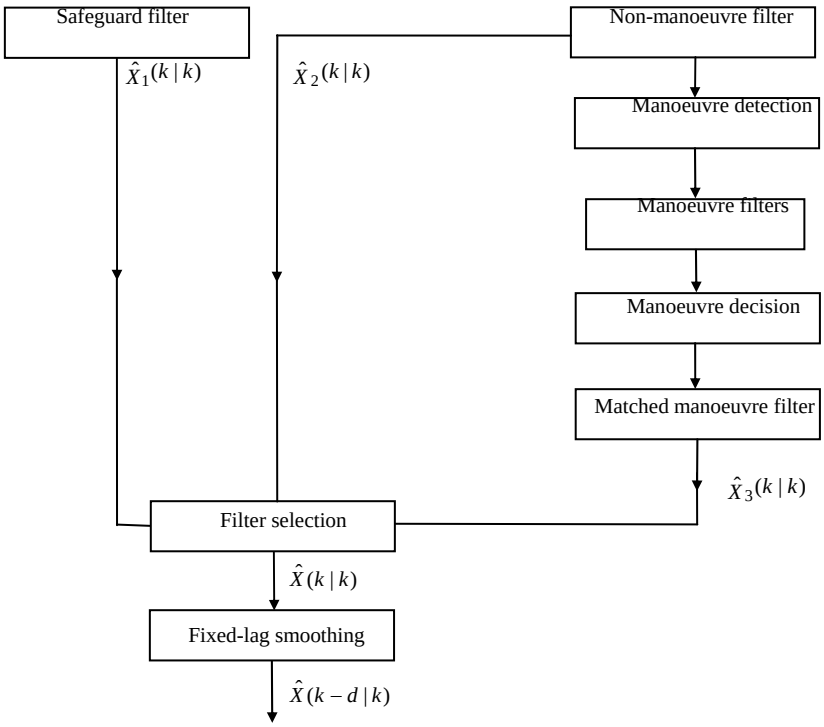


Fig. 1.1 Procedure of the new multiple-model filter with fixed-lag smoothing

For the manoeuvre detection, researchers have done a lot of works. The most classical detection is to test the statistic of normalised squared-innovation, but it suffers delay or false alarm. In last 30 years, the manoeuvre detection techniques have grown in number at high speed. McAulay and Denlinger (1973) use matched filtering through noisy innovations to detect the manoeuvre; this approach requires the specification of the manoeuvre's structure before detecting it. Bar-Shalom and Birmiwal (1982) developed manoeuvre detection by testing the statistic of normalised squared accelerations. Other manoeuvre detection techniques, such as Chan and Couture's manoeuvre detection (Chan and Couture, 1993) by comparing the projections against measurements, Zhang

and Li's manoeuvre detection (1997, 1998) by observing the probability of each model with IMM, and so on, also cannot avoid detection delay or detailed specification of manoeuvre model. This thesis will use the modified Weston and Norton's (1997) change detection to provide quick detection and find the manoeuvre steps accurately. When the manoeuvre is detected, assuming several possible manoeuvres, and the input is then estimated, also using the modified Weston and Norton's detection to distinguish the manoeuvre.

Structure of manoeuvre is often very complex, but in most of past research works, such as Bar-Shalom and Birmiwal's tracking (1982), Chan and Couture's tracking (1993), researchers suppose that manoeuvre be a simple kind of motion. In practice, the target manoeuvre has a wide variety of possible motions. Whatever how many kinds of motion there are, they can be classified into 3 kinds of motion: straight-line acceleration motion, circular motion and curvilinear acceleration motion. According to the 3 kinds of motion, the multiple-model filter contains the 3 kinds of motion model. Thus, after the manoeuvre is detected, the three model filters are run in parallel to estimate the correspond manoeuvre, and meanwhile the manoeuvre detection is processing along with each filter. The models will be discussed in Chapter 3. During distinguishing the manoeuvre, the tracking is guaranteed by a safeguard filter. Once the manoeuvre is certain and the manoeuvre step is found, the tracking is provided by the model best matched to the manoeuvre.

Construction of the safeguard filter is discussed in Chapter 5, filter selection in Chapter 6, and fixed-lag smoothing in Chapter 7. Here they are not reiterated. This thesis only considers the single plane tracking problem.

1.2 Overview Of The Thesis

This work mainly consists of three parts, respectively covering target motion models, manoeuvre detection and tracking algorithms. The first, Chapter 2, reviews tracking algorithms in the literature, analyses the existing problems in each tracking algorithm, and finally proposes an effective new multiple-model filter for tracking the manoeuvring targets.

Chapter 3 focuses on target motion models. This chapter introduces the motion models used in both the tracking and interception of targets, provides detailed and systematic models in Cartesian co-ordinates from the simplest constant-velocity motion to the more complex constant-rectilinear-acceleration motion via kinematics. This chapter also presents Best and Norton’s (1997) kinematics model which is of benefit to the tracking of curvilinear motion targets. However, Best and Norton's kinematics model is only suitable for manoeuvring target tracking supposing knowledge of along- and/or cross-track accelerations, thus this chapter modifies Best and Norton's kinematics model with an augmented state model, and broadens its applications.

Chapter 4 reviews manoeuvre detection techniques in the literature, and develops several new manoeuvre detection techniques to provide quicker detection and to improve the tracking for manoeuvring target.

Chapters 5 and 6 concentrates on the proposed multiple-model filter. Chapter 5 derives modified Singer's filter by combining Singer's filter and input estimation, and illustrates why the modified Singer's filter can be a part of the proposed multiple-model filter and be used to provide the safeguards for the proposed multiple-model filter tracking. Chapter 6 details the construction of the proposed multiple-model filter, describes how

the multiple-model filter works, and gives simulations to show how effective and economical the proposed multiple-model filter is.

Chapter 7 gives the further improvement for the proposed multiple-model tracking filter by using the fixed-lag smoothing technique. The simulations show that the fixed-lag smoothing multiple-model filter provides much better performance and can be implemented in real time at the costs of a small delay and slight increase in computational load, compared to the proposed multiple-model filter alone.

Finally, Chapter 8 draws conclusions and discusses further work.

CHAPTER 2

PROBLEMS IN TRACKING WITH IMPRECISE INFORMATION

Tracking Manoeuvring targets is a subject which has been growing for many decades and a lot of tracking techniques have been published in the literature over that time. The simplest form of tracker is the α-β tracker which has a fixed correction gain. A more popular, but computationally slower solution, is the Kalman filter with its time-varying gain. This chapter reviews the existing tracking techniques based on Kalman filter, gives a detailed analysis of the existing tracking methods and introduces a new method, which has a modest computational load and good performance for tracking manoeuvring targets.

2.1 Introduction

Since 1960, when the notable Kalman filter was introduced to the world, a lot of applications based on Kalman filter have been explored for tracking manoeuvring targets. The early work of Singer (1970) is an application of the Kalman filter with the target acceleration equation represented by a first-order autoregressive process. This Singer filter tracks a manoeuvring target well but its performance degrades, as compared with a "simple" Kalman filter that assumes no manoeuvres, during constant-velocity, straight-line motions. McAulay and Denlinger (1973), by using a manoeuvre detector, realize the benefits of both filters by tracking normally with the simple Kalman filter and using the augmented filter only when a manoeuvre is detected.

Thorp (1973) models a manoeuvre as an increase in the driving noise, assumed to be a white Gaussian sequence. Thorp's method is effective because the larger covariance

means the tracker puts less emphasis on the model. However, an arbitrary increase in process noise covariance is not a particularly structured approach to maintaining the track. Bekir (1983) presented a method for making the increase in process noise covariance dependent on an estimate of the acceleration magnitude. Also, any increase in process noise covariance causes large sensitivity to observation noise, degrading the accuracy during steady conditions.

Moose et al. (1975; 1977; 1979) model the target manoeuvres as a semi-Markov process whereby N possible acceleration inputs are selected according to some a priori probabilities. In its pure form, this technique initially requires a bank of N Kalman filters, one for each particular input from the set, then N^2 and N^3 for the second and third measurements, and so forth. This complication is necessary because the N acceleration possibilities can also branch to N other possible target manoeuvres at the next measurement. Hence, the optimal solution gives rise to an exponentially increasing number of Kalman filters as the number of possible branches increases with time. Hence, for practical purposes, suboptimal multiple model approaches which maintain a constant number of Kalman filters have been developed. In Moose et al. (1975; 1977; 1979) the manoeuvre accelerations were assumed to be limited to a time invariant set of discrete values and switched values according to a semi-Markov process. Probabilities of the different accelerations having occurred were computed and incorporated into the state estimator. Ricker and Williams (1978) describe a similar technique that requires N filters throughout. They compute a weighted average of the N possible inputs and use it to update the N filters. These N filters are basis for the computationally expensive multiple-model trackers described later.

In the above filters (Singer, 1970; McAulay and Denlinger, 1973; Moose et al., 1975; 1977; 1979; Ricker and Williams, 1978), the manoeuvre characteristics of the target must be known in order that they be modelled either as a first-order autoregressive, or semi-Markov process. When the actual target does not correspond to the model, some deterioration in tracking accuracy is bound to result. Chen et al. (1979; 1982) suggest a input estimator. This tracking scheme, based on the Kalman filter, estimates the acceleration inputs from the residuals and uses the estimates to correct the Kalman filter. The estimation process is independent of target types and a model of the manoeuvre characteristics is not needed. The aim of updating the filter output is simply to remove the filter bias caused by the target deviating from the assumed constant-velocity, straight-line motion. As is well known (Chan et al, 1979; Bar-Shalom and Birmiwal, 1982), bias removal is accompanied by an increase in the estimation variance. Because typical targets spend considerable periods of time on straight-line, constant-velocity tracks, the Kalman filter should not be unnecessarily updated during those periods, else the error variance of the filter will increase in the non-manoeuvre situation. A detector that checks the magnitude of the estimated inputs is used to reduce the frequency of false alarms. Updating is performed only if there is some certainty that the target has manoeuvred. The original input estimation (IE) algorithm presented in (Chan et al., 1979; 1982) consists of the estimation of the unknown input (manoeuvre) over a sliding window and suitable compensation of the state estimate under the assumption that the manoeuvre onset time is the starting point of the sliding window. And also this technique shows poor performance in case of a gentle manoeuve because it tends to over-compensate for the manoeuvre.

To cope with this problem above in Chan et al's method, Bogler (1987) proposed an adaptive filter with a revised input estimation technique which estimates the onset time

of a manoeuvre as well as its magnitude. However, Bogler's technique requires too a long window to be applied to a gently manoeuvring target because it takes a long time to accumulate the effects of the manoeuvre. The long window causes a complex filter structure and an increase of computing time as well. The technique of (Bogler, 1987), while superior to the original IE algorithm (Chan et al, 1979; 1982), requires a large number of filters (together with input estimators and state estimate correctors) to run in parallel (20 such filters were used in the simulations presented in (Bogler, 1987)), requiring a significant amount of computation and memory.

In 1982, Bar-Shalom and Birmiwal (Bar-Shalom and Birmiwal, 1982) proposed a variable-dimension filter. In the algorithm, the tracking filter operates in its normal mode in the absence of any manoeuvres. A detection scheme has been developed to determine that a manoeuvre is indeed occurring. Once a manoeuvre is detected a different state model is used by the filter: new state components are added. The extent of the manoeuvre as detected is then used to yield an estimate for the extra state components, and corrections are made on the other state components. The tracking is then done with the augmented state model util it will be reverted to the normal model by another decision. The switching of models resembles somewhat the approach of (McAulay and Denlinger, 1973) but in Bar-Shalom and Birmiwal work (1982) it is done between models of different complexity. The rationale for using a lower order quiescent model and a higher order manoeuvring model is that it will allow good tracking performance in both situations rather than a compromise.

In 1984, the interacting multiple model (IMM) algorithm was developed by Blom (Blom, 1984) as a very useful algorithm for the tracking of a manoeuvring target. The IMM algorithm is a suboptimal hybrid filter that has been shown to be one of the most

cost-effective hybrid state estimation schemes. The main feature of this algorithm is its ability to estimate the state of a dynamic system with several behaviour modes which can switch from one to another. In particular, the IMM filter can be a self-adjusting variable-bandwidth filter, which makes it natural for tracking manoeuvring targets. It has been successfully applied to a number of problems (Averbuch, Itzikowitz and Kapon, 1991, Bar-Shalom, 1990 and 1992, Bar-Shalom, Chang, and Blom, 1990, Guu and Wei, 1991, Houles and Bar-Shalom, 1989, Li and Bar-Shalom, 1992) and is being implemented in some air traffic control systems (Blom et al, 1992, Dufour and Mariton, 1992,Vacher et al, 1992), especially in manoeuvring target tracking and automatic track formation (Bar-Shalom et al, 1989, 1990, 1992, 1993). However, the IMM filter is constructed by a large number of filters to cover the possible manoeuvre motions. This means a number of filters have to be used, resulting in a heavy computation. The performance of the IMM filter also degrades, as compared with a standard Kalman filter in the absence of manoeuvre and the correct manoeuvre filter in the presence of manoeuvre.

To reduce the computation load of IMM, several adaptive IMM filters have been developed. A variable-dimension IMM (VDIMM) has been suggested by Bar-Shalom (Bar-Shalom, 1989). The VDIMM filter consists of a second-order model for the quiescent mode of the target and one or two third-order models with different process noise levels for the manoeuvre mode. The VDIMM filter is efficient in tracking manoeuvring targets. But this requires a prior suitable choice of process noise covariance of the manoeuvre model according to the manoeuvre input level, and a prior choice of mode transition probability, which may be difficult to supply. Poor choices may degrade its performance.

Lin and Atherton (Lin and Atherton, 1993) suggest the SFIMM algorithm, which uses only a subset of the filters in each scan to reduce the computation time, where the subset is selected according to some decision rule. The tracking accuracy of the SFIMM is slightly worse than that of an IMM algorithm.

In 1994, Munir and Atherton (1994) proposed adaptive IMM (AIMM) approach. In this research, The AIMM algorithm estimates the acceleration, by means of a biasd filter, to send the result to a bank of IMM filters. They then allow the entire bank of filters to move in acceleration space such that the bank is centered on the acceleration estimate. By doing this fewer models are required to cover the acceleration space.

The Interacting Acceleration Compensation (IAC) algorithm (Watson and Blair, 1995) is proposed to reduce the computation cost of the IMM algorithm while maintaining similar performance. The IAC algorithm incorporates the concept of the IMM algorithm for two motion models into the framework of a two-stage estimator (Alonani, Xia, Rice and Blair, 1991). The IAC algorithm is viewed as a two-stage estimator having two acceleration models: the zero acceleration of the constant velocity model and a nearly constant acceleration model. The IMM algorithm is used to compute an acceleration estimate for compensating the estimate of the constant velocity filter.

Another adaptive multiple model estimator is the so-called moving-bank multiple model adaptive estimator (MBMMAE) presented in (Gustafson and Maybeck, 1992, Maybeck and Hentz, 1987) . Because the bank of filters is fixed and does not move, perhaps a more appropriate name might be the moving-window multiple model adaptive estimator. This technique is not based on the IMM; however, the basic idea is easily implemented in the IMM framework. Here all of the models are assumed to be

fixed. However, to reduce computational complexity the MBMMAE sets a window around a subset of the filter bank which is centered on the estimated acceleration. Only the filters that fall within the window propagate and update their estimates. Everything falling outside the window is turned off.

In 1997, Layne and Piyasena (1997) proposed another AIMM approach. First, they reduce the number of filters in the bank so that the acceleration space is covered at coarser levels. Then they add a single adaptive acceleration model. The adaptive acceleration model is designed to capture the target dynamics when its behaviour falls between the fixed models. Their new AIMM outperforms both the classical IMM and the other adaptive IMMs with reduced computational complexity.

In 1998, McGinnity and Irwin (1998) suggested a new fuzzy interacting multiple model (FIMM) algorithm. This considers each Kalman filter to have only local validity, defined by the model conditioning manoeuvre input. Fuzzy sets are used to determine this validity as a similarity measure between an estimate of the acceleration and the conditioning value. By using an appropriate fuzzy set overlap, only a subset of the total number of models needs to be evaluated, and these will be conditioned on acceleration values close to the estimate. This reduces the computational load compared to the IMM algorithm, in which the complete set of models are always evaluated, even if the likelihood of a particular model is minimal. As the number of models evaluated is determined only by the overlap, more models can be added without increasing the computational load significantly, and therefore adequate coverage of the range of manoeuvre variables can always be achieved.

However, all these methods above of adaptive IMM filter construction rely on the accuracy of estimated acceleration. If the estimated acceleration is not good enough, it would cause these adaptive IMM filters to fail for tracking. On other hand, if we can get good estimated accelerations, we do not have to use the expensive IMM filter for tracking.

Thus, we propose a new MM filter to decrease the computational load and to obtain a good performance. The procedure of the new MM filter is that Singer's filter is modified to fit any level of manoeuvre; in the absence of manoeuvre, the unaugmented state model is used for tracking; when the manoeuvre is detected, several kinds of manoeuvre models are assumed, we run these manoeuvre filters in parallel to make sure which manoeuvre occurs, and then tracking filter is switched into the corresponding manoeuvre filter; during ambiguous, the modified Singer's filter is used for tracking. The procedure of the MM filter does not use probabilistic weighting as in IMM.

The following sections will give a detailed analysis of several typical tracking techniques: Singer's filter, input estimation tracking, variable-dimension filter tracking, MM filter tracking, IMM filter tracking, and modified Kalman filter tracking. Section 2.2 describes Singer's filter, Section 2.3 reviews and analyzes input estimation tracking, Section 2.4 gives a review and analysis of variable-dimension tracking filter, Section 2.5 describes MM tracking filter, Section 2.6 reviews IMM tracking filter, Section 2.7 describes modified Kalman filter tracking, Section 2.8 presents a new MM tracking filter, and Section 2.9 gives a summary of this chapter.

2.2 Problems In Singer's Tracking Filter

Approaches based on Kalman filtering include the early work of Singer (1970), who augmented the target motion model with the target acceleration, represented as a first-order autoregressive process.

Singer proposed a manoeuvre model as a first-order Markov process with zero-mean as follows,

$$\dot{a}(t) = -\alpha\, a(t) + w(t) \tag{2.1}$$

where $a(t)$ is an acceleration and $w(t)$ is a zero-mean white noise with variance $E\{w^2(\tau)\} = q\delta(t-\tau)$. Process noise variance q is given by $q = 2\alpha\sigma_m^2$ where σ_m^2 is the variance of target manoeuvre acceleration.

The equations of x-direction and y-direction are similar, so here we only give the model in x-direction. Defining a state vector $X(t)$ of position, velocity, and acceleration in x-direction, the target dynamics model in x-direction can be written by

$$\dot{X}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\alpha \end{bmatrix} X(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w(t) \tag{2.2}$$

The discrete form of the above equation is sought for digital implementation. Many sensors have a constant data rate, sampling target position every T seconds. The appropriate (discrete time) target equations of motions for this application are given by

$$X(k+1) = \begin{bmatrix} 1 & T & \frac{1}{\alpha^2}[-1+\alpha T + e^{-\alpha T}] \\ 0 & 1 & \frac{1}{\alpha}[1-e^{-\alpha T}] \\ 0 & 0 & e^{-\alpha T} \end{bmatrix} X(k) + U(k) \tag{2.3}$$

where $U(k)$ is noise term.

The covariance of $U(k)$ is

$$Q(k) = E[U(k)U^{\mathrm{T}}(k)] = 2\alpha\sigma_m^2 \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \tag{2.4}$$

where

$$q_{11} = \frac{1}{2\alpha^5}[1 - e^{-2\alpha T} + 2\alpha T + \frac{2\alpha^3 T^3}{3} - 2\alpha^2 T^2 - 4\alpha T e^{-\alpha T}]$$

$$q_{12} = \frac{1}{2\alpha^4}[e^{-2\alpha T} + 1 - 2e^{-\alpha T} + 2\alpha T e^{-\alpha T} - 2\alpha T + \alpha^2 T^2]$$

$$q_{13} = \frac{1}{2\alpha^3}[1 - e^{-2\alpha T} - 2\alpha T e^{-\alpha T}]$$

$$q_{22} = \frac{1}{2\alpha^3}[4e^{-\alpha T} - 3 - e^{-2\alpha T} + 2\alpha T]$$

$$q_{23} = \frac{1}{2\alpha^2}[e^{-2\alpha T} + 1 - 2e^{-\alpha T}]$$

$$q_{33} = \frac{1}{2\alpha}[1 - e^{-2\alpha T}]$$

Derivation of $Q(k)$ in detail can be found in Appendix I.

Since $w(t)$ is white noise, $E[U(k)U^{\mathrm{T}}(k+i)] = 0$ for $i \neq 0$ so that $U(k)$ is a discrete time white noise sequence. The state equations just derived are therefore directly suitable for Kalman filter applications.

The tracking sensor measures target along the dimension being transformed to Cartesian coordinates and provides

$$Z(k) = HX(k) + V(k) \tag{2.5}$$

where $H = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$

and $V(k)$ is white noise with variance σ_R^2.

Equations (2.3), (2.4), and (2.5) have the form for which the optimal linear filter is the Kalman filter.

The augmented filter shows good performance for the target with a low-acceleration manoeuvre, but its performance rapidly degrades in the case of high-acceleration manoeuvre and is worse than that with a Kalman filter that assumes no manoeuvre during constant-velocity, straight-line motion. In the case of high-acceleration manoeuvre, the process noise variance has to be chosen big enough to cover the acceleration change so that Singer's filter would respond quick to the manoeuver at the step at which the manoeuvre happens. However, after the manoeuvre occurs, the acceleration may become constant, i.e., the process noise variance should be zero, and the mean of the process noise should be the amplitude of acceleration, excluding the system noise. Thus, the big process noise introduced will increase the estimate covariance, compared with input estimate or augmented variable-dimension filter assuming that the acceleration is a constant during the manoeuvre. During constant-velocity, straight-line motion, the performance of Singer's filter with the big process noise is also significantly worse than that of a Kalman filter that assumes no manoeuvre.

A common method is to use a non-manoeuvre target model for tracking a target moving at a constant velocity and then switch to an appropriate manoeuvre model, when a target manoeuvre is detected. The following variable-dimension (VD) filter will describe this method.

2.3 Problems In Input Estimation Tracking

This method is to construct a linear measurement of the input (manoeuvre) U, in the presence of the additive white noise from the innovation of the non-manoeuvre filter and to get input estimate using the least-squares criterion, then to correct the state estimate with the estimated input.

Consider a linear stochastic system of the form

$$X(k+1) = FX(k) + GU(k) + W(k) \tag{2.6}$$

$$Z(k) = HX(k) + V(k) \tag{2.7}$$

where $W(k) = \begin{bmatrix} w_1(k) & w_2(k) \end{bmatrix}^{\mathrm{T}}$

$E[W(k)] = 0$, $E[W(i)W^{\mathrm{T}}(j)] = Q\delta_{ij}$

$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$, $E[V(k)] = 0$, $E[V(i)V^{\mathrm{T}}(j)] = R\delta_{ij}$

Estimation of the state is done using the model without input (non-manoeuvre model):

$$X(k+1) = FX(k) + W(k) \tag{2.8}$$

where the state is

$X = \begin{bmatrix} x & y & \dot{x} & \dot{y} \end{bmatrix}^{\mathrm{T}}$

From the innovation of the Kalman filter based on the non-manoeuvre model (2.8), the input $U(k)$ is to be detected, estimated and used to correct the state estimate.

Assume that the target starts manoeuvring at time *k*. Its unknown inputs during the time interval [*k*,…, *k*+*s*] are $U(i)$, $i = k,...,k+s-1$. The state estimates from the (now mismatched) filter based on (2.8) will be denoted by an asterisk.

$$\begin{aligned} \hat{X}^*(i+1 \mid i) &= F[I - K(i)H]\hat{X}^*(i \mid i-1) + FK(i)Z(i) \\ &\equiv \Phi(i)\hat{X}^*(i \mid i-1) + FK(i)Z(i) \end{aligned} \tag{2.9}$$

where $i = k,...,k+s-1$

with the initial conditions

$$\hat{X}^*(k \mid k-1) = \hat{X}(k \mid k-1) \tag{2.10}$$

being the correct estimate and prediction before the manoeuvre started.

Recursion (2.9) yields, in terms of the initial conditions (2.10)

$$\hat{X}^*(i+1 \mid i) = [\prod_{j=k}^{i} \Phi(j)]\hat{X}(k \mid k-1) + \sum_{j=k}^{i} [\prod_{m=j+1}^{i} \Phi(m)]FK(j)Z(j) \tag{2.11}$$

where $i = k,...,k+s-1$

If the inputs were known, the correct filter based on (2.6) would yield estimates according to the recursion

$$\hat{X}(i+1 \mid i) = [\prod_{j=k}^{i} \Phi(j)]\hat{X}(k \mid k-1) + \sum_{j=k}^{i} [\prod_{m=j+1}^{i} \Phi(m)][FK(j)Z(j) + GU(j)] \tag{2.12}$$

where $i = k,...,k+s-1$

which is the same as (2.12) except for the last term containing the inputs.

Comparing equation (2.11) with (2.12), if the estimate $\hat{U}(j)$ is got , then the state estimate $\hat{X}(i+1 \mid i)$ can be got. So, the problem is how to get $\hat{U}(j)$.

The innovations of equation (2.6) are

$$v(i+1) \equiv Z(i+1) - H\hat{X}(i+1 \mid i) \tag{2.13}$$

The innovations of equation (2.8) are

$$v^*(i+1) \equiv Z(i+1) - H\,\hat{X}^*(i+1 \mid i) \tag{2.14}$$

From equations (2.13) and (2.14), we can get

$$v^*(i+1) = v(i+1) + H\sum_{j=k}^{i} [\prod_{m=j+1}^{i} \Phi(m)]GU(j) \tag{2.15}$$

Assuming the input to be constant over the time interval [k,…, k+s-1], i.e.,

$$U(j) = U, \qquad j = k,...,k+s-1 \tag{2.16}$$

and yields

$$v^*(i+1) = \Psi(i+1)U + v(i+1) \tag{2.17}$$

where $\Psi(i+1) \equiv H \sum_{j=k}^{i} [\prod_{m=j+1}^{i} \Phi(m)]G$ (2.18)

Equations (2.17) shows that the innovation $v^*(i+1)$ of the non-manoeuvre filter is a "linear measurement" of the input (manoeuvre) U in the presence of the additive "white noise" $v(i+1)$. From equation (2.17) it follows that the input can be estimated, using the generalised least-squares criterion from

$$Y = \Psi U + \varepsilon \tag{2.19}$$

where $Y \equiv \begin{bmatrix} v^*(k+1) \\ \vdots \\ v^*(k+s) \end{bmatrix}$, $\Psi \equiv \begin{bmatrix} \Psi(k+1) \\ \vdots \\ \Psi(k+s) \end{bmatrix}$ and $\varepsilon \equiv \begin{bmatrix} v(k+1) \\ \vdots \\ v(\mathrm{k+s}) \end{bmatrix}$ is zero-mean with block-diagonal covariance matrix

$$S \equiv diag[S(i)] \tag{2.20}$$

where $S(i) = E[v(i)v^{\mathrm{T}}(i)] = E\{[Z(i) - H\hat{X}(k \mid k-1)][Z(i) - H\hat{X}(k|k-1)]^{\mathrm{T}}\}$

$= E\{[HX(i) + V(i) - H\hat{X}(k \mid k-1)][HX(i) + V(i) - H\hat{X}(k|k-1)]^{\mathrm{T}}\} = HP(i \mid i-1)H^{\mathrm{T}} + R(i)$

$R(i)$ is the covariance matrix of measurement noise.

The estimation can be done in batch form as

$$\hat{U} = (\Psi^{\mathrm{T}} S^{-1} \Psi)^{-1} \Psi^{\mathrm{T}} S^{-1} Y \tag{2.21}$$

After the input has been estimated, the state has to be corrected as follows:

$$\hat{X}^U(k+s+1 \mid k+s) = \hat{X}^*(k+s+1 \mid k+s) + M\hat{U} \tag{2.22}$$

where $M \equiv \sum_{j=k}^{k+s} [\prod_{m=j+1}^{k+s} \Phi(m)]G$

The covariance associated with the estimate (2.22) is

$$P^U(k+s+1\,|\,k+s) = P(k+s+1\,|\,k+s) + ML\,M^{\mathrm{T}} \tag{2.23}$$

where $L = (\Psi^{\mathrm{T}} S^{-1} \Psi)^{-1}$.

From the procedure of input estimation, it is not difficult to see that the accuracy of the estimated input relies on the signal-to-noise ratio. If the manoeuvre is low level, the IE would show poor performance because it tends to over-compensate the manoeuvre. Furthermore, the IE is based on the assumption that the manoeuvre onset time is the starting point of the sliding window. While this assumption is not explicitly stated, the performance of IE algorithm will be degraded.

2.4 Problems In Variable-Dimension Tracking Filter

The variable-dimension filter adds extra state variables once a manoeuvre is detected. The extent of the manoeuvre as detected is then used to estimate the extra state components, and corrections are made to the other state variables. Tracking employs the augmented state model until it reverts to the normal model on detection of the end of the manoeuvre.

In the absence of manoeuvre, the target motion is modelled as constant-velocity motion in a plane, subject to variations in velocity induced by piecewise constant zero-mean accelerations with white sample-instant values. The corresponding state equation is

$$X(k+1) = FX(k) + GW(k) \tag{2.24}$$

where, using Cartesian coordinates, the state is

$$X = \begin{bmatrix} x & y & \dot{x} & \dot{y} \end{bmatrix}^{\mathrm{T}}$$

$$F = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad W(k) = \begin{bmatrix} w_1(k) & w_2(k) \end{bmatrix}^{\mathrm{T}}$$

$$G = \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ \mathrm{T} & 0 \\ 0 & \mathrm{T} \end{bmatrix}, \quad E[W(k)] = 0, \quad E[W(i)W^{\mathrm{T}}(j)] = Q\delta_{ij}$$

and T is the sampling interval. The initial state estimate is $\hat{X}(0 \mid 0)$ with covariance $P(0 \mid 0)$. Position measurements (transformed to Cartesian coordinates) are made, so

$$Z(k) = HX(k) + V(k) \tag{2.25}$$

where

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad E[V(k)] = 0, \quad E[V(i)V^{\mathrm{T}}(j)] = R\delta_{ij}$$

In the manoeuvre model, the augmented state is

$$X^m = \begin{bmatrix} x & y & \dot{x} & \dot{y} & \ddot{x} & \ddot{y} \end{bmatrix}^{\mathrm{T}}$$

and the state equation is

$$X^m(k+1) = F^m X^m(k) + G^m W^m(k) \tag{2.26}$$

where

$$F^m = \begin{bmatrix} 1 & 0 & T & 0 & T^2/2 & 0 \\ 0 & 1 & 0 & T & 0 & T^2/2 \\ 0 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \qquad G^m = \begin{bmatrix} T^2/4 & 0 \\ 0 & T^2/4 \\ T/2 & 0 \\ 0 & T/2 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$W^m = \begin{bmatrix} w_{m1} & w_{m2} \end{bmatrix}^{\mathrm{T}}, \quad E[W^m(k)] = 0, \quad E\{W^m(i)[W^m(j)]^{\mathrm{T}}\} = Q^m\delta_{ij}$$

The corresponding observation equation is

$$Z(k) = H^m X(k) + V(k) \tag{2.27}$$

where

$$H^m = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

When the manoeuvre is detected, the filter is reinitialised by retreating to the state prior to the detection window, $\hat{X}(k-\Delta-1 \mid k-\Delta-1)$, and updating the state to the current time using all measurements within the window and the manoeuvre model. An estimate of target acceleration is required to begin the reinitialisation process and this is found from the measurement at $k-\Delta$.

$$\begin{bmatrix} \hat{\ddot{x}}(k-\Delta \mid k-\Delta) \\ \hat{\ddot{y}}(k-\Delta \mid k-\Delta) \end{bmatrix} = \frac{2}{T^2}[Z(k-\Delta) - H\hat{X}(k-\Delta \mid k-\Delta-1)] \tag{2.28}$$

where $\hat{X}(k-\Delta \mid k-\Delta-1)$ comes from the constant-velocity tracker.

With the acceleration found, the process noise covariance Q^m is selected according to the estimated acceleration; the process noise standard deviation is usually taken as 5% of the estimated acceleration (Bar-Shalom and Fortmann, 1988); and the state estimate for $k-\Delta$ is re-calculated

$$\hat{x}^m(k-\Delta \mid k-\Delta) = z_1(k-\Delta) = x(k-\Delta) + V_x(k-\Delta) \tag{2.29}$$

$$\hat{\dot{x}}^m(k-\Delta \mid k-\Delta) = \hat{\dot{x}}(k-\Delta-1 \mid k-\Delta-1) + T\hat{\ddot{x}}(k-\Delta-1 \mid k-\Delta-1) \tag{2.30}$$

$$\hat{\ddot{x}}^m(k-\Delta \mid k-\Delta) = \frac{2}{T^2}[z_1(k-\Delta) - \hat{x}(k-\Delta-1 \mid k-\Delta-1) - T\hat{\dot{x}}(k-\Delta-1 \mid k-\Delta-1)] \tag{2.31}$$

The elements of covariance matrix of estimate error are

$$P_{11}^m(k-\Delta \mid k-\Delta) = R_{11} \tag{2.32}$$

$$P_{55}^m(k-\Delta \mid k-\Delta) = E[\tilde{\ddot{x}}^m(k-\Delta \mid k-\Delta)]^2$$

$$= \frac{4}{T^4}[R_{11} + P_{11}(k-\Delta \mid k-\Delta) + 2P_{13}(k-\Delta \mid k-\Delta)T + P_{33}(k-\Delta \mid k-\Delta)T^2] \tag{2.33}$$

$$P_{33}^m(k-\Delta \mid k-\Delta) = E[\tilde{\dot{x}}^m(k-\Delta|k-\Delta)]^2$$
$$= \frac{4}{T^2}R_{11} + \frac{4}{T^2}P_{11}(k-\Delta \mid k-\Delta) + \frac{2}{T}P_{13}(k-\Delta \mid k-\Delta) + \frac{2}{T^2}P_{33}(l-\Delta \mid k-\Delta) \tag{2.34}$$

$$P_{13}^m(k-\Delta \mid k-\Delta) = E[\tilde{x}^m(k-\Delta \mid k-\Delta)\tilde{\dot{x}}(k-\Delta \mid k-\Delta)]$$
$$= \frac{2}{T}R_{11} \tag{2.35}$$

$$P_{15}^m(k-\Delta \mid k-\Delta) = E[\tilde{x}^m(k-\Delta \mid k-\Delta)\tilde{\ddot{x}}^m(k-\Delta \mid k-\Delta)]$$
$$= \frac{2}{T^2}R_{11} \tag{2.36}$$

$$P_{35}^m(k-\Delta \mid k-\Delta) = E[\tilde{\dot{x}}^m(k-\Delta \mid k-\Delta)\tilde{\ddot{x}}^m(k-\Delta \mid k-\Delta)$$
$$= \frac{4}{T^3}R_{11} + \frac{4}{T^3}P_{11}(k-\Delta \mid k-\Delta) + \frac{2}{T}P_{33}(k-\Delta \mid k-\Delta) + \frac{6}{T^2}P_{13}(k-\Delta \mid k-\Delta) \tag{2.37}$$

Similarly, the estimate and covariance associated with *y* direction can be obtained.

From the procedure of the VD algorithm, the VD filter presumes that the target starts to manoeuvre at the starting point of a sliding window running back from when the manoeuvre is detected, but the actual and assumed manoeuvres may differ in timing, leading to large tracking errors. The VD filter has to reconstruct the process noise covariance and the state estimates within the sliding window when changing to the manoeuvre model. The filter uses only measurements at the start of the sliding window to initialise the augmented filter. This also may increase the tracking error. Furthermore, the acceleration estimates must approach zero before the filter model can be switched from constant acceleration to constant velocity, However, if the constructed

process noise covariance is too big, causing prior to non-manoeuvre detected, the tracking may pose a significant problem.

2.5 Problems In MM Tracking Filter

The common version of MM filter (Bar-Shalom and Fortmann, 1988) can be shown in Fig. 2.1.

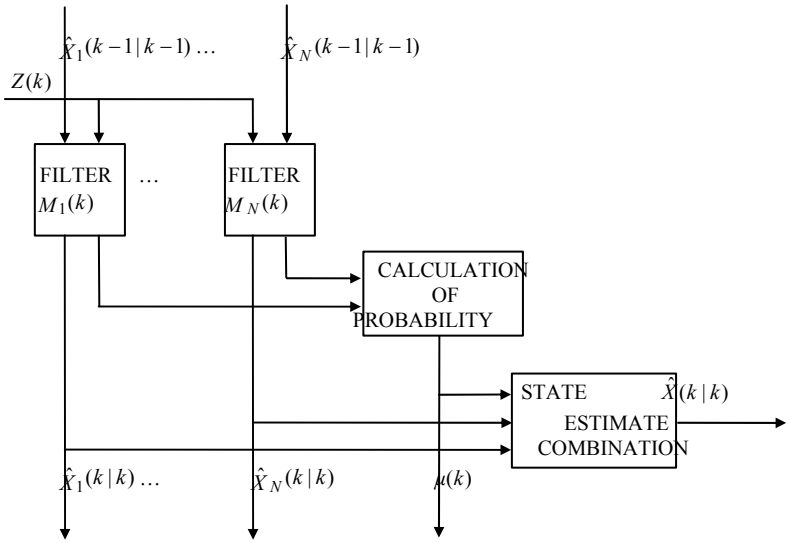


Fig. 2.1 MM algorithm

Let M_j be the event that model j is correct with prior probability

$$\mathrm{P}\{M_j\}=\mu_j(0),\ j=1,...,N$$

The likelihood function of the measurements up to time k under the assumption of model j is

$$\lambda_j(k)=p(Z^k\mid M_j)=\prod_{i=1}^{k}p[v_j(i)]$$

where, under the Gaussian assumption, the PDF of the innovation from filter j is

$$p[v_j(k)] = \frac{1}{\sqrt{|2\pi S_j(k)|}} e^{-0.5 v_j^{\mathrm{T}}(k) S_j^{-1}(k) v_j(k)}$$

Using Bayes' rule, the posterior probability that model j is correct at time k is

$$\mu_j(k) \equiv \mathrm{P}\{M_j \mid Z^k\} = \frac{p[v_j(k)]\mu_j(k-1)}{\sum_{l=1}^{N} p[v_l(k)]\mu_l(k-1)}$$

The state estimate is a weighted average of the model-conditioned estimates with the above probabilities as weights,

$$\hat{X}(k \mid k) = \sum_{i=1}^{N} \mu_i(k)\, \hat{X}_i(k \mid k) \qquad (2.38)$$

The covariance of the resulting combined estimate is:

$$P(k \mid k) = \sum_{i=1}^{N} \mu_i(k)[P_i(k \mid k) + (\hat{X}_i(k \mid k) - \hat{X}(k \mid k))(\hat{X}_i(k|k) - \hat{X}(k|k))^{\mathrm{T}}] \qquad (2.39)$$

The above derivation is exact under the following assumptions:

1) The correct model is among the set of N models considered.

2) The same model has been in effect from the initial time.

From the procedure of MM filter, to achieve a good performance for MM filter, the MM filter should consist of a huge number of filters to cover possible manoeuvres, causing heavy computation. This approach is based on the "non-interacting" MM method: the single-model-based filters are running in parallel without mutual interaction, i.e., each filter operates independently at all times. Such an approach is quite effective in handling problems with an unknown structure or parameter but without structural or parametric changes. If the system structure or parameter changes, the MM filter could fail for tracking unless the models are updated.

A famous recent advance in MM estimation is the development of the IMM estimator. It overcomes the above-mentioned weakness of the non-interacting MM approach by

explicitly modelling the abrupt changes of the system by "switching" from one model to another in a probabilistic manner. The following Section 2.6 will give its detailed description.

2.6 Problems In IMM Tracking Filter

The IMM algorithm is a modern approach to merging the different model hypotheses (Blom, 1984). A linear system with Markovian switching coefficients can be represented as

$$\begin{aligned} X(k) &= F[k-1, M(k)]X(k-1) + G[k-1, M(k)]W(k-1, M(k)) \\ Z(k) &= H[k, M(k)]X(k) + V(k, M(k)) \end{aligned} \tag{2.40}$$

where $M(k)$ is a finite state Markov chain taking values in {1, 2, …, N} according to the probability P_{ij} of transitioning from model i to model j,

and $M(k) \in \{M_j(k)\}_{j=1}^{N}$

The structure of the system and/or the statistics of the noises can differ from mode to mode:

$$F[k-1, M_j(k)] = F_j(k-1) \qquad G[k-1, M_j(k)] = G_j(k-1) \qquad H[k, M_j(k)] = H_j(k)$$

$$W(k-1, M_j(k)) \sim N(\overline{W}_j(k-1), Q_j(k-1))$$

$$V(k, M_j(k)) \sim N(0, R_j(k))$$

The IMM algorithm consists of a filter for each model, a model probability evaluator, an estimate mixer at the input of the filters, and an estimate combiner at the output of the filters. The multiple models interact through the mixing to track a manoeuvring target. Assuming the model switching is governed by an underlying Markov chain, the mixer uses the model probabilities and the model switching probabilities to compute a mixed estimate for each filter. Each filter then uses a mixed estimate and the

measurement to compute a new estimate and a likelihood. The likelihood, prior model probabilities, and model switching probabilities are then used to compute new model probabilities. An overall state estimate is computed with the new state estimates and their model probabilities. The IMM algorithm is outlined for *N* models in the following 5 steps, as illustrated in Fig. 2.2

Step 1 Mixing of state estimates

The filtering process starts with state estimates $\hat{X}_j(k-1|k-1)$, state error covariance $P_j(k-1|k-1)$, and associated model probabilities $\mu_j(k-1)$. The initial state estimate for model *j* at time *k*, $M_j(k)$, is computed as

$$\hat{X}_{0j}(k-1|k-1) = \sum_{i=1}^{N} \mu_{i|j}(k-1|k-1)\,\hat{X}_i(k-1|k-1) \tag{2.41}$$

where

$$\mu_{i|j}(k-1|k-1) = \frac{1}{\bar{c}_j}\mu_i(k-1)\,P_{ij}$$
$$\bar{c}_j = \sum_{i=1}^{N} \mu_i(k-1)\,P_{ij} \tag{2.42}$$

The mixed covariance for $M_j(k)$ is computed as

$$P_{0j}(k-1|k-1) = \sum_{i=1}^{N} \mu_{i|j}(k-1|k-1)[P_i(k-1|k-1) + (\hat{X}_i(k-1|k-1) - \hat{X}_{0j}(k-1|k-1))(\hat{X}_i(k-1|k-1) - \hat{X}_{0j}(k-1|k-1))^{\mathrm{T}}] \tag{2.43}$$

Step 2 Model-conditioned updates

The Kalman filtering equations provide the model-conditioned updates.

$$\hat{X}_j(k|k-1) = F_j(k-1)\,\hat{X}_{0j}(k-1|k-1) + G_j(k-1)\,\overline{W}_j(k-1) \tag{2.44}$$

$$P_j(k|k-1) = F_j(k-1)\,P_{0j}(k-1|k-1)\,F_j^{\mathrm{T}}(k-1) + G_j(k-1)\,Q_j(k-1)\,G_j^{\mathrm{T}}(k-1) \tag{2.45}$$

$$v_j(k) = Z(k) - H_j(k)\,\hat{X}_j(k|k-1) \tag{2.46}$$

$$S_j(k) = H_j(k)\,P_j(k|k-1)\,H_j^{\mathrm{T}}(k) + R_j(k) \tag{2.47}$$

$$K_j(k) = P_j(k \mid k-1) H_j^{\mathrm{T}}(k) S_j^{-1}(k) \tag{2.48}$$

$$\hat{X}_j(k \mid k) = \hat{X}_j(k \mid k-1) + K_j(k) v_j(k) \tag{2.49}$$

$$P_j(k \mid k) = P_j(k \mid k-1) - K_j(k) S_j(k) K_j^{\mathrm{T}}(k) \tag{2.50}$$

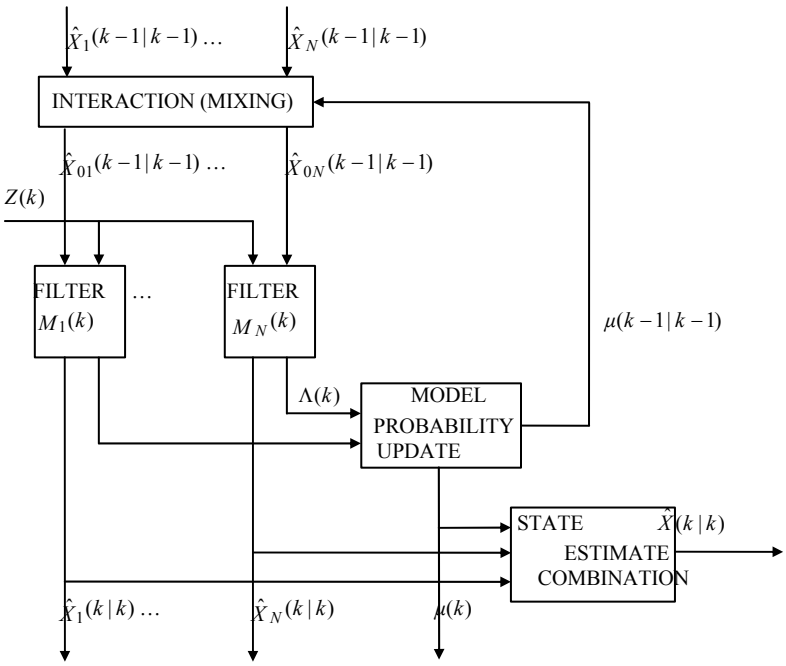


Fig. 2.2 IMM algorithm

Step 3 Model likelihood computations

The likelihood of $M_j(k)$, $\Lambda_j(k)$, is computed as

$$\Lambda_j(k) = \frac{1}{\sqrt{\left|2\pi S_j(k)\right|}} e^{-0.5 v_j^{\mathrm{T}}(k) S_j^{-1}(k) v_j(k)} \tag{2.51}$$

Step 4 Model probabilities update

The model probabilities are updated as

$$\mu_j(k) = \frac{1}{c} \Lambda_j(k) \bar{c}_j \quad \text{with} \quad c = \sum_{i=1}^{N} \Lambda_j(k) \bar{c}_i \tag{2.52}$$

Step 5 Combination of state estimates

The output state estimate and error covariance are obtained by

$$\hat{X}(k \mid k) = \sum_{i=1}^{N} \mu_i(k)\, \hat{X}_i(k \mid k) \tag{2.53}$$

$$P(k \mid k) = \sum_{i=1}^{N} \mu_i(k)[P_i(k \mid k) + (\hat{X}_i(k \mid k) - \hat{X}(k \mid k))(\hat{X}_i(k|k) - \hat{X}(k|k))^{\mathrm{T}}] \tag{2.54}$$

From the procedure of IMM filter, the IMM filter has to incorporates a large number of filters to cover the possible manoeuvre motions, resulting in heavy computation. The IMM filter also needs a prior suitable choice of model transition probability, which may be difficult to supply.

2.7 Problems In Tracking Using Other Adaptive Filters

Thorp (1973) and Bekir (1983) suggested increasing the parameter $Q(k)$, i.e., the Kalman filter process noise covariance matrix. Increasing this parameter effectively increases the bandwidth of the Kalman filter thus making the filter more responsive to manoeuvring targets. But, it is the remaining problem, the correction problem. This problem is important as significant errors are occurring at a time when it is critical that the tracker perform well. Potentially, loss of track can result.

In order to compensate the error of model, some researchers (Bar-Shalom and Fortmann, 1988) recommended modifying the Kalman filter by multiplication of the state covariance with a scalar ϕ>1 at every sampling time.

This amounts to letting

$$P^*(k \mid k) = \beta P(k \mid k) \quad (\beta > 1) \tag{2.55}$$

and then using $P^*(k \mid k)$ in the covariance update equation.

Multiplication of the state covariance by a scalar at every sampling time can compensate the error of manoeuver model. This result is that the past data are "discounted" and the future target dynamics is made more responsive by attaching a

higher covariance (lower accuracy) for the Kalman filter. However, this may lead to erratic and/or counterintuitive filter behaviour.

2.8 Choice Of Algorithm For Tracking

Singer's augmented filter shows good performance for the target with a low level manoeuvre, but its performance rapidly degrades in the case of high level manoeuvre and is worse than that with a Kalman filter that assumes no manoeuvres, during constant-velocity, straight-line motion. Fortunately, if the manoeuvre level is estimated by input estimation technique, the modified Singer's filter can be constructed by considering target accelerations as a perturbation around the estimated acceleration level, so in this way, the modified Singer's filter would track any level manoeuver target well. This work uses the modified Singer's filter to provide the state estimation in the cases of uncertain manoeuvre and low level manoeuvre.

The input estimation shows poor performance in the case of low level manoeuvre because it tends to over-compensate the manoeuvre. However, the input estimation technique has good performance for the target in rectilinear acceleration motion with a high level manoeuvre if the manoeuvre is detected quickly. Thus, this work suggests to use Chapter 4 manoeuvre detection to get quick and accurate manoeuvre detection, and then to estimate the manoeuvre accurately using IE to avoid of reinitialising for manoeuvre model like the VD filter.

Therefore, this thesis proposes a new multiple-model filter for tracking manoeuvring targets. This new multiple model approach employs three manoeuvre models. One is straight-line acceleration manoeuvre, one is curvilinear acceleration manoeuvre, and the other is circular motion. The switching decision from one model to another is made on

the basis of the manoeuvre detection and the estimated accelerations. For any ambiguous manoeuvre, the tracking is provided by the modified Singer's filter. The procedure of the new MM filter has no compromises between non-manoeuvre and manoeuvre, or between manoeuvres except the ambiguous cases, producing a good performance. The new MM filter is simple like the VD filter but has no need for reconstruction of manoeuvre, and the new MM filter also considers all kind of motions like IMM filter but with a limited number of models, reducing computational load. The new MM filter will be discussed in Chapter 5 in detail.

A further improvement for multiple-model tracking is provided by using a fixed-lag smoothing technique. The fixed-lag smoothing technique will be described in Chapter 6.

2.9 Summary

This chapter gave a detailed analysis of tracking techniques for manoeuvring targets in the literature. Singer's filter (1970) shows good performance for the target with a low manoeuuvre, but its performance rapidly degrades in the case of high manoeuvre.

IE algorithm (Chan et al., 1979; 1982) consists of the estimation of the unknown input (manoeuvre) over a sliding window and suitable compensation of the state estimate under the assumption that the manoeuvre onset time is the starting point of the sliding window. It is apparent that the performance of IE algorithm will be degraded. And also this technique shows poor performance in case of low level manoeuvre because it tends to over-compensate the manoeuvre. Bogler's technique(Bogler, 1987), while superior to the original IE algorithm (Chan et al, 1979; 1982), requires a large number of filters (together with input estimators and state estimate correctors) to run in parallel, resulting in requiring a significant amount of computation and memory.

The performance of the VD filter (Bar-Shalom et al,1982) is superior to that of the IE filter in the case of low level manoeuvre, and the computational requirements of the VD algorithm is less than that of the IE algorithm. However, the VD algorithm may increase the tracking error because of the reconstruction of manouvre model.

The MM filter usually consists of a large number of filters to cover possible manoeuvres, causing heavy computation. This approach is based on the "non-interacting" MM method: the single-model-based filters are running in parallel without mutual interaction, i.e., each filter operates independently at all times. Such an approach is quite effective in handling problems with an unknown structure or parameter but without structural or parametric changes. If the system structure or parameter changes, the MM filter could fail for tracking unless the models are updated.

The interacting multiple model (IMM) algorithm is a suboptimal hybrid filter. The IMM algorithm has been shown to be one of the most cost-effective schemes for the estimation of hybrid systems. Its main feature is its ability to estimate the state of a dynamic system which can switch between many behaviour modes according to a pre-defined Markov switching process. In particular, the IMM filter can act as a self-adjusting variable-bandwidth filter for tracking manoeuvring targets. Usually, the IMM filter incorporates a large number of filters to cover the possible manoeuvre motions, resulting in heavy computation. To reduce the computation load of IMM, the several adaptive IMM filters have been developed. However, all those methods above of adaptive IMM filter construction rely on the accuracy of estimated acceleration. If the estimated acceleration is not good enough, it would cause these adaptive IMM filters to fail for tracking.

The proposed MM filter consists of modified Singer's filter which shows good performance around estimated manoeuvre and during the manouvre changes, and three manoeuvre models which are straight-line acceleration manoeuvre, curvilinear acceleration manoeuvre, and circular motion. The switching decision from one model to another is made by the manoeuvre detection and the estimated accelerations. For any ambiguous manoeuvre, the tracking is provided by modified Singer's filter. The new MM filter has no any compromises between non-manoeuvre and manoeuvre, and between manoeuvres except the ambiguous cases, producing a good performance. The operation of the new MM filter is simple like the VD filter but no need for reconstruction of manoeuvre, and the new MM filter also considers all kind of motions like IMM filter but with a limited number of models, reducing computational load.

CHAPTER 3

TARGET MOTION MODELS

This chapter introduces the motion models used in both the tracking and interception of targets. The most general models are derived from mechanics and involve the forcing acting on the targets, such as thrust, lift and drag. However, these models are too complicated to implement in a practical system. The usual approach is to use kinematics to construct the motion models. This chapter provides detailed and systematic models from the simplest constant-velocity motion to the more complex constant-rate-of-acceleration-variation motion via kinematics. This chapter also presents Best and Norton's kinematics model which is of benefit to the tracking of curvilinear motion targets, and reviews Singer's model which is useful for manoeuvring target tracking during the periods when the target changes manoeuvre.

3.1 Introduction

There are a wide variety of models available for target tracking, ranging from the very simple four-state-variable constant-velocity models to complex models based on mechanics and involved the forces acting upon the targets (Berg, 1983; Mook and Shyu, 1992). The latter may be sophisticated models with many variables and are often unwieldy to implement. Also we usually do not know or can not estimate the forces. Most of the target tracking literature uses kinematics models which are based solely on the target's position, velocity and acceleration. The kinematics models are utilized throughout this work.

Since the measurements usually come in the form of a range and bearing, the most obvious choice of modelling system is polar co-ordinates (Gholson and Moose, 1997, Aidala and Hammel, 1983). However, the modelling system of polar co-ordinates means that even the simplest of target motions, straight-line travel at constant speed, results in a non-linear target model. The most common choice of co-ordinates is Cartesian co-ordinates with the origin at the tracker's initial position. The measurements are then non-linear in state but techniques to overcome this non-linearity are available and discussed in Section 3.8. This work chooses the Cartesian co-ordinates for system modelling.

For the target trajectories, the trajectory candidates could be constant-velocity motion, rectilinear acceleration motion, and curvilinear acceleration motion. For constant-velocity motion, the target trajectories are modelled as straight-line constant-velocity with slight changes in speed (Bar-Shalom and Fortmann, 1988). For simplicity, the rectilinear acceleration motion only considers constant-acceleration and constant-rate-of-acceleration-variation. The constant-acceleration motion is modelled as straight-line constant acceleration with slight changes in acceleration. The constant-rate-of-acceleration-variation motion is modelled as constant rate of acceleration variation with slight changes in the rate. All these rectilinear motion models do not explicitly model the cross-track acceleration. If the cross-track acceleration is modelled, the direction of the accelerations' application is state- and hence time-dependent. As a consequence, the model for the target motion becomes non-linear. To overcome this, the rate of change of target heading is assumed small so that the acceleration vector can be assumed constant for at least one observation interval. This results in the second-order kinematics model (Bar-Shalom and Birmiwal, 1982, Chan and Couture, 1993) and the constant-

acceleration motion model mentioned above or equations (3.8), (3.17) and (3.18) below with zero cross-track accelerations. However, the approximation causes loss of track when compared to true target motion. The alternative method is to assume the target does not change speed and so follows a circular path (Lerro and Bar-Shalom, 1993, Roecker and MrGillem, 1989) with zero along-track accelerations. In practice, the manoeuvres are much more complicated; the accelerations acting on the target may be both along-track and cross-track. For the curvilinear motion, Best and Norton suggest a more general model with both cross- and along-track accelerations (Best and Norton, 1997).

The main problem with the rectilinear acceleration model and curvilinear acceleration model above is that accelerations are assumed constant in the x and y directions, and cross-track and along-track directions, over each sampling interval. Therefore, if these models are used in target tracking, significant modelling errors arise when the target accelerations suddenly change in the x, y, cross-track or along-track directions. However, because of the rectilinear acceleration model's linearity, it was used in many of the target trackers. Singer (Singer, 1970) developed a tracker which is still linear but has a better performance than the rectilinear-acceleration and curvilinear-acceleration models during the periods when the target changes manoeuvre. He augments the state model with a target-acceleration equation represented by a first-order autoregressive process. Singer's model filter tracks the manoeuvre well during the periods when the target changes manoeuvre. This model was presented in Section 2.2 of Chapter 2 and is used in this work. This model will not be reiterated in this chapter.

The noise processes affecting the target motion must also be considered in tracker design. For convenience, they are usually modelled as white noise with a zero-mean

white Gauss distribution, for example, the slight changes in speed of the constant-velocity target motion model are modelled as random velocities with a zero-mean white Gauss distribution, and the slight changes in acceleration of the constant-acceleration target model are modelled as random accelerations with a zero-mean white Gauss distribution.

Section 3.2 derives the constant-velocity target model, Section 3.3 and Section 3.4 present the constant-acceleration target model and the constant-rate-of-acceleration-variation target model respectively. Section 3.5, Section 3.6 and Section 3.7 review Best and Norton's cross- and along-track acceleration motion model, and circular motion model respectively. Section 3.5, Section 3.6 and Section 3.7 also modify Best and Norton's models to broaden their application. Section 3.8 gives the polar-to-Cartesian measurement conversion. Section 3.9 presents a summary of the models.

3.2 Constant-Velocity Motion Model

The following equations (Sections 3.2 to 3.4) only give one-dimensional motion models. For two- or three-dimensional motion, the modelling scheme is the same as what follows.

A constant-velocity target for a generic coordinate x is described by the equation

$$\ddot{x}(t) = 0 \tag{3.1}$$

Since the position $x(t)$ evolves (in the absence of noise) according to a polynomial in time, such a model is also called polynomial and the resulting filter is called a polynomial filter.

In practice, the velocity undergoes at least slight changes. This can be modelled by the continuous-time white noise $\tilde{v}$ as follows:

$$\ddot{x}(t) = \tilde{v}(t) \tag{3.2}$$

where

$$E\tilde{v}(t) = 0 \tag{3.3}$$

$$E\{\tilde{v}(t)\tilde{v}(\tau)\} = q(t)\delta(t-\tau) \tag{3.4}$$

The state vector corresponding to (3.2) is

$$X = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \tag{3.5}$$

In many applications, the same model is used for each coordinate, and the motion along each coordinate is assumed "decoupled" from the others. The noises entering into the various coordinates are also assumed mutually independent but with possibly different and time-varying intensities.

The continuous-time state equation is

$$\dot{X}(t) = AX(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix}\tilde{v}(t) \tag{3.6}$$

where

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \tag{3.7}$$

The discrete-time state equation with sampling interval T is

$$X(k+1) = FX(k) + V(k) \tag{3.8}$$

where

$$F = e^{AT} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \tag{3.9}$$

and the discrete-time process noise relates to the continuous-time version as follows:

$$V(k) = \int_0^T e^{A(T-\tau)} \begin{bmatrix} 0 \\ 1 \end{bmatrix}\tilde{v}(kT+\tau)d\tau \tag{3.10}$$

The covariance of $V(k)$, assuming q to be constant and using (3.4), is

$$Q = E[V(k)V^{\mathrm{T}}(k)] = \int_0^T \begin{bmatrix} T-\tau \\ 1 \end{bmatrix} \begin{bmatrix} T-\tau & 1 \end{bmatrix} q d\tau = \begin{bmatrix} \frac{1}{3}T^3 & \frac{1}{2}T^2 \\ \frac{1}{2}T^2 & T \end{bmatrix} q \tag{3.11}$$

3.3 Constant-Acceleration Motion Model

A constant-acceleration target for a generic coordinate *x* is descried by the equation

$$\dddot{x}(t) = 0 \tag{3.12}$$

As in (3.2), the acceleration is never exactly constant and its slight changes can be modelled by zero-mean white noise as follows

$$\dddot{x}(t) = \tilde{v}(t) \tag{3.13}$$

(The smaller the variance q of $\tilde{v}$, the more nearly constant is the acceleration.) The state vector corresponding to the above is

$$X = \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix} \tag{3.14}$$

and its continuous-time state equation is

$$\dot{X}(t) = AX(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tilde{v}(t) \tag{3.15}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.16}$$

The discrete time state equation with sampling interval *T* is as in (3.8) with

$$F = e^{AT} = \begin{bmatrix} 1 & T & \frac{1}{2}T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \tag{3.17}$$

and the covariance matrix of $V(k)$, the process noise discretized from continuous time, is

$$Q = E[V(k)V^{\mathrm{T}}(k)] = \begin{bmatrix} \frac{1}{20}T^5 & \frac{1}{8}T^4 & \frac{1}{6}T^3 \\ \frac{1}{8}T^3 & \frac{1}{3}T^3 & \frac{1}{2}T^2 \\ \frac{1}{6}T^3 & \frac{1}{2}T^2 & T \end{bmatrix} q \tag{3.18}$$

3.4 Constant-Rate-Of-Acceleration-Variation Motion Model

A constant-rate-of-acceleration-variation target for a generic coordinate *x* is described by the equation

$$x^{(4)}(t) = 0 \tag{3.19}$$

As in (2), the acceleration-variation-rate is never exactly constant and its slight changes can be modelled by zero-mean white noise as follows

$$x^{(4)}(t) = \tilde{v}(t) \tag{3.20}$$

(The smaller the variance *q* of $\tilde{v}$, the more nearly constant is the acceleration-variation-rate.) The state vector corresponding to the above is

$$X = \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ \dddot{x} \end{bmatrix} \tag{3.21}$$

and its continuous-time state equation is

$$\dot{X}(t) = AX(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tilde{v}(t) \tag{3.22}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.23}$$

The discrete time state equation with sampling interval T is as in (3.8) with

$$F = e^{AT} = \begin{bmatrix} 1 & T & \frac{1}{2}T^2 & \frac{1}{6}T^3 \\ 0 & 1 & T & \frac{1}{2}T^2 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.24}$$

and the covariance matrix of $V(k)$, the process noise discretized from continuous time, is

$$Q = E[V(k)V^{\mathrm{T}}(k)] = \begin{bmatrix} \frac{1}{252}T^7 & \frac{1}{72}T^6 & \frac{1}{30}T^5 & \frac{1}{24}T^4 \\ \frac{1}{72}T^6 & \frac{1}{20}T^5 & \frac{1}{8}T^4 & \frac{1}{6}T^3 \\ \frac{1}{30}T^5 & \frac{1}{8}T^4 & \frac{1}{3}T^3 & \frac{1}{2}T^2 \\ \frac{1}{24}T^4 & \frac{1}{6}T^3 & \frac{1}{2}T^2 & T \end{bmatrix} q \tag{3.25}$$

The discrete-time plant equation for the two-state kinematic model excited by continuous-time white noise is given by (3.8)-(3.9) with the discrete-time process noise covariance (3.11). Similarly, for a three-state kinematic model, the discretized model is given by (3.8), (3.17) and (3.18); for a four-state kinematic model, the discretized model is given by (3.8), (3.24) and (3.25).

3.5 Cross- And Along-Track-Acceleration Motion Model

The model developed by Best and Norton (1997) allows both cross- and along-track accelerations but assumes that the proportional change in speed over one observation interval is small.

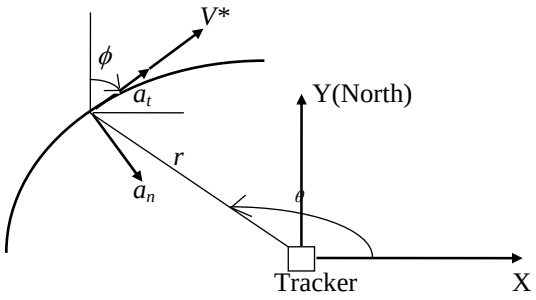


Fig. 3.1 Model of target motion.

The minimum set of state variable for a target with two-dimensional motion consists of two position and two velocity components. The position origin is arbitrary and placed at the tracker platform (assumed static). The velocity can be described by a speed *V** and heading ϕ measured clockwise from North. The target may have an along-track acceleration a_t and/or an acceleration a_n normal to the velocity vector (Fig. 3.1). The model splits the acceleration vector into a component constant over each updating interval, treated as known in the updating, and an unknown forcing (process noise) component, the effects of which, over successive updating intervals, may be treated as white, zero-mean state increments.

The continuous-time model is

$$\frac{d\phi}{dt} = \frac{a_n(t)}{V^*(t)} \tag{3.26a}$$

$$\frac{dV^*(t)}{dt} = a_t(t) \tag{3.26b}$$

$$\frac{dx}{dt} = V^*(t)\sin\phi(t) \tag{3.26c}$$

$$\frac{dy}{dt} = V^*(t)\cos\phi(t) \tag{3.26d}$$

where $V^*(t)$ is the vector magnitude.

which are the standard equations of curvilinear motion (Becker, 1954).

The model (3.26) has non-linearities in three of the state equations. However, a simplification is to define the velocity by its components

$$\dot{x}(t) = V^*(t)\sin\phi(t) \tag{3.27a}$$

$$\dot{y}(t) = V^*(t)\cos\phi(t) \tag{3.27b}$$

giving

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \ddot{x}(t) \\ \ddot{y}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \\ \dot{x}(t) \\ \dot{y}(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \dfrac{\dot{x}(t)}{\sqrt{\dot{x}^2(t)+\dot{y}^2(t)}} & \dfrac{\dot{y}(t)}{\sqrt{\dot{x}^2(t)+\dot{y}^2(t)}} \\ \dfrac{\dot{y}(t)}{\sqrt{\dot{x}^2(t)+\dot{y}^2(t)}} & -\dfrac{\dot{x}(t)}{\sqrt{\dot{x}^2(t)+\dot{y}^2(t)}} \end{bmatrix} \begin{bmatrix} a_t \\ a_n \end{bmatrix} \tag{3.28}$$

which is of the form

$$\dot{X}(t) = A\dot{X}(t) + B(X(t))\mathbf{a}(t) \tag{3.29}$$

The discrete-time state equation with sampling interval T is

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} +$$

$$\begin{bmatrix} \dfrac{T\cos\phi_k}{\omega_k} + \dfrac{\sin\phi_k}{\omega_k^2} - \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k^2} & \dfrac{\cos\phi_k}{\omega_k^2} - \dfrac{T\sin\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k^2} \\ \dfrac{\cos\phi_k}{\omega_k^2} - \dfrac{T\sin\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k^2} & \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k^2} - \dfrac{\sin\phi_k}{\omega_k^2} - \dfrac{T\cos\phi_k}{\omega_k} \\ \dfrac{\cos\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k} & \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k} - \dfrac{T\cos\phi_k}{\omega_k} \\ \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\sin\phi_k}{\omega_k} & \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\cos\phi_k}{\omega_k} \end{bmatrix} \begin{bmatrix} a_{tk} \\ a_{nk} \end{bmatrix} \tag{3.30}$$

where

$\phi_k = \tan^{-1}\dfrac{\dot{x}_k}{\dot{y}_k}$, $\omega_k = \dfrac{a_{nk}}{\sqrt{\dot{x}_k^2 + \dot{y}_k^2}}$, $\begin{bmatrix} a_{tk} \\ a_{nk} \end{bmatrix} = \begin{bmatrix} a_{tk}^{\det} \\ a_{nk}^{\det} \end{bmatrix} + \begin{bmatrix} n_{tk} \\ n_{nk} \end{bmatrix}$, and $\begin{bmatrix} a_{tk}^{\det} \\ a_{nk}^{\det} \end{bmatrix}$ is deterministic accelerations.

Suppose $\begin{bmatrix} a_{t(k+1)} \\ a_{n(k+1)} \end{bmatrix} = \begin{bmatrix} a_{tk} \\ a_{nk} \end{bmatrix} + \begin{bmatrix} n_{tk} \\ n_{nk} \end{bmatrix}$ and acceleration varies near-linearly over $kT < t \le (k+1)T$, then equation (3.30) is approximated by

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} +$$

$$\begin{bmatrix} \dfrac{T\cos\phi_k}{\omega_k} + \dfrac{\sin\phi_k}{\omega_k^2} - \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k^2} & \dfrac{\cos\phi_k}{\omega_k^2} - \dfrac{T\sin\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k^2} \\ \dfrac{\cos\phi_k}{\omega_k^2} - \dfrac{T\sin\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k^2} & \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k^2} - \dfrac{\sin\phi_k}{\omega_k^2} - \dfrac{T\cos\phi_k}{\omega_k} \\ \dfrac{\cos\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k} & \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k} - \dfrac{T\cos\phi_k}{\omega_k} \\ \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\sin\phi_k}{\omega_k} & \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\cos\phi_k}{\omega_k} \end{bmatrix} \frac{\begin{bmatrix} a_{tk} + a_{t(k+1)} \\ a_{nk} + a_{n(k+1)} \end{bmatrix}}{2}$$

Thus, the augmented state model can be formulated as

$$
\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \\ a_{t(k+1)} \\ a_{n(k+1)} \end{bmatrix} =
\begin{bmatrix}
1 & 0 & T & 0 & \frac{T\cos\phi_k}{\omega_k} + \frac{\sin\phi_k}{\omega_k^2} - \frac{\sin(\phi_k+\omega_k T)}{\omega_k^2} & \frac{\cos\phi_k}{\omega_k^2} - \frac{T\sin\omega_k}{\omega_k} - \frac{\cos(\phi_k+\omega_k T)}{\omega_k^2} \\
0 & 1 & 0 & T & \frac{\cos\phi_k}{\omega_k^2} - \frac{T\sin\phi_k}{\omega_k} - \frac{\cos(\phi_k+\omega_k T)}{\omega_k^2} & \frac{\sin(\phi_k+\omega_k T)}{\omega_k^2} - \frac{\sin\phi_k}{\omega_k^2} - \frac{T\cos\phi_k}{\omega_k} \\
0 & 0 & 1 & 0 & \frac{\cos\phi_k}{\omega_k} - \frac{\cos(\phi_k+\omega_k T)}{\omega_k} & \frac{\sin(\phi_k+\omega_k T)}{\omega_k} - \frac{T\cos\phi_k}{\omega_k} \\
0 & 0 & 0 & 1 & \frac{\sin(\phi_k+\omega_k T)}{\omega_k} - \frac{\sin\phi_k}{\omega_k} & \frac{\cos(\phi_k+\omega_k T)}{\omega_k} - \frac{\cos\phi_k}{\omega_k} \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \\ a_{tk} \\ a_{nk} \end{bmatrix}
+
\begin{bmatrix}
\frac{T\cos\phi_k}{2\omega_k} + \frac{\sin\phi_k}{2\omega_k^2} - \frac{\sin(\phi_k+\omega_k T)}{2\omega_k^2} & \frac{\cos\phi_k}{2\omega_k^2} - \frac{T\sin\phi_k}{2\omega_k} - \frac{\cos(\phi_k+\omega_k T)}{2\omega_k^2} \\
\frac{\cos\phi_k}{2\omega_k^2} - \frac{T\sin\phi_k}{2\omega_k} - \frac{\cos(\phi_k+\omega_k T)}{2\omega_k^2} & \frac{\sin(\phi_k+\omega_k T)}{2\omega_k^2} - \frac{\sin\phi_k}{2\omega_k^2} - \frac{T\cos\phi_k}{2\omega_k} \\
\frac{\cos\phi_k}{2\omega_k} - \frac{\cos(\phi_k+\omega_k T)}{2\omega_k} & \frac{\sin(\phi_k+\omega_k T)}{2\omega_k} - \frac{T\cos\phi_k}{2\omega_k} \\
\frac{\sin(\phi_k+\omega_k T)}{2\omega_k} - \frac{\sin\phi_k}{2\omega_k} & \frac{\cos(\phi_k+\omega_k T)}{2\omega_k} - \frac{\cos\phi_k}{2\omega_k} \\
1 & 0 \\
0 & 1
\end{bmatrix}
\begin{bmatrix} n_{tk} \\ n_{nk} \end{bmatrix}
\tag{3.31}
$$

3.6 Circular Motion Model

By setting the deterministic component of along-track acceleration in (3.30) to zero, the state equation of circular motion model, i.e. constant cross-track acceleration model, can be reformulated as

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dfrac{\sin(\omega_k T)}{\omega_k} & \dfrac{1-\cos(\omega_k T)}{\omega_k} \\ 0 & 1 & \dfrac{\cos(\omega_k T)-1}{\omega_k} & \dfrac{\sin(\omega_k T)}{\omega_k} \\ 0 & 0 & \cos(\omega_k T) & \sin(\omega_k T) \\ 0 & 0 & -\sin(\omega_k T) & \cos(\omega_k T) \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} +$$

$$\begin{bmatrix} \dfrac{T\cos\phi_k}{\omega_k} + \dfrac{\sin\phi_k}{\omega_k^2} - \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k^2} & \dfrac{\cos\phi_k}{\omega_k^2} - \dfrac{T\sin\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k^2} \\ \dfrac{\cos\phi_k}{\omega_k^2} - \dfrac{T\sin\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k^2} & \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k^2} - \dfrac{\sin\phi_k}{\omega_k^2} - \dfrac{T\cos\phi_k}{\omega_k} \\ \dfrac{\cos\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k} & \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\sin\phi_k}{\omega_k} \\ \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\sin\phi_k}{\omega_k} & \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\cos\phi_k}{\omega_k} \end{bmatrix} \begin{bmatrix} n_{tk} \\ n_{nk} \end{bmatrix} \tag{3.32}$$

where

$$\omega_k = \frac{a_n^{\det}}{\sqrt{\dot{x}_k^2 + \dot{y}_k^2}}$$

Suppose $a_{n(k+1)} = a_{nk} + n_{nk}$, then the augmented state model can be formulated as

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \\ a_{n(k+1)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dfrac{\sin(\omega_k T)}{\omega_k} & \dfrac{1-\cos(\omega_k T)}{\omega_k} & 0 \\ 0 & 1 & \dfrac{\cos(\omega_k T)-1}{\omega_k} & \dfrac{\sin(\omega_k T)}{\omega_k} & 0 \\ 0 & 0 & \cos(\omega_k T) & \sin(\omega_k T) & 0 \\ 0 & 0 & -\sin(\omega_k T) & \cos(\omega_k T) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \\ a_{nk} \end{bmatrix} +$$

$$\begin{bmatrix} \dfrac{T\cos\phi_k}{\omega_k} + \dfrac{\sin\phi_k}{\omega_k^2} - \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k^2} & \dfrac{\cos\phi_k}{\omega_k^2} - \dfrac{T\sin\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k^2} \\ \dfrac{\cos\phi_k}{\omega_k^2} - \dfrac{T\sin\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k^2} & \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k^2} - \dfrac{\sin\phi_k}{\omega_k^2} - \dfrac{T\cos\phi_k}{\omega_k} \\ \dfrac{\cos\phi_k}{\omega_k} - \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k} & \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\sin\phi_k}{\omega_k} \\ \dfrac{\sin(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\sin\phi_k}{\omega_k} & \dfrac{\cos(\phi_k+\omega_k T)}{\omega_k} - \dfrac{\cos\phi_k}{\omega_k} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} n_{tk} \\ n_{nk} \end{bmatrix} \tag{3.33}$$

Several other authors have used a model for target motion corresponding to a circular track (Dufour and Mariton, 1992, Lerro and Bar-Shalom, 1993, Tanner et al, 1993). Here their corresponding circular motion models are not reiterated.

3.7 Straight-Line Acceleration Motion Model

By setting the deterministic component of cross-track acceleration in (3.30) to zero, the state equation of straight-line acceleration model, i.e. along-track acceleration model, can be formulated approximately as

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} + \begin{bmatrix} \frac{T^2 \sin\phi_k}{2} & \frac{T^2 \cos\phi_k}{2} \\ \frac{T^2 \cos\phi_k}{2} & \frac{T^2 \sin\phi_k}{2} \\ T\sin\phi_k & T\cos\phi_k \\ T\cos\phi_k & -T\sin\phi_k \end{bmatrix} \begin{bmatrix} a_{tk} \\ n_{nk} \end{bmatrix} \tag{3.34}$$

Suppose $a_{t(k+1)} = a_{tk} + n_{tk}$ and acceleration varies near-linearly over $kT < t \le (k+1)T$, then equation (3.34) is approximated by

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} + \begin{bmatrix} \frac{T^2 \sin\phi_k}{2} & \frac{T^2 \cos\phi_k}{2} \\ \frac{T^2 \cos\phi_k}{2} & -\frac{T^2 \sin\phi_k}{2} \\ T\sin\phi_k & T\cos\phi_k \\ T\cos\phi_k & -T\sin\phi_k \end{bmatrix} \begin{bmatrix} \frac{a_{t(k+1)} + a_{tk}}{2} \\ n_{nk} \end{bmatrix}$$

thus, the augmented state model can be formulated as

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \\ a_{t(k+1)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & T & 0 & \frac{T^2 \sin\phi_k}{2} \\ 0 & 1 & 0 & T & \frac{T^2 \cos\phi_k}{2} \\ 0 & 0 & 1 & 0 & T\sin\phi_k \\ 0 & 0 & 0 & 1 & T\cos\phi_k \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \\ a_{tk} \end{bmatrix} + \begin{bmatrix} \frac{T^2 \sin\phi_k}{2} & \frac{T^2 \cos\phi_k}{2} \\ \frac{T^2 \cos\phi_k}{2} & -\frac{T^2 \sin(\phi_k)}{2} \\ T\sin\phi_k & T\cos\phi_k \\ T\cos\phi_k & -T\sin\phi_k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} n_{tk} \\ n_{nk} \end{bmatrix}$$

(3.35)

3.8 Polar-To-Cartesian Measurement Conversion

This work assumes that the sensor has range and bearing information. The measurements themselves are noise corrupted and the uncertainty in the measurements is usually modelled as an additive zero-mean white Gaussian process. Some individual measurements may be more than the Gaussian distribution would normally allow for, and these are called outliers. They are caused by target scintillation (glint). This work ignores outliers.

The measured range r_m and measured bearing θ_m are defined w.r.t. the true range *r* and true bearing θ (as shown in Fig. 3.2) as

$$r_m = r + \tilde{r} \qquad \theta_m = \theta + \tilde{\theta} \tag{3.36}$$

where the errors $\tilde{r}$, $\tilde{\theta}$, are assumed to be independent with zero mean and standard deviations σ_r and σ_θ, respectively.

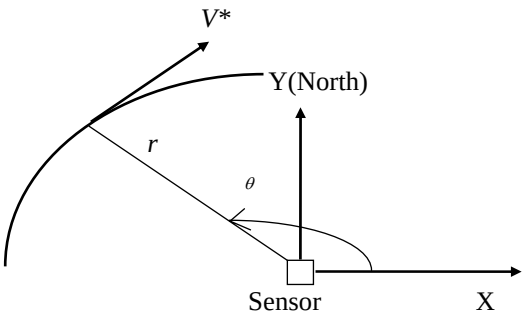


Fig. 3.2 Model of measurement

The standard conversion (Bar-Shalom and Li, 1995)

These polar measurements are transformed to Cartesian by the standard coordinate conversion as follows

$$x_m = r_m \cos\theta_m \qquad y_m = r_m \sin\theta_m \tag{3.37}$$

The error statistics obtained from linearization

Denoting by (x, y) the true Cartesian position and taking the first order terms of the Taylor series expansion of (3.37) at (r_m, θ_m), i.e., using linearization, yields the Cartesian coordinate errors

$$x_m - x \approx \tilde{r} \cos\theta_m - \tilde{\theta}\, r_m \sin\theta_m \equiv \tilde{x}_L \tag{3.38}$$

$$y_m - y \approx \tilde{r} \sin\theta_m + \tilde{\theta}\, r_m \cos\theta_m \equiv \tilde{y}_L \tag{3.39}$$

The mean of the errors as given by (3.38)-(3.39) is zero

$$\mu_L \equiv \begin{bmatrix} E[\tilde{x}_L] \\ E[\tilde{y}_L] \end{bmatrix} = 0 \tag{3.40}$$

The elements of the corresponding covariance matrix R_L are

$$R_L^{11} \equiv \mathrm{var}(\tilde{x}_L) = r_m^2 \sigma_\theta^2 \sin^2\theta_m + \sigma_r^2 \cos^2\theta_m \tag{3.41}$$

$$R_L^{22} \equiv \mathrm{var}(\tilde{y}_L) = r_m^2 \sigma_\theta^2 \cos^2\theta_m + \sigma_r^2 \sin^2\theta_m \tag{3.42}$$

$$R_L^{12} \equiv \mathrm{cov}(\tilde{x}_L, \tilde{y}_L) = (\sigma_r^2 - r_m^2 \sigma_\theta^2) \sin\theta_m \cos\theta_m \tag{3.43}$$

The true error statistics (Bar-Shalom and Li, 1995)

Rather than using linearization as in (3.38)-(3.39), the exact errors $\tilde{x}$ and $\tilde{y}$ in the Cartesian coordinates can be found by expanding

$$x_m \equiv x + \tilde{x} = (r + \tilde{r})\cos(\theta + \tilde{\theta}) \tag{3.44}$$

$$y_m \equiv y + \tilde{y} = (r + \tilde{r})\sin(\theta + \tilde{\theta}) \tag{3.45}$$

using trigonometric identities, to obtain

$$\tilde{x} = r\cos\theta(\cos\tilde{\theta} - 1) - \tilde{r}\sin\theta\sin\tilde{\theta} - r\sin\theta\sin\tilde{\theta} + \tilde{r}\cos\theta\cos\tilde{\theta} \tag{3.46}$$

$$\tilde{y} = r\sin\theta(\cos\tilde{\theta} - 1) + \tilde{r}\cos\theta\sin\tilde{\theta} + r\cos\theta\sin\tilde{\theta} + \tilde{r}\sin\theta\cos\tilde{\theta} \tag{3.47}$$

The mean and covariance of the errors (3.44)-(3.45) can be obtained explicitly and exactly assuming

(i) zero-mean Gaussian errors in the polar measurements (3.36), and

(ii) knowledge of the true location (r,θ).

Assumption (i) yields

$$E[\cos\tilde{\theta}] = e^{-\frac{\sigma_\theta^2}{2}} \qquad E[\sin\tilde{\theta}] = 0 \qquad E[\sin\tilde{\theta}\cos\tilde{\theta}] = 0 \tag{3.48}$$

$$E[\cos^2\tilde{\theta}] = \frac{1+e^{-2\sigma_\theta^2}}{2} \qquad E[\sin^2\tilde{\theta}] = \frac{1-e^{-2\sigma_\theta^2}}{2} \tag{3.49}$$

With the above, the true mean of the error in the converted measurements (3.37), conditioned on the true location according to Assumption (ii), is obtained as

$$\mu_t(r,\theta) \equiv \begin{bmatrix} E[\tilde{x} \mid r,\theta] \\ E[\tilde{y} \mid r,\theta] \end{bmatrix} = \begin{bmatrix} r\cos\theta(e^{-\frac{\sigma_\theta^2}{2}} - 1) \\ r\sin\theta(e^{-\frac{\sigma_\theta^2}{2}} - 1) \end{bmatrix} \tag{3.50}$$

The elements of the true covariance R_t of the converted measurement are given by

$$\begin{aligned} R_t^{11} &\equiv \mathrm{var}(\tilde{x} \mid r,\theta) = r^2 e^{-\sigma_\theta^2}\{\cos^2\theta[\cosh(\sigma_\theta^2) - 1] + \sin^2\theta\sinh(\sigma_\theta^2)\} \\ &+ \sigma_r^2 e^{-\sigma_\theta^2}[\cos^2\theta\cosh(\sigma_\theta^2) + \sin^2\theta\sinh(\sigma_\theta^2)] \end{aligned} \tag{3.51}$$

$$\begin{aligned} R_t^{22} &\equiv \mathrm{var}(\tilde{y} \mid r,\theta) = r^2 e^{-\sigma_\theta^2}\{\sin^2\theta[\cosh(\sigma_\theta^2) - 1] + \sin^2\theta\sinh(\sigma_\theta^2)\} \\ &+ \sigma_r^2 e^{-\sigma_\theta^2}[\sin^2\theta\cosh(\sigma_\theta^2) + \cos^2\theta\sinh(\sigma_\theta^2)] \end{aligned} \tag{3.52}$$

$$R_t^{12} \equiv \mathrm{var}(\tilde{x},\tilde{y} \mid r,\theta) = \sin\theta\cos\theta\, e^{-2\sigma_\theta^2}[\sigma_r^2 + r^2(1 - e^{\sigma_\theta^2})] \tag{3.53}$$

Equations (3.50) and (3.51)-(3.53) are exact explicit expressions for the bias and covariance of the converted measurements.

The converted measurements have a significant bias for large cross-range errors (long range and large bearing errors).

Expressions (3.50) and (3.51)-(3.53) cannot be used due to Assumption (ii) — they are conditioned on the true values of range and bearing which are not available in practice. To make these results useful, the expected values of these true moments have to be evaluated conditioned on the measured position.

Use of the true error statistics in practice (Bar-Shalom and Li, 1995)

The average true converted measurement bias and average true converted measurement covariance are

$$E[\mu_t(r,\theta) \mid r_m, \theta_m] \equiv \mu_a \tag{3.54}$$

$$E[R_t(r,\theta) \mid r_m, \theta_m] \equiv R_a \tag{3.55}$$

Using (3.48)-(3.50) and applying trigonometric identities the mean (3.54) becomes

$$\mu_a = \begin{bmatrix} E[\tilde{x} \mid r_m, \theta_m] \\ E[\tilde{y} \mid r_m, \theta_m \end{bmatrix} = \begin{bmatrix} r_m \cos\theta_m (e^{-\sigma_\theta^2} - e^{-\frac{\sigma_\theta^2}{2}}) \\ r_m \sin\theta_m (e^{-\sigma_\theta^2} - e^{-\frac{\sigma_\theta^2}{2}}) \end{bmatrix} \tag{3.56}$$

Similarly, the covariance (3.55) has elements

$$\begin{aligned} R_a^{11} \equiv \text{var}(\tilde{x} \mid r_m, \theta_m) &= r_m^2 e^{-2\sigma_\theta^2} [\cos^2\theta_m (\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) \\ &+ \sin^2\theta_m (\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)] + \sigma_r^2 e^{-2\sigma_\theta^2} [\cos^2\theta_m (2\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) \\ &+ \sin^2\theta_m (2\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)] \end{aligned} \tag{3.57}$$

$$\begin{aligned} R_a^{22} \equiv \text{var}(\tilde{y} \mid r_m, \theta_m) &= r_m^2 e^{-2\sigma_\theta^2} [\sin^2\theta_m (\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) \\ &+ \cos^2\theta_m (\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)] + \sigma_r^2 e^{-2\sigma_\theta^2} [\sin^2\theta_m (2\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) \\ &+ \cos^2\theta_m (2\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)] \end{aligned} \tag{3.58}$$

$$R_a^{12} \equiv \text{var}(\tilde{x}, \tilde{y} \mid r_m, \theta_m) = \sin\theta_m \cos\theta_m e^{-4\sigma_\theta^2} [\sigma_r^2 + (r_m^2 + \sigma_r^2)(1 - e^{\sigma_\theta^2})] \tag{3.59}$$

Thus, with μ_a given in (3.56), the debiased conversion is

$$\begin{bmatrix} x^{dc} \\ y^{dc} \end{bmatrix} = \begin{bmatrix} r_m \cos\theta_m \\ r_m \sin\theta_m \end{bmatrix} - \mu_a \tag{3.60}$$

The average covariance of the conversion is R_a with elements (3.57)-(3.59), which are larger than those in (3.51)-(3.53) — they account for the additional errors incurred by evaluating it at the measured position.

3.9 Summary

This chapter gave the detailed and systematic rectilinear motion model and curvilinear motion model in Cartesian co-ordinates. The rectilinear model is simple and easy to construct but risks loss of track during a turn. In contrast, the curvilinear model is much more complex but effective during curvilinear motion. This chapter modified Best and Norton's curvilinear motion model with augmented state model to broaden its application.

The accuracy of curvilinear acceleration motion model relies on the accuracy of the estimated accelerations of cross-track and along-track, and heading. Thus, during the periods when the target changes manoeuvre, this approximate model becomes invalid. Furthermore, the cross-track acceleration and along-track acceleration are non-linearly related to position and speed of the target, therefore, the along-track acceleration and cross-track acceleration are difficult to estimate by using linear equations or linear estimation. So, the curvilinear acceleration model is used only in the case in which the target is absolutely sure in cross- and along-track acceleration motion.

Although the straight-line acceleration motion model can be derived from the curvilinear motion model, and is theoretically a special case of the curvilinear motion model with zero cross-track accelerations, but any estimation error in cross-track accelerations with the curvilinear motion model will increase the tracking errors of the

straight-line acceleration motion. For the same reason, the zero along-track acceleration motion model should be separately constructed.

Therefore, the models in a multiple model tracker include constant-velocity motion model (described in Section 3.2), cross-track acceleration motion model (described in Section 3.6), along-track acceleration motion model (described in Section 3.7), and along- and cross-track accelerations motion model (described in Section 3.5). Another model, Singer's model, is a very effective approximation for the target motion during the periods when the target changes manoeuvre. Singer's model (described in Section 2.2) is included in the multiple model as well. All these models will be utilised throughout the multiple model tracking algorithms.

The constant-velocity motion model is described by equations (3.8)-(3.11); the rectilinear constant-accelerations motion model is given by equations (3.8), (3.17) and (3.18); the rectilinear constant-rate-of-acceleration-variation motion model by equations (3.8), (3.24) and (3.25); cross- and along-track- acceleration motion model by equations (3.30) and (3.31); the circular motion model by equations (3.32) and (3.33); straight-line acceleration motion model by equations (3.34) and (3.35); and Singer's model, the correlated acceleration model, by equations (2.3) and (2.4) in Chapter 2.

CHAPTER 4

MANOEUVRE DETECTION

Without manoeuvre detection the tracking systems would exhibit unacceptable natural behaviour for manoeuvring targets. Quicker and more accurate detection of the manoeuvre could give better tracking. Good detection means quicker detection and fewer false alarms. This chapter reviews the classical and modern manoeuvre detection techniques, and also presents several new manoeuvre detection techniques to provide quicker detection and to improve the tracking.

4.1 Introduction

Manoeuvre detection has received growing attention during the last 30 years. McAulay and Denlinger (1973) use matched filtering to detect manoeuvre, which is to pass the residual sequence generated by the Kalman filter through a bank of filters matched to possible manoeuvre structures, and compare their output with a fixed threshold. When one of matched filters give the most output and the most output exceeds a threshold, the manoeuvre corresponding to that matched filter with the most output has occurred. However, these matched filters are not easy to construct in most manoeuvre cases because of the model complexity. Also, this approach requires the specification of the particular manoeuvre's structure, which may be difficult to supply.

Most methods for detecting a manoeuvre involve monitoring the innovations of a Kalman filter. These should be a zero-mean white Gauss sequence if the tracker model matches the target motion and all noise processes are zero-mean white Gauss distributed. When the target manoeuvres, the constant-velocity model becomes mismatched causing a bias

to appear in the innovation sequence. This can be used to detect manoeuvres. Unfortunately, for a constant-velocity target, the constant-velocity tracker produces a non-zero mean when finite window sized innovations are considered, leading to false alarms. Only as the window size tends to infinity will the bias due to noise approach zero. Consequently, a short window means fast detection and high false alarm rates; but a large window means a long delay.

A classical method is to use chi-squared tests to look for a structure in the innovations without specifying its form, which is to test the statistic of normalised squared-innovation. This manoeuvre detection typically suffers much delay or false alarm. Bar-Shalom and Birmiwal (1982) suggest that the end of the manoeuvre be detected by testing statistic of normalised squared-acceleration using the manoeuvring filter. However, the manoeuvring model has to completely reconstruct the process noise covariance and the estimate of the state variable within the sliding window when changing the the manoeuvre model. Furthermore, the filter only uses measurements at the starting point of sliding window to initilise the manoeuvre filter. Thus, it is difficult to reconstruct a good process-noise covariance in accord with the manoeuvre. The effectiveness and reliability depend on a good reconstruction of process-noise covariance of the manoeuvre filter. If the process-noise covariance is too large, it causes false alarms, but if it too small, detection is delayed. Thus, it is difficult to detect the end of a manoeuvre reliably by this method.

Chan and Couture (Chan and Couture, 1993) have developed a manoeuvre detector, which projects the target positions s intervals forward and compares the projections against measurements. This is different from the common innovations-based manoeuvre detector. In this detector, the Kalman filter estimates at time kT serve as the reference for

position projections towards (k+1), …, (k+s)T. If there is a manoeuvre, comparing the projections against the measurement will have a larger difference than comparing the Kalman filter estimates against the measurements (the innovation approach). Unfortunately, this detection method is susceptible to noise problems as well, a false alarm could be generated or the detection delayed.

Hägglund (1983) proposed a method to detect parameter changes in records with time-varying noise levels. Chen and Norton (1987) proposed that rapid parameter changes are detected by a vector sequence based on Hägglund's method. This method is effective, robust and reasonably simple. Its ability to track sudden parameter changes is excellent in comparison with some other well known methods. For the tracking, when the manoeuvre occurs, the statistic of normalised squared-innovation will increase in size abruptly. So the statistic of normailsed squared innovations is defined as the parameter in Chen and Norton's statistic, and then this test of rapid parameter changes would give effective manoeuvre detection, i.e., one of our proposed manoeuvre detection methods.

Weston and Norton (1997) recommend a simple and effective detection technique. It is in fact a scheme for detecting changes in forcing, in general. The technique detects impulses in the input by testing a statistic consisting of differences between forwards (filtered) prediction and backwards (smoothed) estimation for significance with offline processing, based on fixed-interval optimal smoothing. But unfortunately, it is used in offline processing, and cannot be directly applied into the tracking with online processing. Could it be modified to work on line (e.g. with fixed-lag smoothing)?

Zhang and Li (1997, 1998), and Isaksson and Gustafsson (1995) detect manoeuvres using the interacting multiple model (IMM) by observing the probability of each model. The

IMM algorithm is a suboptimal hybrid filter. The main feature of this algorithm is its ability to estimate the state of a dynamic system with several behaviour modes which can switch from one to another. In particular, the IMM filter can be a self-adjusting variable-bandwidth filter, which makes it natural for tracking manoeuvring targets. However, the IMM filter is constructed by a large number of filters to cover the possible manoeuvre motions. This results in heavy computation. This technique also needs a prior suitable choice of model transition probability, it may need tuning to match real target behaviour.

In order to overcome the problems in the technique of Bar-Shalom and Birmiwal (Bar-Shalom and Birmiwal, 1982), we use the augmented state model (with position, speed, and acceleration) throughout the process, with process noise covariance the same as the normal model, and fixed-lag smoothing is used to provide better estimates of the accelerations and their error covariance. The statistic and test is based on smoothed estimated accelerations. The test based on normalised squared smoothed-acceleration is reliable and quick for detecting the beginning and end of manoeuvres. This manoeuvre detection is discussed in Section 4.6 and demonstrated in Section 4.9.

The advantage of Weston and Norton's method is simplicity and effectiveness. If fixed-lag smoothing were used instead of fixed-interval smoothing in Weston and Norton's method, this would give a useful manoeuvre detection with online processing for the tracking. Fortunately, the fixed-lag smoothed estimates and their covariances can be generated by fixed-interval optimal smoothing algorithms (Brown and Hwang, 1992). The fixed-lag smoothed estimates and their covariances are part of the fixed-interval smoothed estimates and the associated covariances. So the more effective manoeuvre detection can be derived, based on the Weston and Norton’s change detection with fixed-

lag smoothing instead of fixed-interval smoothing, and this method can be used to find the start and the end of manoeuvre steps with online processing.

Therefore, this work proposes that the normalized squared-innovation is defined as parameter in Chen and Norton's method to provide a simple and effective manoeuvre detection; that the statistic of normalised squared smoothed-acceleration is used to give reliable and quick detection for detecting the beginning and end of manoeuvre; and that the statistic with fixed-lag smoothing, based on Weston and Norton's method, is used to obtain the more accurate manoeuvre steps.

Sections 4.2, 4.3, 4.4, and 4.5 review classical manoeuvre detection with the statistic of normalised squared innovations, manoeuvre detection based on the difference between projections and measurements, Chen and Norton's manoeuvre detection, and Weston and Norton's manoeuvre detection, respectively. Section 4.6 presents manoeuvre detection by testing normalised squared smoothed-acceleration statistic. Section 4.7 presents modified Chen and Norton's manoeuvre detection. Section 4.8 gives modified Weston and Norton's detection. Section 4.9 presents the simulations. Section 4.10 gives the summary of this chapter.

4.2 Classical Manoeuvre Detection By Testing The Statistic Of Normalised Squared-Innovation

One of the simplest detection methods uses the normalised squared-innovation

$$\varepsilon_v(k) = v_k^{\mathrm{T}} S_k^{-1} v_k \qquad (4.1)$$

(where v_k is the innovation, and S_k its covariance.)

which is a chi-square variate with n_z (the dimension of measurement) degrees of freedom when the measurements are commensurate with the linear model. A threshold is set up based on the target model (for the non-manoeuvring situation):

$$P\{\varepsilon_v(k) \leq \varepsilon_{max}\} = 1-\alpha \qquad (4.2)$$

If the threshold is exceeded, a manoeuvre is assumed to have occurred but this method suffers false alarms heavily because of the noise problems.

A less noise-sensitive method, uses a windowed sample mean of the normalised innovations squared:

$$\varepsilon_v(k) = \frac{1}{N} \sum_{i=k-N+1}^{k} v_i^{\mathrm{T}} S_i^{-1} v_i \qquad (4.3)$$

$N\varepsilon_v(k)$ is chi-square with Nn_z degrees of freedom. If the statistic exceeds a threshold, say, the 95% confidence bound from χ^2 tables, then a manoeuvre is deemed to have occurred because there is only a 5% chance. However, manoeuvre detection by testing this statistic incurs considerable delay or causes false alarms because of the noise problems. For example, if the noise over the sliding window samples produces measurements to one side of the track, a false alarm could be generated or the detection delayed.

Another technique is that a fading-memory average of the innovation from the estimator based on the quiescent model is computed as follows:

$$\rho(k)=\alpha\ \rho(k-1)+ \varepsilon_v(k) \qquad (4.4)$$

where $0<\alpha<1$. The manoeuvre is held to be taking place if $\rho(k)$ exceeds a certain threshold. This method provides faster but less reliable response.

4.3 Chan And Couture's Manoeuvre Detection

Chan and Couture (Chan and Couture, 1993) proposed a new manoeuvre detector. This manoeuvre detector is to project the target positions *s* intervals forward and then compare the projections against measurements.

Suppose that the target manoeuvre starts at *k* and continues on to *k+i*, results from unknown accelerations, and ignoring the process noise, the target position at *k+i* is

$$x(k+i) = x(k) + iT\dot{x}(k) + \eta(i) \tag{4.5}$$

where

$$\eta(i) = \sum_{j=0}^{i-1}[2(i-j)-1]\frac{T^2}{2}a_x(k+j) \tag{4.6}$$

If $a_x(k+j) = a_x =$ constant for all *j*, then

$$\eta(i) = \frac{(iT)^2}{2}a_x \tag{4.7}$$

Let the projections from the non-manoeuvring Kalman filter, at *k* for *k+i*, be

$$\breve{x}(k+i) = \hat{x}(k) + iT\hat{\dot{x}}(k) \tag{4.8}$$

where the estimate $\hat{x}(k)$ and $\hat{\dot{x}}(k)$ are generated by non-manoeuvring Kalman filter.

The position measurement at *k+i* is

$$\hat{x}(k+i) = x(k+i) + v_x(k+i) \tag{4.9}$$

so that the difference between the measurement and projection is

$$\begin{aligned} d_x(k+i) &= \hat{x}(k+i) - \breve{x}(k+i) \\ &= (x(k) - \hat{x}(k)) + iT(\dot{x}(k) - \hat{\dot{x}}(k)) + \eta(i) + v_x(k+i) \\ &= \tilde{x}(k) + iT\tilde{\dot{x}} + \eta(i) + v_x(k+i) \end{aligned} \tag{4.10}$$

Summing *s* of these differences at fixed *k* yields

$$\Lambda_x(s) = \sum_{i=1}^{s} d_x(k+i) = s\tilde{x}(k) + T\tilde{\dot{x}}\sum_{i=1}^{s} i + \sum_{i=1}^{s}\eta(i) + \sum_{i=1}^{s} v_x(k+i) \tag{4.11}$$

Let $\Lambda_{x0}(s)$ and $\Lambda_{x1}(s)$ denote, respectively, the values of $\Lambda_x(s)$ when there is no manoeuvre and when there is. If the errors terms $v_x(k+i)$ have zero mean and are mutually uncorrelated, i.e.

$$E\{v_x(m)v_x(n)\} = \sigma_x^2 \delta_{m,n} \tag{4.12}$$

then

$$E\{\Lambda_{x0}(s)\} = 0 \tag{4.13}$$

and

$$E\{\Lambda_{x1}(s)\} = \sum_{i=1}^{s} \eta(i) \tag{4.14}$$

Furthermore, the random variables $\Lambda_{x0}(s)$ and $\Lambda_{x1}(s)$ have identical covariances

$$V^2 = s^2 P_x + T^2 P_{\dot{x}} (\sum_{i=1}^{s} i)^2 + 2sT(\sum_{i=1}^{s} i) P_{x\dot{x}} + s\sigma_x^2 \tag{4.15}$$

$\Lambda_x(s)$ and its covariance can create a chi-square statistic

$$\varepsilon(s) = (\Lambda_x(s))^{\mathrm{T}} (V^2)^{-1} \Lambda_x(s) \tag{4.16}$$

which is used in a similar manner to the classical method to detect the manoeuvre. A detection variable for the y-direction can be obtained in a similar manner.

4.4 Chen And Norton's Manoeuvre Detection

Hägglund (1983) proposed a method to detect parameter changes in records with time-varying noise levels. The method operates on the sequence $\{\Delta\hat{\theta}\}$ of parameter corrections

$$\Delta\hat{\theta}_t = \hat{\theta}_t - \hat{\theta}_{t-1} \quad (t=1,2,\ldots) \tag{4.17}$$

The sequence is low-pass filtered to give {q} from

$$q_t = \alpha_1 q_{t-1} + \Delta\hat{\theta}_t, \ 0 < \alpha_1 < 1 \tag{4.18}$$

A scalar $s_t=\mathrm{sgn}(\Delta\hat{\theta}_t^{\mathrm{T}} q_{t-1})$

is then formed and low-pass filtered, yielding a statistic

$$r_t=\alpha_2 r_{t-1}+(1-\alpha_2)s_t,\ 0<\alpha_1<1 \qquad (4.19)$$

When r_t exceeds a threshold, a parameter change is inferred.

Chen and Norton (1987) proposed that rapid parameter changes be detected by a vector sequence based on Hägglund's method.

A low-pass filtered sequence {q} is formed by

$$q_t=\alpha_1 q_{t-1}+(1-\alpha_1)\Delta\hat{\theta}_t,\ \alpha_1\cong 0.9\sim 0.99 \qquad (4.20)$$

Noise-induced changes in $\hat{\theta}$ will fluctuate about zero and, if α_1 is well chosen, will contribute little to {q}. When a parameter θ_j changes abruptly, the parameter estimate will move systematically towards the new values. The corresponding element q_j of q will move away from zero and its trend will soon be detected by the statistic

$$r_{j,t}=\frac{|q_{j,t}|}{s_{j,t}} \qquad (4.21)$$

becoming comparable with unity, where

$$s_{j,t}=\alpha_1 s_{j,t-1}+(1-\alpha_1)|\Delta\hat{\theta}_{j,t}|\ (j=1,2,\ldots,n) \qquad (4.22)$$

The statistic is similar to the 'rambling factor' of Trulsson (Trulsson, 1983). As soon as a change in any parameter θ_j has been recognized, the estimated parameter error covariance is increased to obtain adaptive parameter estimation.

4.5 Weston And Norton's Manoeuvre Detection

The technique recommended by Weston and Norton (1997) scans a set of input-output records for evidence of change at any step to detect any impulses in the input, based on fixed-interval optimal smoothing.

The system is modeled by

$$\begin{cases} X(k+1) = FX(k) + GW(k) \\ Z(k) = H(k)X(k) + V(k) \quad k = 1,2,...,N \end{cases} \tag{4.23}$$

with $X \in \Re^n$, $W \in \Re^r$, $Z \in \Re^m$, $E[W(k)] = 0$, $E[V(k)] = 0$ and $\text{cov}[W(k)] = Q(k)$, $\text{cov}[V(k)] = R(k)$. An initial state estimate $\hat{X}(0|0)$ and its covariance $P(0|0)$ are also specified. For simplicity, there is assumed to be no deterministic forcing or correlation between $W(k)$ and $V(k)$. The forwards (filtered) estimate of $X(k)$, based on $\hat{X}(0|0)$ and the observations $Z(j)$, $j = 1,2,...,k$, is denoted by $\hat{X}(k|k)$, its prediction from the previous estimate by $\hat{X}(k|k-1)$ and their respective covariances by $P(k|k)$, $P(k|k-1)$. Correspondingly, the fixed-interval smoothed estimate at step *k*, based on observations through step *N*, where $N > k$, is denoted by $\hat{X}(k|N)$ and its covariance by $P(k|N)$.

The change-detection statistic will compare $\hat{X}(k|k-1)$ and $\hat{X}(k|N)$; it yields the probability that the difference

$$\delta(k) \equiv \hat{X}(k|k-1) - \hat{X}(k|N) \tag{4.24}$$

is consistent with its covariance, where the covariance is

$$\Delta(k) = \text{cov}[\delta(k)] = P(k|k-1) - P(k|N) \tag{4.25}$$

If $\delta(k)$ is assumed to be Gaussian (usually reasonable except perhaps near the ends of the records), the statistic

$$d(k) \equiv \delta^{\mathrm{T}}(k)\Delta^{-1}(k)\delta(k) \tag{4.26}$$

is a χ^2 variate with n degrees of freedom. An upper acceptance threshold for $d(k)$ is given directly by the lowest acceptable probability that $\delta(k)$ results by chance under the null hypothesis. If $d(k)$ fails the test, the failure is attributed to a state or input change.

4.6 Manoeuvre Detection By Testing The Statistic Of Normalised Squared Smoothed-Acceleration

We use the augmented-state (with position, speed, and acceleration) model throughout the process, with process noise covariance the same as the normal model, and fixed-lag smoothing is used to provide better estimates of the accelerations and their error covariance. The statistic and test based on smoothed estimated accelerations are:

$$\delta_a(j) = \hat{X}_a^{\mathrm{T}}(j \mid k)\, P_a^{-1}(j \mid k)\, \hat{X}_a(j \mid k) \tag{4.27}$$

$$\mathrm{P}\{\delta_a(j) \le \varepsilon_{\max}\} = 1-\alpha \tag{4.28}$$

where $\hat{X}_a(j \mid k)$ is the smoothed estimate of the acceleration components, $P_a(j \mid k)$ is the corresponding block from the smoothed covariance matrix, and $\delta_a(j)$ has a chi-square distribution with N_a (the dimension of acceleration) degrees of freedom. When the sum

$$\mu_a(j) = \sum_{i=j-p+1}^{j} \delta_a(i) \tag{4.29}$$

exceeds a specified threshold, the hypothesis that a manoeuvre is taking place is accepted, at which point the estimator switches from the quiescent (low-forcing) model to the manoeuvring model. If the accelerations drop rapidly to zero, the smoothed acceleration estimates will decrease in size monotonically. Thus, when the statistic falls twice in succession, the end of the manoeuvre is assumed and the estimator switches from the manoeuvre model to the quiescent model. A more conservative method would

be to apply the test to *p* successive values of the statistic with *p*>2, at the price of greater delay.

The manoeuvre sometimes results from a gradual change in acceleration, rather than a step. We suppose the acceleration change is at an unknown constant rate β. In a method similar to the above, we use an augmented state (with position, speed, acceleration and acceleration rate) model throughout the process, and fixed-lag smoothing is used to provide better estimates of the rate of acceleration and its estimate error covariance. The statistic and test based on smoothed rate of acceleration are:

$$\delta_\beta(j) = \hat{X}_\beta^{\mathrm{T}}(j \mid k) P_\beta^{-1}(j \mid k) \hat{X}_\beta(j \mid k) \tag{4.30}$$

$$\mathrm{P}\{\delta_\beta(j) \le \varepsilon_{\max}\} = 1\text{-}\alpha \tag{4.31}$$

where $\hat{X}_\beta(j \mid k)$ is the smoothed estimate of the acceleration rate components, $P_\beta(j \mid k)$ is the corresponding block from the smoothed covariance matrix, and $\delta_a(j)$ has a chi-square distribution with N_a (the dimension of acceleration rate) degrees of freedom. The statistic in a window is

$$\mu_\beta(j) = \sum_{i=j-p+1}^{j} \delta_\beta(i) \tag{4.32}$$

if $\mu_\beta(j)$ exceeds a certain threshold, the manoeuvre is held to be happening.

4.7 Modified Chen And Norton's Manoeuvre Detection

When the manoeuvre happens, the statistic of normalised squared innovations will increase in size rapidly; otherwise, in the absence of the manoeuvre, the changes in the statistic of normalised squared innovations will fluctuate about zero.

So the statistic of normailsed squared innovations is defined as the parameter $\hat{\theta}$

$$\hat{\theta}(i) \equiv v^{\mathrm{T}}(i) S^{-1}(i) v(i) \tag{4.33}$$

A low-pass filtered sequence {q} is formed by

$$q(i) = \alpha_1 q(i-1) + (1-\alpha_1)\Delta\hat{\theta}(i),\ \alpha_1 \cong 0.9 \sim 0.99 \tag{4.34}$$

When the changes in $\hat{\theta}$ fluctuates about zero and, if α_1 is well chosen, the changes will contribute little to $\{q\}$. When a parameter $\hat{\theta}$ changes abruptly, the parameter estimate will move systematically towards the new values. The corresponding element $q(i)$ will move away from zero and its trend will soon be detected by the statistic

$$r(i) = \frac{|q(i)|}{s(i)} \tag{4.35}$$

becoming comparable with unity, where

$$s(i) = \alpha_1 s(i-1) + (1-\alpha_1)\left|\Delta\hat{\theta}(i)\right|,\ (i = 1,2,...,n) \tag{4.36}$$

When the statistic exceeds a threshold, the manoeuvre is assumed.

4.8 Modified Weston And Norton's Manoeuvre Detection

The proposed manoeuvre detection is to use the fixed-lag smoothed estimate and its estimate error covariance instead of the fixed-interval smoothed estimate and the corresponding estimate error covariance in (Weston and Norton, 1997), respectively.

The change-detection statistic is to compare the forwards prediction $\hat{X}(j \mid j-1)$ with smoothed estimation $\hat{X}(j \mid k)$,

$$\delta(j) \equiv \hat{X}(j \mid k) - \hat{X}(j \mid j-1) \tag{4.37}$$

and suppose X is *n*-dimension state vectors.

The statistic

$$d(j) \equiv \delta^{\mathrm{T}}(j)[P(j|j-1) - P(j|k)]^{-1}\delta(j)\ d(j) \equiv \delta^{\mathrm{T}}(j)[\,P(j \mid j\text{-}1) - P(j \mid k)]^{-1} \tag{4.38}$$

is a χ^2 variate with *n* degrees of freedom. A threshold is set up based on the lowest acceptable probability with a conventional Kalman filter.

When the sum

$$\mu(j) = \sum_{i=j-p+1}^{j} d(i) \tag{4.39}$$

exceeds the threshold at step *j* over a window of length *p*, a manoeuvre may be assumed to have taken place, the statistic with manoeuvre filter is recomputed supposing the manoeuvre happens around step *j*, and the step at which the statistic provides the minimum value is regarded as the real manoeuvre step, i.e., the statistics $\mu_1(i)$, $\mu_2(i)$ and $\mu_3(i)$ at the same step *i* with manoeuvre filter, are recomputed using equation (4.39) supposing the manoeuvre happens at the step *j*-1, *j* and *j*+1 respectively, and the real manoeuvre step *t** is

$$t^* = \arg\min\{\mu_1(i) \mid t=j-1,\ \mu_2(i) \mid t=j,\ \mu_3(i) \mid t=j+1\} \tag{4.40}$$

4.9 Simulations

The simulations compare the classical manoeuvre detection, the manoeuvre detection methods by testing statistic of normalised squared-acceleration, modified Chen and Norton's manoeuvre detection, and modified Weston and Norton's manoeuvre detection by using the same example and the same observation sequence.

This example considers a target whose position is sampled every T = 1s in the two-dimension space. The target is on a constant course and speed until t = 40s, and starts to manoeuvre at t = 40s, and then completes to manoeuvre in 20 sampling period. The process noise covariance $Q = 10^{-2}$ and the measurement noises have 30m and 0.5 degrees standard deviation for the range and bearing, respectively. The initial conditions of the target are given by $x(0)$ = 2000m, $\dot{x}(0)$ = 0m/s, $y(0)$ = 10000m, and $\dot{y}(0)$ = -500m/s.

The manoeuvre is the result of the acceleration input as follows

u_x=u_y=35m/s^2, $40\text{s} \le t \le 60\text{s}$.

For the classical manoeuvre detection by testing the statistic of normalised squared-innovation. The window size is 5 sample intervals, and then the threshold is taken to be 18.3, given in equation (4.3) according to 0.95 confidence region for a 10-degrees-of-freedom chi-square distribution and the end of manoeuvre is also detected by the statistic of normalised innovations squared.

For the manoeuvre detection by testing the statistic of normalised squared smoothed-acceleration, the lag is 2 sample intervals, and the window size is 2. The threshold for $\mu_a(k)$, given in equation (4.29), is 9.49 according to 0.95 confidence region for a 4-degrees-of-freedom chi-square distribution.

For the modified Chen and Norton's method by testing the changes of normalised squared-innovation, the α_1 is 0.95 in equations (4.34) and (4.36), and the threshold is taken to be 0.4, given in equation (4.35).

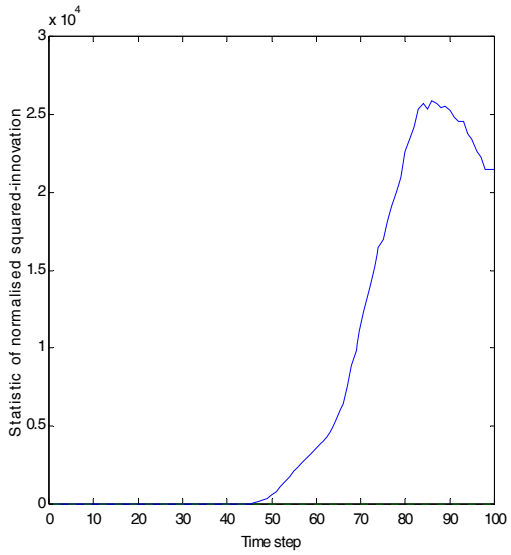


Fig. 4.1 Statistic of normalised squared-innovation

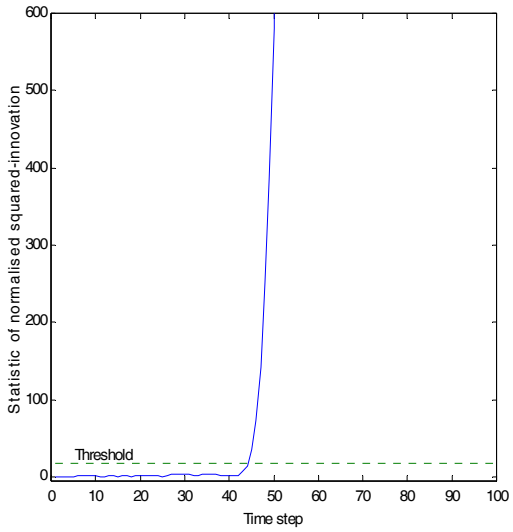


Fig. 4.2 Statistic of normalised squared-innovation (Amplification of Fig. 4.1).

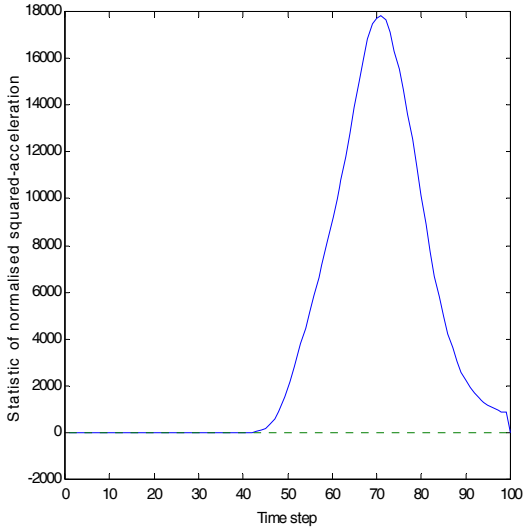


Fig. 4.3 Statistic of normalised squared-acceleration.

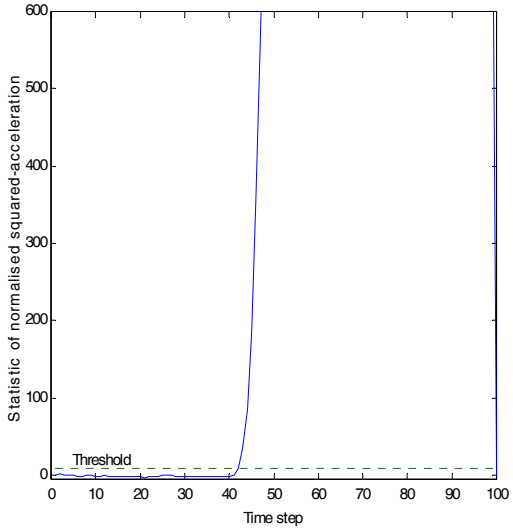


Fig. 4.4 Statistic of normalised squared-acceleration(Amplification of Fig. 4.3).

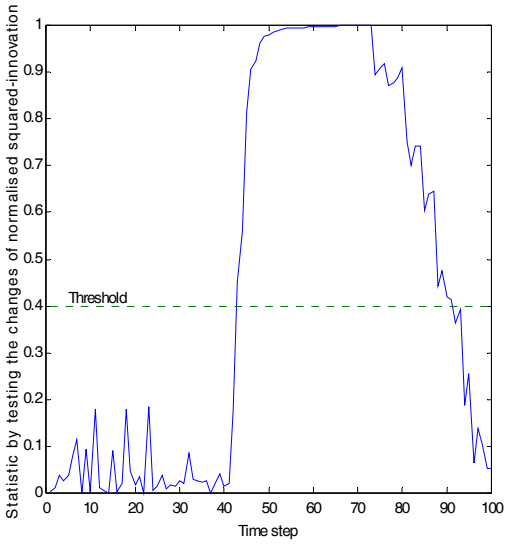


Fig. 4.5 Statistic of modified Chen and Norton's method.

In the modified Weston and Norton's manoeuvre detection, the fixed lag is 3 sample intervals, the sliding widow size is 2 sample intervals The threshold for $\mu_a(k)$, (given in equation (4.39) with p=2 and α=0.005), is 21.955 according to 0.995 confidence region for a 8-degrees-of-freedom chi-square distribution. After the manoeuvre is detected, the model switched into augmented state model with 6-state, the lag is still 3 sample intervals, and the sliding widow size is 2 sample intervals. The threshold for $\mu_a(k)$, (given in equation (4.39) with p=2 and α=0.005), is 28.299 according to 0.995 confidence region for a 12-degrees-of-freedom chi-square distribution.

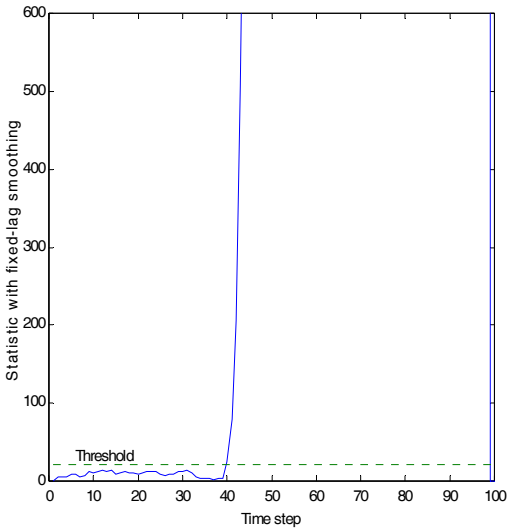


Fig. 4.6 Statistic with fixed-lag smoothing.

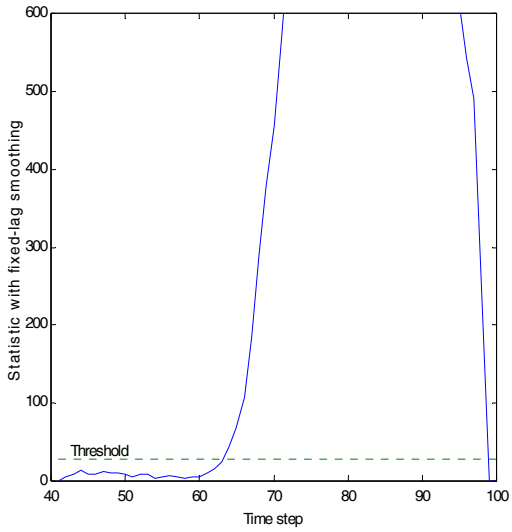


Fig. 4.7 Statistic with fixed-lag smoothing.

Each run gives the similar results. Thus Figures 4.1-4.7 only present one-run results of the manoeuvre detection by testing the statistic of normalised squared-innovation, by testing the statistic of normalised smoothed squared-acceleration, by using modified Chen and Norton's method, and by using modified Weston and Norton's method, respectively. Figures 4.1 and 4.2 show that the manoeuvre is detected at step 45, and that the end of the manoeuvre is difficult to detect. Figures 4.3 and 4.4 show that the manoeuvre is detected at step 42, and the end of manoeuvre is detected at step 73. Figure 4.5 shows that the manoeuvre is detected at step 43. Figure 4.6 shows that the manoeuvre is detected at step 40, and Figure 4.7 shows that the end of manoeuvre is detected at step 63.

From these results of simulation, the classical manoeuvre detection is shown to suffer much more delay than any of the proposed methods. The manoeuvre detection by modified Weston and Norton's method, based on fixed-lag smoothing, is much more effective and quicker although including the fixed lag (3 sample intervals).

4.10 Discussion And Summary

This chapter reviews and discusses many manoeuvre detection techniques in the literature in the last 30 years, and also develops several manoeuvre detection methods.

The classical manoeuvre detection by testing the normalised squared innovations is simple but suffers much delay or false alarm. Chan and Couture's manoeuvre still is susceptible to noise problems. A false alarm could be generated or the detection delayed. McAulay and Denlinger (1973) use matched filtering to detect manoeuvre; and Zhang and Li's , and Isaksson and Gustafsson's manoeuvre detection by observing the probability of each model using the IMM require the knowledge of the manoeuvre characteristics before detecting it, and seems difficult to tune to real situations.

Chen and Norton's rapid parameter changes detection, based on Hägglund's method, is used in adaptive parameter estimation and adaptive detecting rapid parameter changes, and effective. Weston and Norton's manoeuvre detection is to detect impulses in the input via full measurements with offline processing. Its advantage is simplicity and effectiveness. But unfortunately, it cannot be applied into the tracking with online processing.

Finally, we developed the following manoeuvre detection methods. The fixed-lag smoothing is used to provide better estimates of the accelerations and their error covariance, and then the statistic of normalised squared smoothed accelerations is used to provide quick detection. We also modified Weston and Norton's manoeuvre detection with the fixed-lag smoothing instead of the fixed-interval smoothing based on Weston and Norton's method, and make it be used in online processing to obtain the more accurate manoeuvre step and quicker detection. Because Chen and Norton's detection

can detect the rapid parameter changes and the feature of manoeuvre is that the statistic of normalised squared innovations increase in size abruptly, detecting the changes of the statistic of normailsed squared-innovation would give effective manoeuvre detection.

The classical manoeuvre detection is described by equation (4.3); Chan and Couture's manoeuvre detection by (4.16); Chen and Norton's manoeuvre detection by equations (4.20), (4.21) and (4.22); Weston and Norton's manoeuvre detection by equation (4.26); Manoeuvre detection by testing a normalised-squared-smoothed acceleration statistic by equations (4.27) and (4.29); Manoeuvre detection by testing the statistic of normalised-squared-smoothed-rate in acceleration variation by equations (4.30) and (4.32); Modified Chen and Norton's manoeuvre detection by equations (4.33), (4.34), (4.35) and (4.36); Modified Weston and Norton's manoeuvre detection by equations (4.37), (4.38), (4.39) and (4.40).

CHAPTER 5

MODIFIED SINGER'S FILTER FOR MULTIPLE-MODEL TRACKING FILTER

This chapter presents modified Singer's tracking filter. As mentioned in the preceding chapters, Singer's filter has a good performance for the target with low level manoeuvre, but a poor performance for the target with high level manoeuvre. Thus, this chapter uses the input estimation technique to estimate manoeuvre level, and then regards target accelerations as a perturbation around the estimated acceleration level to construct the Singer's filter. The final Singer's filter is called as modified Singer's filter in this thesis. The modified Singer's filter would track any level manoeuvring target well. This chapter also compares this method with the IMM and Singer's filter mentioned in the preceding chapters. The simulation results show how the improvement of the modified Singer's filter is, compared with Singer's filter. The simulation results also show the comparable performance and economic computational load of the modified Singer's filter, compared with IMM filter. Therefore, the modified Singer's filter is used to provide safeguards for the new multiple-model filter of Chapter 6 during uncertain manoeuvres.

5.1 Introduction

In several publications (Bar-Shalom,1989 and Lin,1993), the interacting multiple model (IMM) algorithm has been suggested as a very useful algorithm for the tracking of a manoeuvring target. The IMM algorithm is a suboptimal hybrid filter that has been shown to be one of the most cost-effective hybrid state estimation schemes. The main feature of this algorithm is its ability to estimate the state of a dynamic system with

several behaviour modes which can “switch” from one to another. In particular, the IMM filter can be a self-adjusting variable-bandwidth filter, which makes it natural for tracking manoeuvring targets. However, the IMM filter incorporates a large number of filters to cover the possible manoeuvre motions, resulting in heavy computation.

Input estimation (IE) algorithm (Chan et al., 1979; 1982) consists of the estimation of the unknown input (manoeuvre) over a sliding window and suitable compensation of the state estimate under the assumption that the manoeuvre onset time is the starting point of the sliding window. This technique shows poor performance in case of low level manoeuvring target because it tends to over-compensate for the manoeuvre. However, this technique has a good performance in case of high level manoeuvring target when the manoeuvre onset time is detected.

In 1970, Singer (1970) proposed a target model in which a manoeuvring motion is included as a Markov first order process with zero-mean and correction time constant τ. A tracking filter with Singer's model shows good performance for the target with a low level manoeuuvre, but its performance rapidly degrades in the case of high level manoeuvring target. The proposed modified filter is a filter which combines advantages of IE filter and Singer's filter. Low level manoeuvre is handled by Singer's filter while high level manoeuvre is detected and compensated by input estimation, i.e., high manoeuvre level is estimated by input estimation technique, and then Singer's filter is constructed by considering target accelerations as a perturbation around the estimated acceleration level, so in this way, the modified Singer’s filter should track any level manoeuvring target well.

This work uses the modified Singer's filter to provide the state estimation in the cases of uncertain manoeuvre and low level manoeuvre as part of the new multiple-model filter in Chapter 6.

Section 5.2 presents the modified Singer's filter in detail. Section 5.3 gives the simulation results. Finally, Section 5.4 summarizes this chapter.

5.2 Modified Singer's Tracking Filter

5.2.1 Target models

Singer proposed a manoeuvre model as a first-order Markov process with zero-mean and correlation time τ as follows,

$$\dot{\alpha}(t) = -\frac{1}{\tau}\alpha(t) + w(t) \qquad (5.1)$$

where $\alpha(t)$ is an acceleration and $w(t)$ is a zero-mean white noise with variance $E\{w^2(\tau)\} = q\delta(t-\tau)$. And process noise variance q is given by $q = 2\frac{\sigma_m^2}{\tau}$ where σ_m is a variance of target manoeuvre acceleration. In order to account an abrupt or high manoeuvre, we propose a new acceleration model, viz.,

$$a(t) = \alpha(t) + b_0 + (b - b_0)1(t - t_n) \qquad (5.2)$$

In (5.2), $\alpha(t)$ is the same variable modelled in(5.1), and it represents a slow, varying acceleration. The other portion of the right-hand side of (5.2) represents an abrupt change in acceleration, where $1(t)$ is a unit step function. And it is assumed that the manoeuvre level is changed from b_0 to b at time t_n. From (5.1) and (5.2), the dynamics of the acceleration $a(t)$ can be obtained by

$$\dot{a}(t) = -\frac{1}{\tau}\alpha(t) + w(t) + (b - b_0)\delta(t - t_n)$$

$$= -\frac{1}{\tau}a(t) + \frac{1}{\tau}\{b_0 + (b - b_0)1(t - t_n)\} + (b - b_0)\delta(t - t_n) + w(t) \qquad (5.3)$$

where $\delta(t)$ is a Dirac delta function. Defining a state vector $X(t)$ of position, velocity, and acceleration in two axes, the target dynamics model can be written by

$$\dot{X}(t) = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & I \\ 0 & 0 & -\frac{1}{\tau}I \end{bmatrix} X(t) + \begin{bmatrix} 0 \\ 0 \\ I \end{bmatrix} w(t) + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{\tau}I \end{bmatrix} [b_0 + (b - b_0)1(t - t_n)] + \begin{bmatrix} 0 \\ 0 \\ I \end{bmatrix} (b - b_0)\delta(t - t_n) \qquad (5.4)$$

where I is the 2x2 identity matrix. The discrete-time form of the above equation is sought for digital implementation. Let the sampling interval be *T*, and assume that $t_n = nT$. Define $X(k)$ to be a state vector at time $t = kT$. Similar to (Singer, 1970; Berg, 1983; Song et al, 1998), the discrete-time form of (5.4) can be obtained as follows,

$$X(k+1) = F[X(k) + B(b - b_0)\delta(k - n)] + G(b - b_0)1(k - n) + G\,b_0 + W(k)$$

where

$$\delta(k-n) = \begin{cases} 1; & k = n \\ 0; & \text{otherwise} \end{cases}, \qquad 1(k-n) = \begin{cases} 1; & k \geq n \\ 0; & \text{otherwise} \end{cases},$$

$$F = \begin{bmatrix} I & TI & \tau^2(-1 + \frac{T}{\tau} + e^{-\frac{T}{\tau}})I \\ 0 & I & \tau(1 - e^{-\frac{T}{\tau}})I \\ 0 & 0 & e^{-\frac{T}{\tau}}I \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ I \end{bmatrix}, \quad G = \begin{bmatrix} [-\tau T + \frac{T^2}{2} + \tau^2(1 - e^{-\frac{T}{\tau}})]I \\ [T + \tau(e^{-\frac{T}{\tau}} - 1)]I \\ [1 - e^{-\frac{T}{\tau}}]I \end{bmatrix}$$

and $W(k)$ is a zero-mean white noise vector with the same variance as in Chapter 2 of Singer's filter.

In this model, a high manoeuvre is represented as a transition of manoeuvre levels, while a low manoeuvre is modeled as a perturbation around the manoeuvre level. In

order to construct a filter using this model, a manoeuvre level change must be checked at every time stage, and a new manoeuvre level and its starting time should be estimated.

5.2.2 Manoeuvre level estimation

Consider a linear stochastic system of the form

$$X(k+1) = F[X(k) + B(b-b_0)\delta(k-n)] + G(b-b_0)1(k-n) + G\,b_0 + W(k) \quad (5.5)$$

$$Z(k) = HX(k) + V(k) \qquad (k \geq n)$$

Estimation of the state is done using the model without input b (model with manoeuvring level b_0):

$$X(k+1) = FX(k) + G\,b_0 + W(k) \quad (5.6)$$

From the innovation of the Kalman filter based on the model with manoeuvring level b_0 (5.6), the correction of input is to be detected, estimated and used to correct the state estimate.

Assume that the target starts new manoeuvring at time k. Its unknown correction of input during the time interval [k,…,k+s] are Δb_i, $i = k,...,k+s-1$. The state estimates from the (now mismatched) filter based on (5.6) will be denoted by an asterisk.

$$\hat{X}^*(i+1\,|\,i) = F[I - K(i)H]\,\hat{X}^*(i\,|\,i-1) + G\,b_0 + FK(i)Z(i)$$

$$\hat{X}^*(i+1\,|\,i) = \Phi(i)\,\hat{X}^*(i\,|\,i-1) + G\,b_0 + FK(i)Z(i) \quad (5.7)$$

where $i = k,...,k+s-1$

with the initial condition

$$\hat{X}^*(i\,|\,i-1) = \hat{X}(i\,|\,i-1) \quad (5.8)$$

being the correct estimate before the new manoeuvre started.

Recursion (5.7) yields, in terms of the initial condition (5.8)

$$\hat{X}^*(i+1|i) = [\prod_{j=k}^{i}\Phi(j)]\hat{X}(k|k-1) + \sum_{j=k}^{i}[\prod_{m=j+1}^{i}\Phi(m)][FK(i)Z(j) + Gb_0] \qquad (5.9)$$

where $i = k,...,k+s-1$

If the inputs were known, the correct filter based on (5.5) would yield estimates according to the recursion

$$\hat{X}(i+1|i) = [\prod_{j=k}^{i}\Phi(j)]\hat{X}(k|k-1) + [\prod_{j=k+1}^{i}\Phi(j)]FB(b-b_0) + \sum_{j=k}^{i}[\prod_{m=j+1}^{i}\Phi(m)][FK(j)Z(j) + Gb]$$

$$\hat{X}(i+1|i) = [\prod_{j=k}^{i}\Phi(j)]\hat{X}(k|k-1) + [\prod_{j=k+1}^{i}\Phi(j)]FB(b-b_0) + \sum_{j=k}^{i}[\prod_{m=j+1}^{i}\Phi(m)][FK(j)Z(j) + G(b-b_0) + Gb_0]$$

$$\hat{X}(i+1|i) = \hat{X}^*(i+1|i) + \{[\prod_{j=k+1}^{i}\Phi(j)]FB + \sum_{j=k}^{i}[\prod_{m=j+1}^{i}\Phi(m)]G\}(b-b_0) \qquad (5.10)$$

where $i = k,...,k+s-1$

Comparing equation (5.9) with (5.10), if the estimate $(b-b_0)$ is got , then the state estimate $\hat{X}(i+1|i)$ can be got. So, the problem is how to get $(b-b_0)$.

The innovations of equation (5.5) are

$$v(i+1) = Z(i+1) + H\hat{X}(i+1|i) \qquad (5.11)$$

The innovations of equation (5.6) are

$$v^*(i+1) = Z(i+1) + H\,\hat{X}^*(i+1|i) \qquad (5.12)$$

From equations (5.11) and (5.12), we can get

$$v^*(i+1) = H\{[\prod_{j=k+1}^{i}\Phi(j)]FB + \sum_{j=k}^{i}[\prod_{m=j+1}^{i}\Phi(m)]G\}(b-b_0) + v(i+1) \qquad (5.13)$$

and yields

$$v^*(i+1) = \Psi(i+1)(b-b_0) + v(i+1) \qquad (5.14)$$

where $\Psi(i+1) \equiv H\{[\prod_{j=k+1}^{i}\Phi(j)]FB + \sum_{j=k}^{i}[\prod_{m=j+1}^{i}\Phi(m)]G\}$ (5.15)

Equation (5.14) shows that the innovation $v^*(i+1)$ of the manoeuvring filter based on manoeuvre b_0 is a "linear measurement" of the correction of input $(b-b_0)$ in the presence of the additive "white noise" $v(i+1)$. From equation (5.14) it follows that the input can be estimated using the least-squares criterion from

$$Y = \Psi\ \Delta b + \varepsilon \tag{5.16}$$

$$\text{where } Y \equiv \begin{bmatrix} v^*(k+1 \\ \vdots \\ v^*(k+s) \end{bmatrix} \text{ and } \Psi \equiv \begin{bmatrix} \Psi(k+1) \\ \vdots \\ \Psi(k+s) \end{bmatrix} \tag{5.17}$$

are the stacked measurement vector and matrix, the noise

$$\varepsilon \equiv \begin{bmatrix} v(k+1) \\ \vdots \\ v(k+s) \end{bmatrix} \tag{5.18}$$

is zero-mean with block-diagonal covariance matrix

$$S = \text{diag}\{S(k+1),\ldots,S(k+s)\} \tag{5.19}$$

and the correction of input

$$\Delta b \equiv b - b_0$$

The estimation can be obtained in batch form as

$$\Delta\hat{b} = (\Psi^{\mathrm{T}} S^{-1}\Psi)^{-1}\Psi^{\mathrm{T}} S^{-1} Y \tag{5.20}$$

with the resulting covariance matrix

$$L = (\Psi^{\mathrm{T}} S^{-1}\Psi)^{-1} \tag{5.21}$$

After the correction of input is estimated, the state has to be corrected as follows:

$$\hat{X}(i+1\,|\,i) = \hat{X}^*(i+1\,|\,i) + M\Delta\hat{b} \tag{5.22}$$

$$\text{where } M \equiv [\prod_{j=k+1}^{i} \Phi(j)]FB + \sum_{j=k}^{i} [\prod_{m=j+1}^{i} \Phi(m)]G$$

The state $\hat{X}(i+1\,|\,i+1)$ also can be corrected by the estimated input and Kalman filter.

The covariance associated with the estimate (5.22) is

$$P(i+1|i) = P^*(i+1|i) + ML\,M^{\mathrm{T}} \tag{5.23}$$

where $P^*(i+1|i)$ is the covariance associated with estimate $\hat{X}^*(i+1|i)$.

5.3 Simulations

The simulations were done on a Intel 82371AB Pentium II processor and written in MATLAB with version 5.3. MATLAB is a technical computing environment for high-performance numeric computation and visualization. MATLAB integrates numerical analysis, matrix computation, signal processing, and graphics in an easy-to-use environment where problems and solutions are expressed just as they are written mathematically.

The performance of target trackers relies on their estimation accuracy and how fast they do the calculations. The accuracy is evaluated through the root-mean-square (rms) estimation error for position, speed and acceleration.

$$\text{rms position error}_{\text{sample}\,k} \equiv \tilde{p}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{x}_k^i - x_k^i)^2 + (\hat{y}_k^i - y_k^i)^2)}{N}}$$

$$\text{rms speed error}_{\text{sample}\,k} \equiv \tilde{v}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{\dot{x}}_k^i - \dot{x}_k^i)^2 + (\hat{\dot{y}}_k^i - \dot{y}_k^i)^2)}{N}}$$

$$\text{rms acceleration error}_{\text{sample}\,k} \equiv \tilde{a}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{a}_{nk}^i - a_{nk}^i)^2 + (\hat{a}_{tk}^i - a_{tk}^i)^2)}{N}}$$

$$\text{or rms acceleration error}_{\text{sample}\,k} \equiv \tilde{a}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{a}_{x(k|k)}^i - a_{xk}^i)^2 + (\hat{a}_{y(k|k)}^i - a_{yk}^i)^2)}{N}}$$

where $\hat{x}_k^i$ is the estimate of x at time k in simulation run number i; the actual value is x_k^i. The computational load is evaluated using their executing time of CPU work.

For each test, a Monte Carlo test of N=100 runs was performed.

The following gives three typical examples to compare modified Singer's filter with Singer's filter and IMM filter in performance and computation load. One is the commonest case of a simple target with rectilinear acceleration motion, the other two are practical cases: a circular motion target and an agile target.

5.3.1 Comparison of trackers using a simple target with rectilinear acceleration motion

The first example is the common case, the target's trajectory is in a constant-velocity, and then has an acceleration motion to a new velocity, and finally executes the new constant-velocity motion.

The example considers a target which begins 15km North and 20km West of the tracker travelling South at 600m/s in the two-dimension space. After 39 seconds of constant-velocity motion (subject to acceleration process noise of variance $\sigma_{a_x}^2 = \sigma_{a_y}^2 = 0.01$) the target executes a left turn with a lateral acceleration of 20m/s^2 for 31 seconds, and then returns to constant-velocity motion. Measurements are taken at a rate of 1Hz for 110 seconds with additive Gaussian noise of standard deviation σ_r=30m and σ_θ=0.5^o from an observer at the origin.

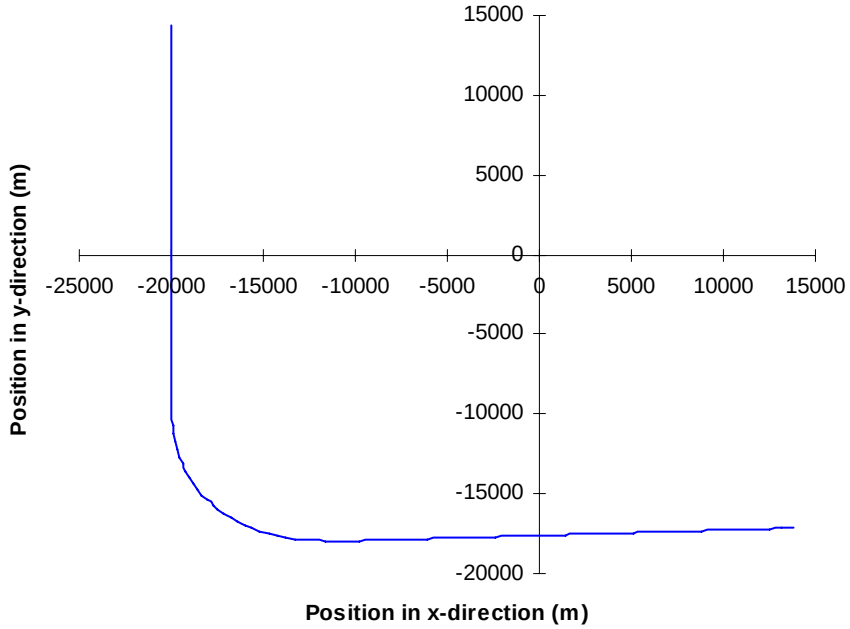


Fig. 5.1 Target trajectory

Three trackers are tested here,

tracker	details of tracker
Singer's Filter	manoeuvre correction time: 10s manoeuvre acceleration variance for target model: 3^2
Modified Singer's Filter	manoeuvre level estimation: input estimation manoeuvre detection: testing the statistic of normalised squared innovations window size: 5 samples target accelerations around estimated manoeuvre level: modelled by Singer's model manoeuvre correction time and manoeuvre acceleration variance: the same as for Singer's filter.
IMM	x-direction accelerations used: $a_x=\{\pm 50.0,\pm 25.0,0.0\}\text{m/s}^2$ y-direction accelerations used: $a_y=\{\pm 50.0,\pm 25.0,0.0\}\text{m/s}^2$ with a probability of no change in manoeuvre of 0.9.

Table 5.1 Trackers used in simulation comparison

The filters do not respond to initial conditions sensitively (Brown and Hwang, 1992), therefore, the initial conditions are approximately obtained as follows.

For IMM filter, the initial estimates of the states (Bar-Shalom and Birmiwal, 1982) are

$$\hat{x}(0 \mid 0) = Z_x(0) = x(0) + V_x(0) = r_m(0)\cos\theta_m(0)$$

$$\hat{y}(0 \mid 0) = Z_y(0) = y(0) + V_y(0) = r_m(0)\sin\theta_m(0)$$

$$\hat{\dot{x}}(0 \mid 0) = \frac{Z_x(0) - Z_x(-1)}{T} = \frac{r_m(0)\cos\theta_m(0) - r_m(-1)\cos\theta_m(-1)}{T}$$

$$\hat{\dot{y}}(0 \mid 0) = \frac{Z_y(0) - Z_y(-1)}{T} = \frac{r_m(0)\sin\theta_m(0) - r_m(-1)\sin\theta_m(-1)}{T}$$

where $Z_x(-1) = x(0) - \dot{x}(0)T + V_x(-1)$ and $Z_y(-1) = y(0) - \dot{y}(0)T + V_y(-1)$

$$P(0 \mid 0) = \begin{bmatrix} R_L^{11}(0) & R_L^{12}(0) & \frac{R_L^{11}(0)}{T} & \frac{R_L^{12}(0)}{T} \\ R_L^{12}(0) & R_L^{22}(0) & \frac{R_L^{12}(0)}{T} & \frac{R_L^{22}(0)}{T} \\ \frac{R_L^{11}(0)}{T} & \frac{R_L^{12}(0)}{T} & \frac{R_L^{11}(0) + R_L^{11}(-1)}{T^2} & \frac{R_L^{12}(0) + R_L^{12}(-1)}{T^2} \\ \frac{R_L^{12}(0)}{T} & \frac{R_L^{22}(0)}{T} & \frac{R_L^{12}(0) + R_L^{12}(-1)}{T^2} & \frac{R_L^{22}(0) + R_L^{22}(-1)}{T^2} \end{bmatrix}$$

For Singer's filter and modified Singer's filter, the initial estimates of the states (Singer, 1970) are

$$\hat{x}(0 \mid 0) = Z_x(0) = x(0) + V_x(0) = r_m(0)\cos\theta_m(0)$$

$$\hat{y}(0 \mid 0) = Z_y(0) = y(0) + V_y(0) = r_m(0)\sin\theta_m(0)$$

$$\hat{\dot{x}}(0 \mid 0) = \frac{Z_x(0) - Z_x(-1)}{T} = \frac{r_m(0)\cos\theta_m(0) - r_m(-1)\cos\theta_m(-1)}{T}$$

$$\hat{\dot{y}}(0 \mid 0) = \frac{Z_y(0) - Z_y(-1)}{T} = \frac{r_m(0)\sin\theta_m(0) - r_m(-1)\sin\theta_m(-1)}{T}$$

$$\hat{\ddot{x}}(0 \mid 0) = 0$$

$$\hat{\ddot{y}}(0 \mid 0) = 0$$

$$P(0\mid 0)=\begin{bmatrix} R_L^{11}(0) & R_L^{12}(0) & \frac{R_L^{11}(0)}{T} & \frac{R_L^{12}(0)}{T} & 0 & 0 \\ R_L^{12}(0) & R_L^{22}(0) & \frac{R_L^{12}(0)}{T} & \frac{R_L^{22}(0)}{T} & 0 & 0 \\ \frac{R_L^{11}(0)}{T} & \frac{R_L^{12}(0)}{T} & \frac{R_L^{11}(0)+R_L^{11}(-1)}{T^2} & \frac{R_L^{12}(0)+R_L^{12}(-1)}{T^2} & 0 & 0 \\ \frac{R_L^{12}(0)}{T} & \frac{R_L^{22}(0)}{T} & \frac{R_L^{12}(0)+R_L^{12}(-1)}{T^2} & \frac{R_L^{22}(0)+R_L^{22}(-1)}{T^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where $R_L^{11}=r_m^2\sigma_\theta^2\sin^2\theta_m+\sigma_r^2\cos^2\theta_m$

$R_L^{22}=r_m^2\sigma_\theta^2\cos^2\theta_m+\sigma_r^2\sin^2\theta_m$

and $R_L^{12}=(\sigma_r^2-r_m^2\sigma_\theta^2)\sin\theta_m\cos\theta_m$ (presented in Section 3.8)

The simulation results are shown in Figs 5.2-5.7. Figs 5.2-5.4 show the performance of modified Singer's filter and Singer's filter in position, speed and acceleration. Figs 5.5-5.7 show the performance of modified Singer's filter and IMM filter in position, speed and acceleration.

Table 5.2 shows the computation loads of trackers used

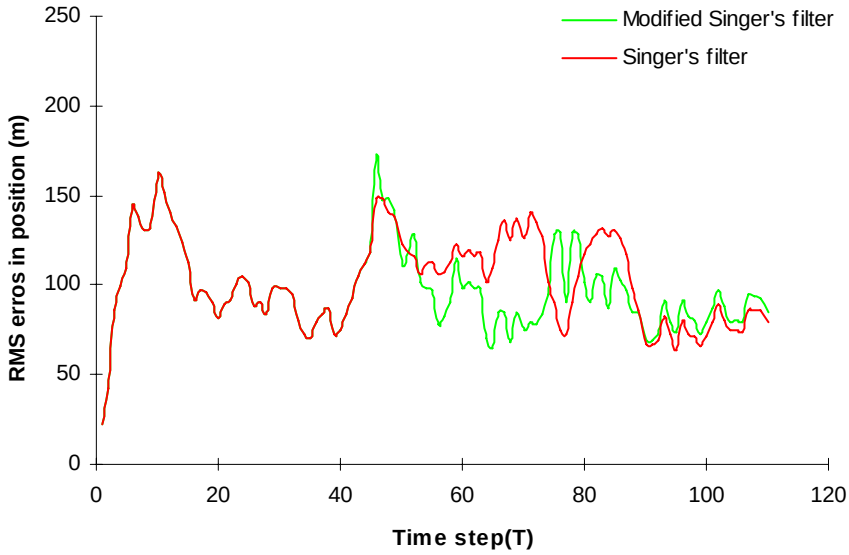


Fig. 5.2 RMS errors of modified Singer's filter and Singer's filter in position

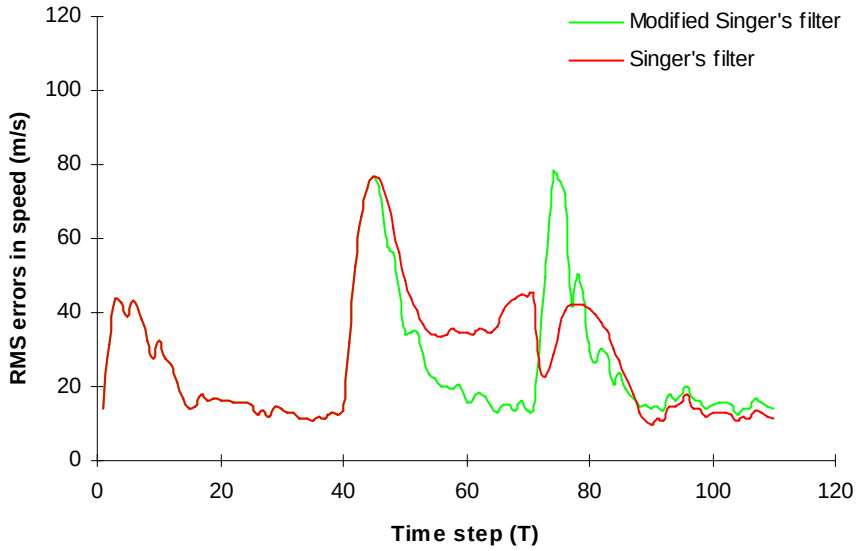


Fig. 5.3 RMS errors of modified Singer's filter and Singer's filter in speed

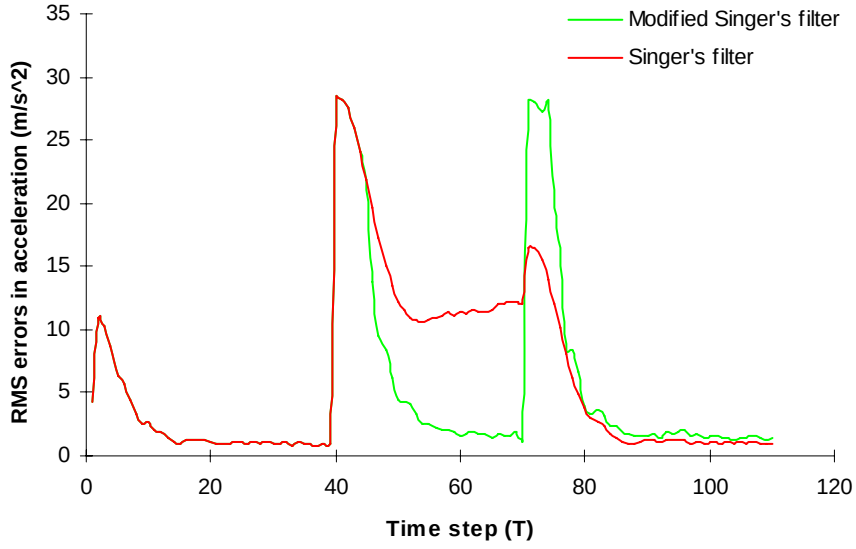


Fig. 5.4 RMS errors of modified Singer's filter and Singer's filter in acceleration

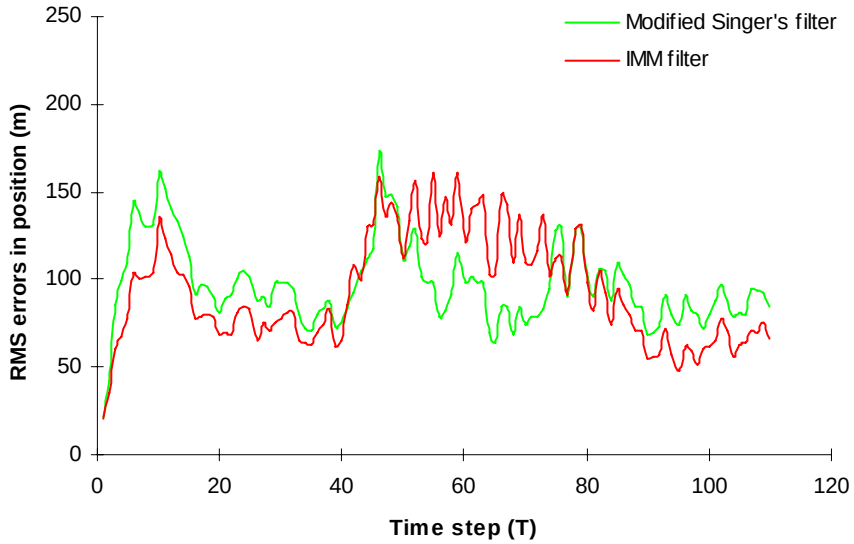


Fig. 5.5 RMS errors of modified Singer's filter and IMM filter in position

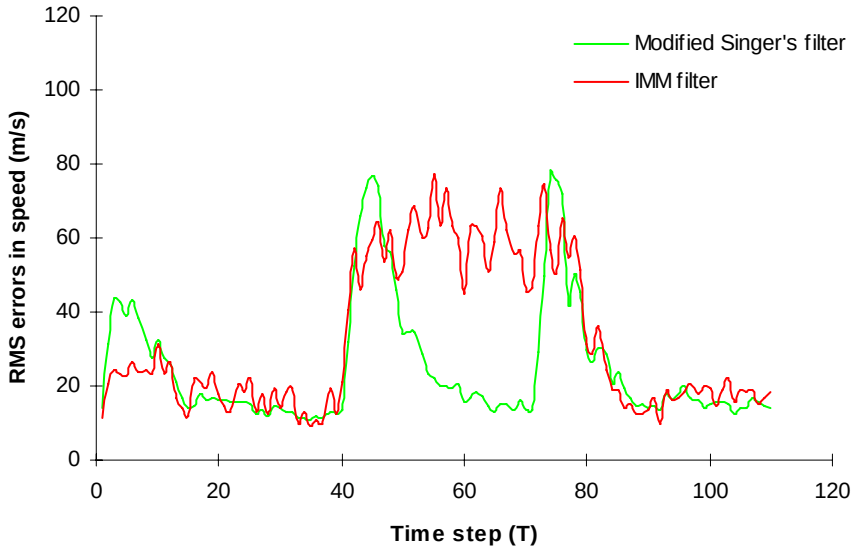


Fig. 5.6 RMS errors of modified Singer's filter and IMM filter in speed

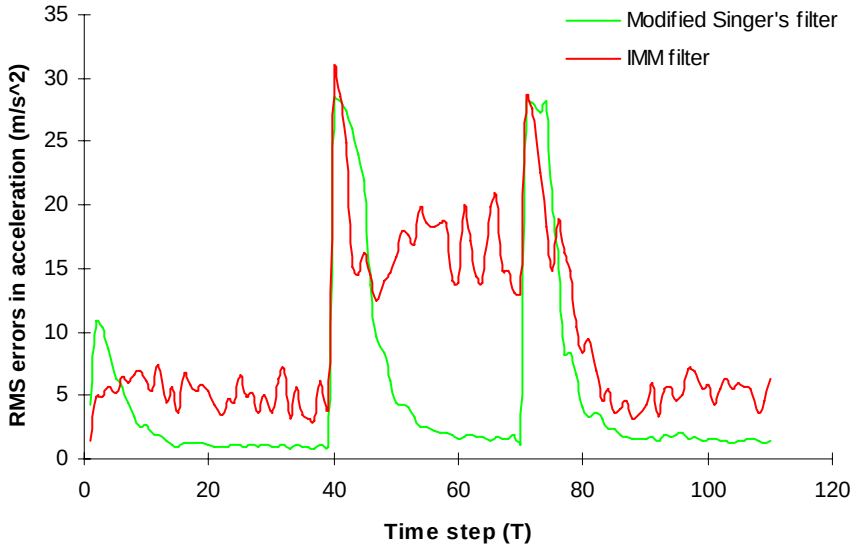


Fig. 5.7 RMS errors of modified Singer's filter and IMM filter in acceleration

Figs 5.2-5.4 show that the modified Singer's filter gives better performance than Singer's filter during the manoeuvre, and also show that the modified Singer's filter provides a little worse performance than Singer's filter after the manoeuvre ends. These figures also show that the process noise variance given in Singer's filter has not covered the manoeuvre well in the present of manoeuvre. In other hand, these figures demonstrate the improvement of the modified Singer's filter during the manoeuvre. These figures also demonstrate that the performance of modified Singers filter is poorer than Singer's filter after the manoeuvre ends because of the error of input estimation.

Figs 5.5-5.7 show that the modified Singer's filter gives better performance than IMM filter during the manoeuvre, and also show that the modified Singer's filter provides a little worse performance than IMM filter in the absence of manoeuvre. In the absence of manoeuvre, the filter with zero accelerations dominates in IMM, which estimates the probability of each model. In the presence of manoeuvre, the tracking is performed by five filters (one filter with zero in x-direction and 25m/s^2 in y-direction, one filter with

zero in *x*-direction and 50m/s^2 in *y*-direction, one filter with 25m/s^2 in *x*-direction and zero in *y*-direction, one filter with 50m/s^2 in *x*-direction and zero in *y*-direction, and another filter with 25m/s^2 in *x*-direction and in *y*-direction) in IMM. However, the modified Singer's filter in fact increases the process noise through the whole processing to worsen the performance than IMM in the absence of manoeuvre, and is compensated by input estimation and handles better than IMM during the manoeuvre.

Comparing the computation times of trackers

	Computation time(s)
Singer's Filter	0.164
Modified Singer's Filter	1.022
IMM	14.520

Table 5.2 Computation loads of trackers used in simulation comparison

5.3.2 Comparison of trackers with a circular motion target

The second example is another typical case, the target's trajectory is kept in a constant-speed motion, the track contains a constant-speed turn and two constant velocities motion.

The example considers a target which begins 25km South and 9km East West of the tracker travelling East at 500m/s in the two-dimension space. After 74 seconds of constant-velocity motion (subject to acceleration process noise of variance $\sigma_{a_n}^2 = \sigma_{a_t}^2 = 0.01$) the target executes a right circular turn with an acceleration of 30m/s^2 for 52 seconds and then returns to a constant-velocity motion. Measurements are taken at a rate of 1Hz for 200 seconds with additive Gaussian noise of standard deviation σ_r=30m and σ_θ=0.5^o.

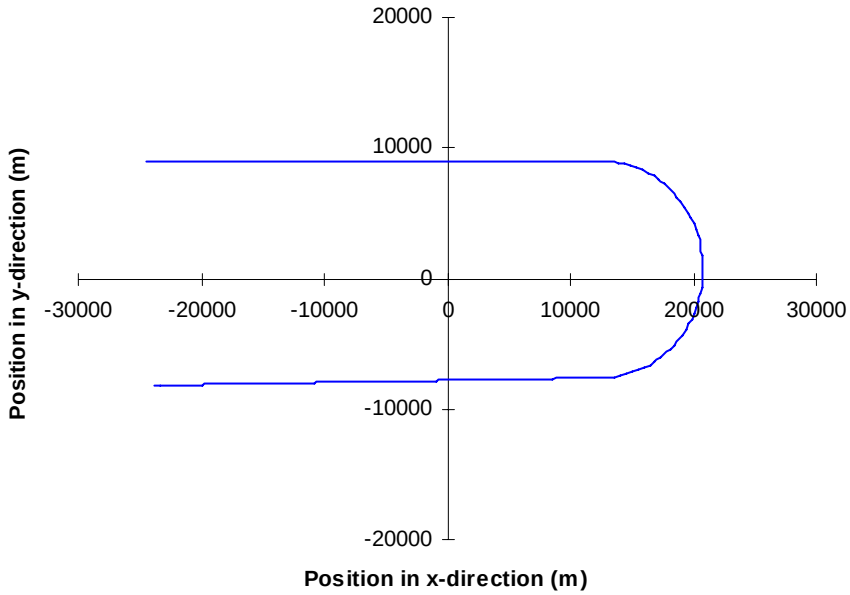


Fig. 5.8 Target trajectory

Three trackers are tested here,

tracker	details of tracker
Singer's Filter	manoeuvre correction time: 10s manoeuvre acceleration variance for target model: 3^2
Modified Singer's Filter	manoeuvre level estimation: input estimation manoeuvre detection: testing the statistic of normalised squared innovations window size: 5 samples target accelerations around estimated manoeuvre level: modelled by Singer's model manoeuvre correction time and manoeuvre acceleration variance: the same as for Singer's filter.
IMM	along-track accelerations used: a_t={±15.0,±7.5.0,0.0}m/s^2 cross-track accelerations used: a_n={±50.0,±25.0,0.0}m/s^2 with a probability of no change in manoeuvre of 0.9.

Table 5.3 Trackers used in simulation comparison

The initialization of the filters are the same as in the first example. The detail of initialization is not reiterated here.

The simulation results are shown in Figs 5.9-5.14. Figs 5.9-5.11 show the performance of modified Singer's filter and Singer's filter in position, speed and acceleration. Figs 5.12-5.14 show the performance of modified Singer's filter and IMM filter in position, speed and acceleration.

Table 5.4 shows the computation loads of trackers used.

Figs 5.9-5.11 show that the modified Singer's filter gives better performance than Singer's filter during the manoeuvre, and also show that the modified Singer's filter provides a little worse performance than Singer's filter after the manoeuvre ends. These simulation results are similar to the results in the first example with a rectilinear acceleration motion (Section 5.3.1).

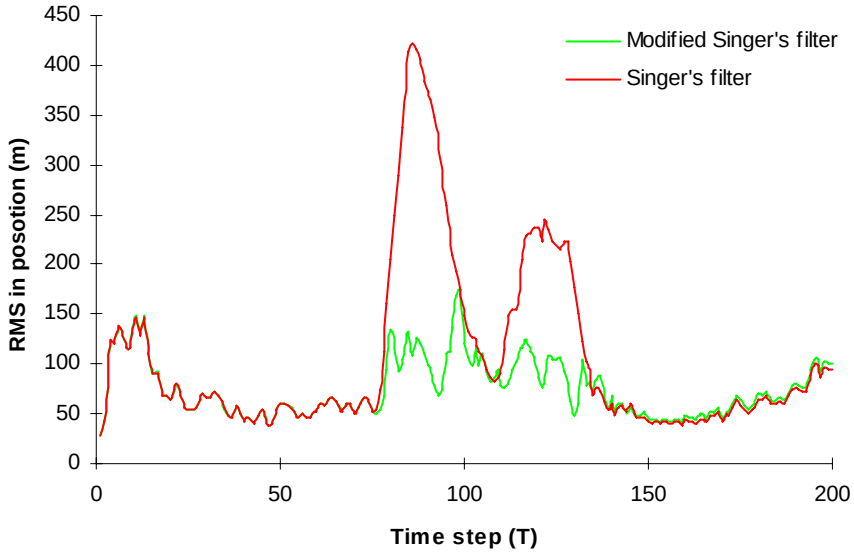


Fig. 5.9 RMS errors of modified Singer's filter and Singer's filter in position

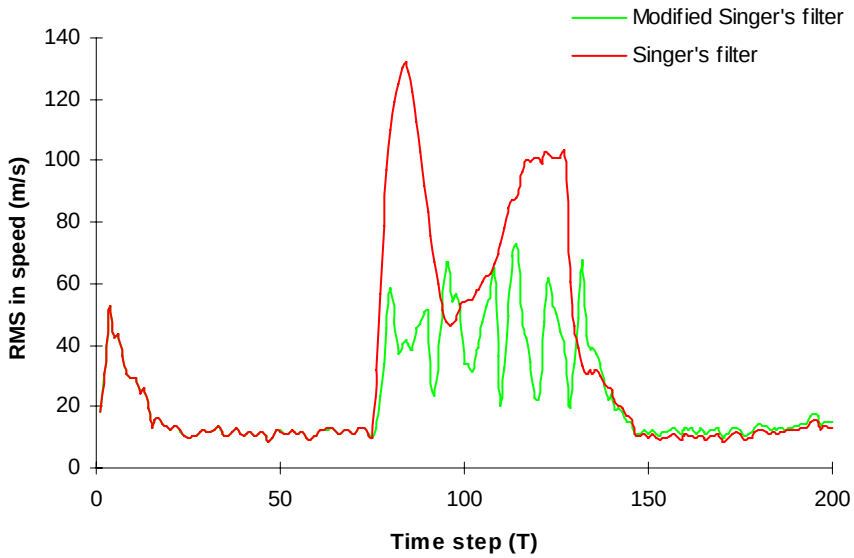


Fig. 5.10 RMS errors of modified Singer's filter and Singer's filter in speed

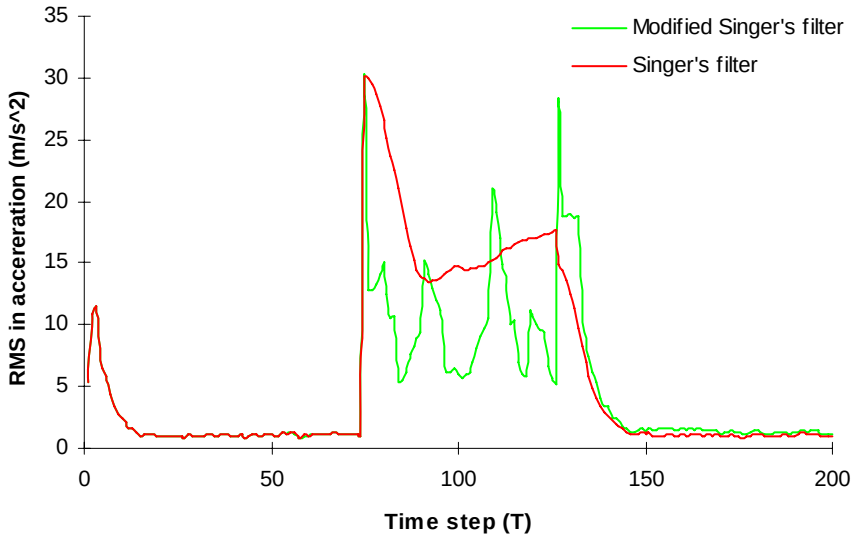


Fig. 5.11 RMS errors of modified Singer's filter and Singer's filter in acceleration

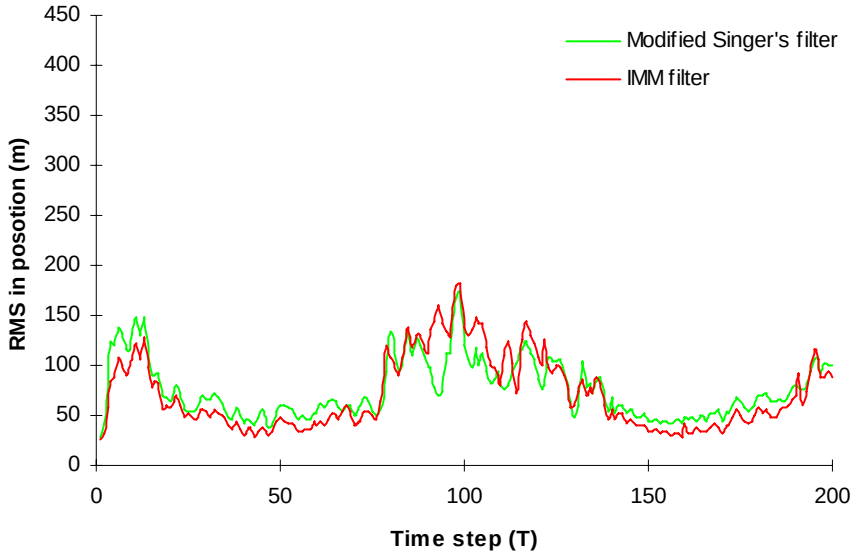


Fig. 5.12 RMS errors of modified Singer's filter and IMM filter in position

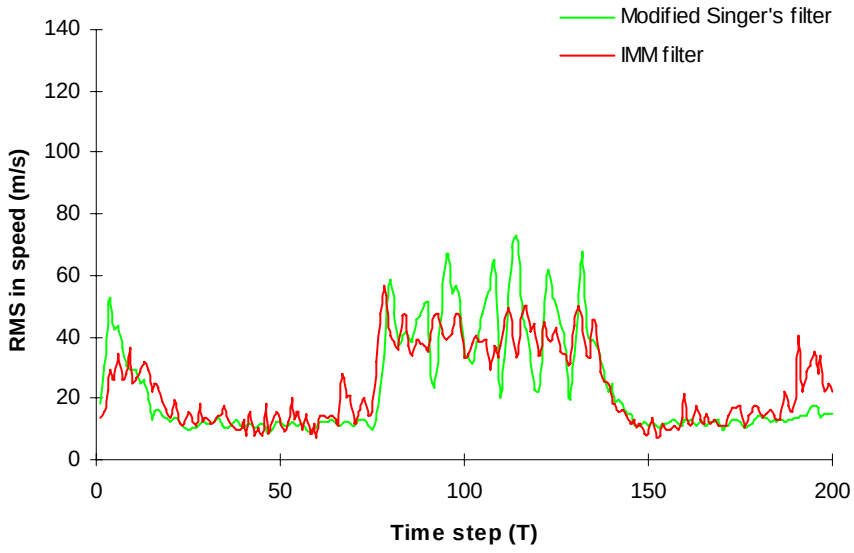


Fig. 5.13 RMS errors of modified Singer's filter and IMM filter in speed

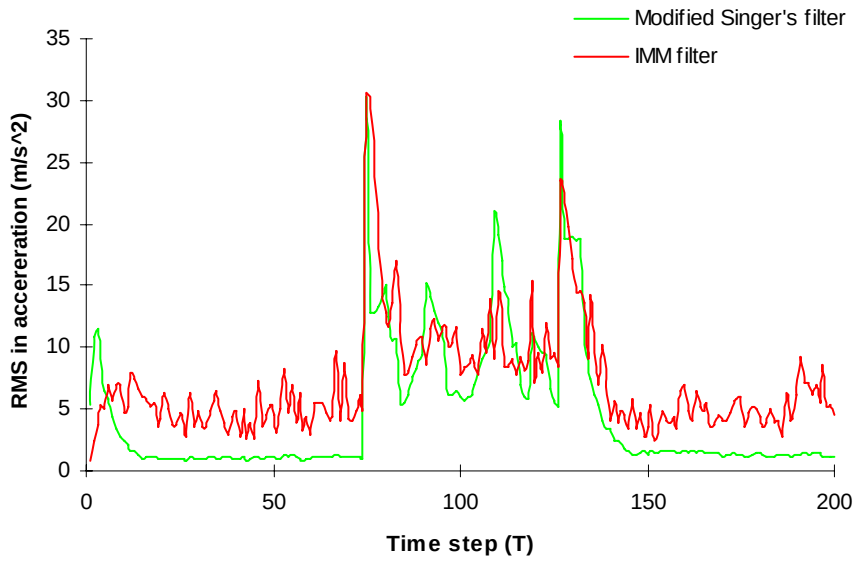


Fig. 5.14 RMS errors of modified Singer's filter and IMM filter in acceleration

Figs 5.12-5.14 show that the modified Singer's filter gives a little worse performance than IMM filter in the absence of manoeuvre, and comparable performance with IMM filter in the presence of manoeuvre. In the absence of manoeuvre, the filter with zero accelerations dominates for most of time for IMM tracking. But sometimes, the zero acceleration filter, along with some non-zero acceleration filters, dominates the IMM filter according to the probability of each model. In the presence of manoeuvre, the tracking is performed mainly by ten filters (one filter with zero in along-track and $25m/s^2$ in cross-track, one filter with zero in along-track and $50m/s^2$ in cross-track, one filter with $7.5m/s^2$ in along-track and $25m/s^2$ in cross-track, one filter with $7.5m/s^2$ in along-track and $50m/s^2$ in cross-track, and so on) in IMM. However, the modified Singer's filter in fact increases the process noise through the whole processing and is compensated with input estimation during the manoeuvre, and handles it better than IMM or gives a performance comparable with IMM filter.

Comparing the computation times of trackers

	Computation time(s)
Singer's Filter	0.494
Modified Singer's Filter	2.504
IMM	29.080

Table 5.4 Computation loads of trackers used in simulation comparison

5.3.3 Comparison of trackers with an agile target

The third example is more complex case, the target's trajectory consists of several constant velocities, constant-speed turns and an inconstant-speed turn motion.

The example considers a target which begins 15km South and 7.5km West of the tracker travelling East at 400m/s in the two-dimension space. After 149 samples of constant-velocity motion (subject to acceleration process noise of variance $\sigma_{a_n}^2 = \sigma_{a_t}^2 = 0.01$) the target sustains a manoeuvre or nonmanoeuvre every 60 samples. At first, the target executes a left circular turn with an acceleration of 75m/s^2 and then returns to a constant-velocity motion before entering a 75m/s^2 right turn. The target continues at constant velocity before turning left at 3.5*g* with the along-track acceleration 2g. The target now stop accelerating. Measurements are taken at a rate of 4Hz for 510 samples with additive Gaussian noise of standard deviation σ_r=30m and σ_θ=0.5^o.

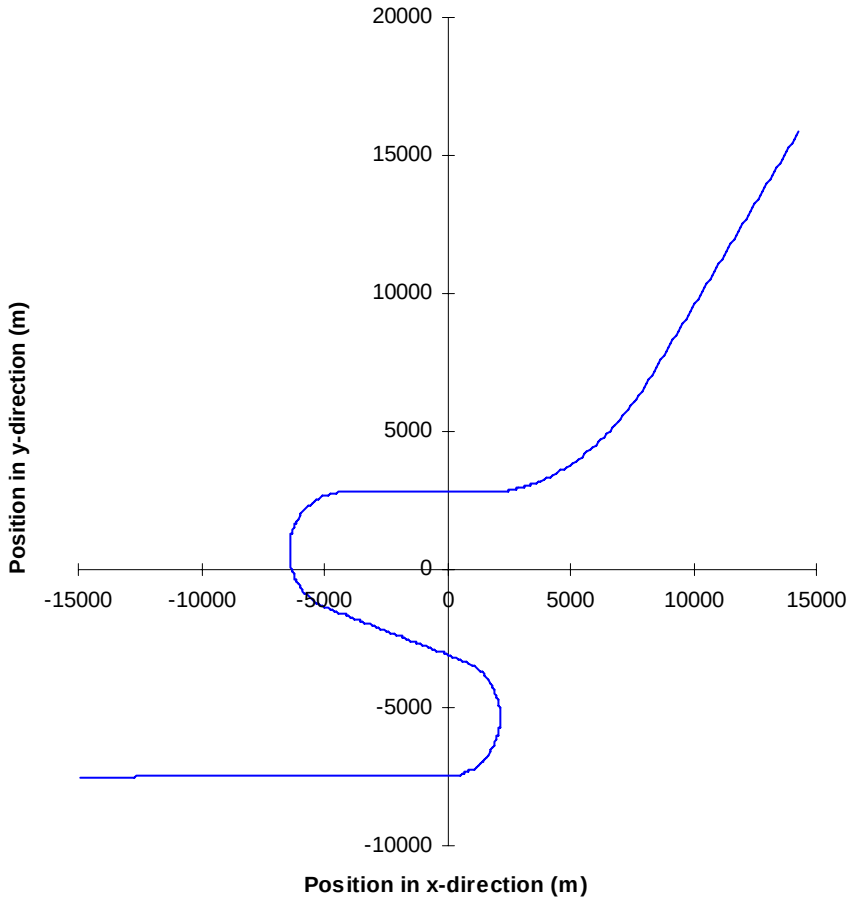


Fig. 5.15 Target trajectory

Three trackers are tested here,

tracker	details of tracker
Singer's Filter	manoeuvre correction time: 10s manoeuvre acceleration variance for target model: 3^2
Modified Singer's Filter	manoeuvre level estimation: input estimation manoeuvre detection: testing the statistic of normalised squared innovations window size: 5 samples target accelerations around estimated manoeuvre level: modelled by Singer's model manoeuvre correction time and manoeuvre acceleration variance: the same as for Singer's filter.
IMM	along-track accelerations used: $a_t=\{\pm 30.0,\pm 15.0,0.0\}\text{m/s}^2$ cross-track accelerations used: $a_n=\{\pm 100.0,\pm 50.0,0.0\}\text{m/s}^2$ with a probability of no change in manoeuvre of 0.9.

Table 5.5 Trackers used in simulation comparison

The initialization of the filters are the same as in the first example. The detail of initialization is not reiterated here.

The simulation results are shown in Figs 5.16-5.21. Figs 5.16-5.18 show the performance of modified Singer's filter and Singer's filter in position, speed and acceleration. Figs 5.19-5.21 show the performance of modified Singer's filter and IMM filter in position, speed and acceleration.

Table 5.6 shows the computation loads of trackers used.

Figs 5.16-5.18 show that the modified Singer's filter gives better performance than Singer's filter. These figures prove that the process noise variance given in Singer's filter has not covered the manoeuvre well in the present of manoeuvre and Singer's filter has much delay to be adjusted to non-manoeuvre target in the absence of manoeuvre.

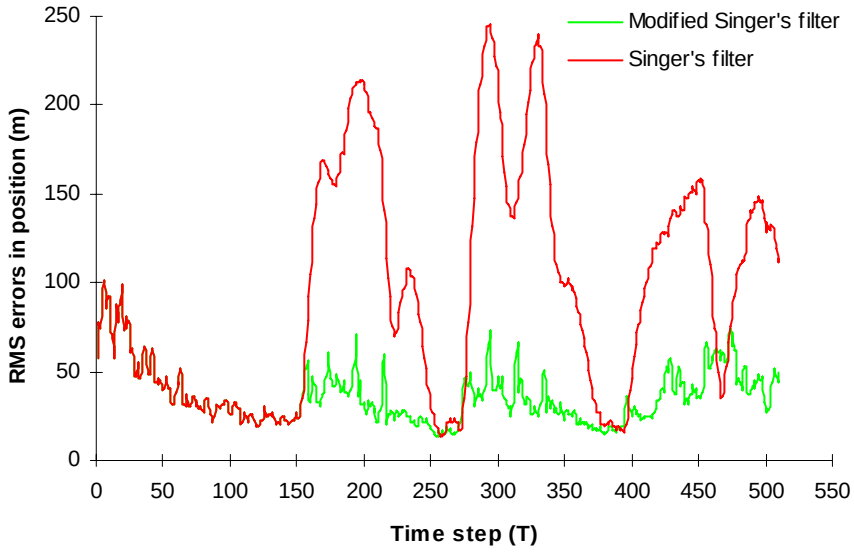


Fig. 5.16 RMS errors of modified Singer's filter and Singer's filter in position

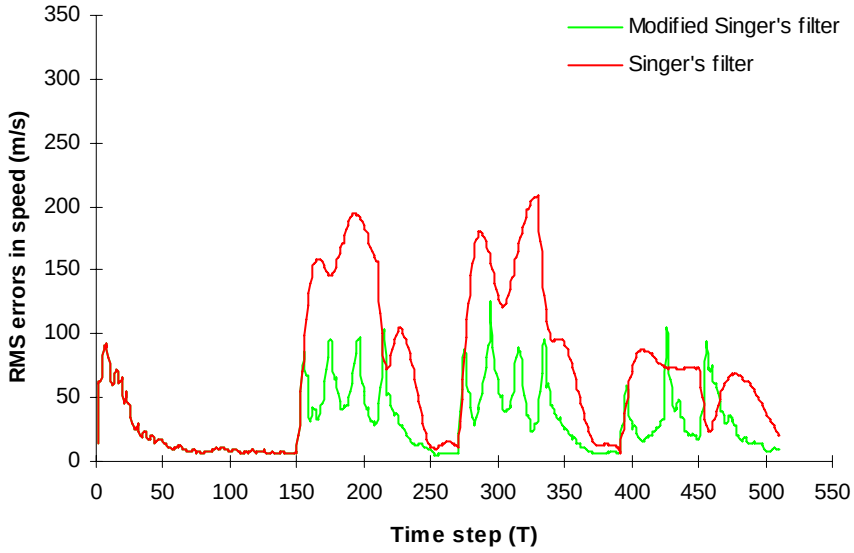


Fig. 5.17 RMS errors of modified Singer's filter and Singer's filter in speed

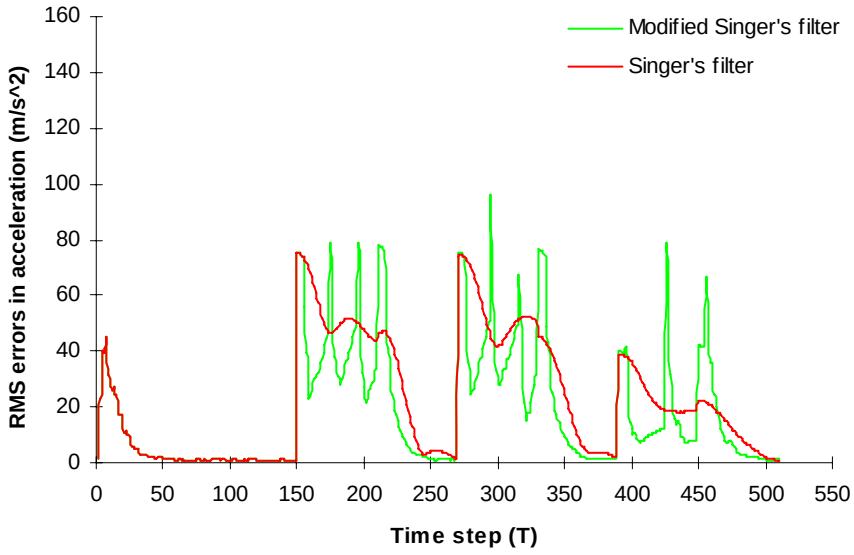


Fig. 5.18 RMS errors of modified Singer's filter and Singer's filter in acceleration

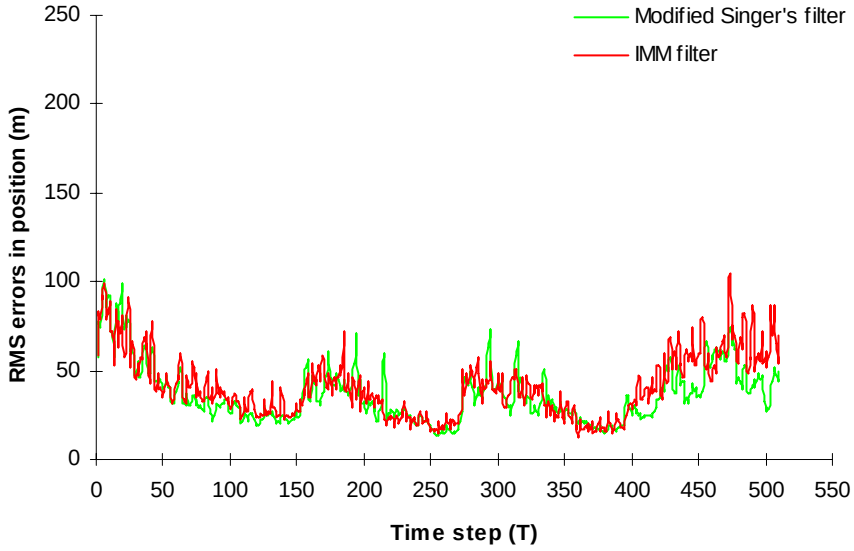


Fig. 5.19 RMS errors of modified Singer's filter and IMM filter in position

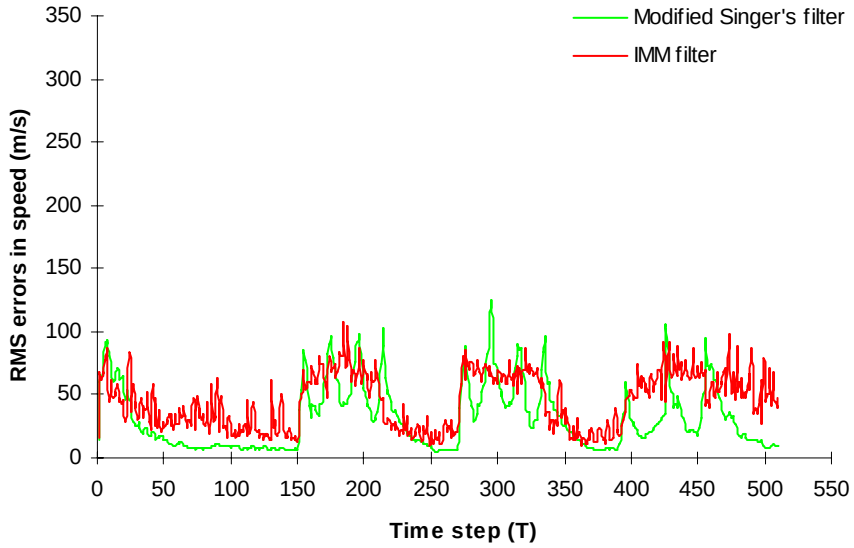


Fig. 5.20 RMS errors of modified Singer's filter and IMM filter in speed

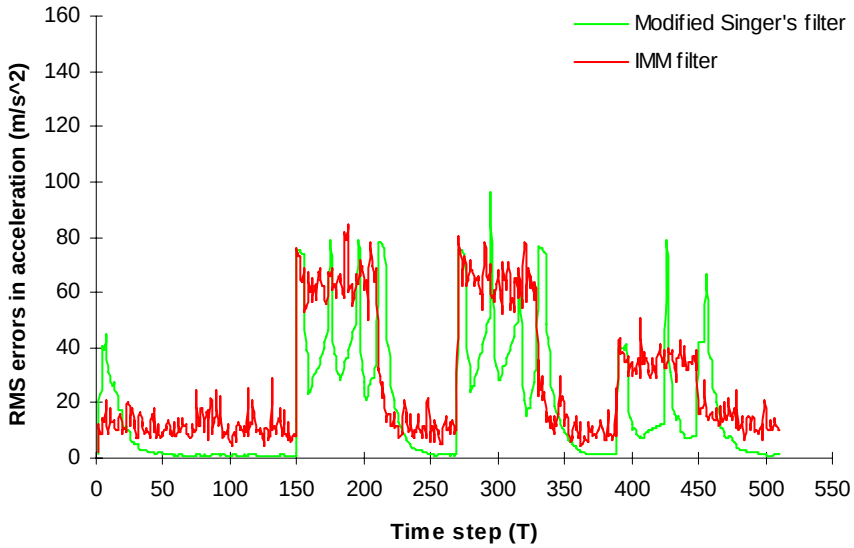


Fig. 5.21 RMS errors of modified Singer's filter and IMM filter in acceleration

Figs 5.19-5.21 show that the modified Singer's filter gives better performance than IMM filter in the absence of manoeuvre, and a comparable performance with IMM filter during the manoeuvre. In the absence of manoeuvre, the filter with zero accelerations dominates for most of time for IMM tracking and sometimes the zero acceleration filter, along with some non-zero acceleration filters dominates the IMM filter for the tracking. In the presence of the first manoeuvre, the tracking is performed mainly by around ten filters (one filter with zero in along-track and -50m/s^2 in cross-track, one filter with zero in along-track and -100m/s^2 in cross-track, one filter with 30m/s^2 in along-track and zero in cross-track, one filter with 30m/s^2 in along-track and -50m/s^2 in cross-track, and so on) in IMM. In the presence of the second and third manoeuvre, the IMM filter works similarly to the case in the first manoeuvre. The modified Singer's filter increases the process noise covariance through the whole processing and is compensated by input estimation during the manoeuvre, and gives performance better than or comparable to the IMM.

Comparing the computation times of trackers

	Computation time(s)
Singer's Filter	2.310
Modified Singer's Filter	8.808
IMM	74.040

Table 5.6 Computation loads of trackers used in simulation comparison

5.4 Summary

This chapter gave the detailed modified Singer's filter, and also compared the performance of modified Singer's filter with that of Singer's filter and IMM filter via three typical examples.

Singer's filter is simple, and its computation load is the lowest, but when the constructed process noise covariance is not able to cover the manoeuvre, track would be lost.

IMM filter is very popular in recent years for tracking and is thought be the most robust. However, its computation is heaviest, increasing the number of filters to increase the possibility of the real manoeuvre model being included causes more computation; decreasing the number of filters risks an incorrect filter being dominated.

Modified Singer's filter is derived by combining Singer's filter and input estimation. During the manoeuvres, input estimation is used to compensate Singer's filter to avoid the problem which the constructed process noise covariance is not able to cover the manoeuvre. Its computation load is modest, and its performance is much better than Singer's filter and at least comparable with IMM filter. This is why the multiple-model filter in Chapter 6 will use the modified Singer's filter in the uncertain manoeuvre.

To recapitulate, the modified Singer's filter model is described by equation (5.5), and the filter by equations (5.9), (5.10), (5.20), (5.22) and (5.23).

CHAPTER 6

MULTIPLE-MODEL TRACKING FILTER AND SIMULATIONS

As we know, the matched model filter with target motion will provide good performance for tracking, on the other hand, the mismatched model filter with the target motion will give a poor performance for tracking. However, the target motion is very complex, sometimes has a manoeuvre. The manoeuvre might need to be detected and distinguished. However, some manoeuvres might not be distinguished. Furthermore, detecting and distinguishing the manoeuvre need times to be done. Thus, a new multiple-model filter is developed in this work. The new multiple-model filter is constructed by modified Singer's filter and several other filters. The modified Singer's filter is used in the case of uncertain manoeuvre. Each of the other filters represents a set of motions, therefore, the new multiple-model is constructed by a limited number of filters.

This chapter will present the new multiple-model tracking filter, and compares this method with the various target tracking algorithms mentioned in the preceding chapters. The simulation results show good performance of the new multiple-model filter for tracking manoeuvring targets, and also highlight the economic computational load of using the new multiple-model filter.

6.1 Introduction

From Chapters 2 and 5, it is easy to summarize the following conclusions about tracking techniques in the literature. Input estimation (IE) algorithm (Chan et al., 1979; 1982) consists of the estimation of the unknown input (manoeuvre) over a sliding

window and suitable compensation of the state estimate under the assumption that the manoeuvre onset time is the starting point of the sliding window.

The performance of the VD filter (Bar-Shalom et al,1982) is superior to that of the IE filter in the case of low level manoeuvre, and the computational requirements of the VD algorithm is less than that of the IE algorithm. However, the VD algorithm may increase the tracking error because of the reconstruction of manouvre model.

The MM filter (Moose et al., 1975; 1977; 1979) usually consists of a large number of filters to cover probable manoeuvres, causing heavy computation. This approach is based on the "non-interacting" MM method: the single-model-based filters are running in parallel without mutual interaction, i.e., each filter operates independently at all times. Such an approach is quite effective in handling problems with an unknown structure or parameter but without structural or parametric changes. If the system structure or parameter changes, the MM filter could fail for tracking unless the models are updated.

The interacting multiple model (IMM) algorithm (Bar-Shalom, 1989; Lin, 1993) is a suboptimal hybrid filter. The IMM algorithm has been shown to be one of the most cost-effective schemes for the estimation of hybrid systems. Its main feature is its ability to estimate the state of a dynamic system which can switch between many behaviour modes according to a pre-defined Markov switching process. In particular, the IMM filter can act as a self-adjusting variable-bandwidth filter for tracking manoeuvring targets. Usually, the IMM filter incorporates a large number of filters to cover the possible manoeuvre motions, resulting in heavy computation. To reduce the computation load of IMM, several adaptive IMM filters have been developed.

However, all those methods of adaptive IMM filters' construction rely on the accuracy of estimated acceleration. If the estimated acceleration is not good enough, it would cause these adaptive IMM filters to fail for tracking.

In Chapter 5, the modified Singer's filter showed at least a comparable performance with IMM, but had a significant decrease in computation load. This Chapter will use the modified Singer's filter to provide the state estimation in the cases of uncertain manoeuvre.

This thesis suggests to use the modified Weston and Norton's manoeuvre detection to get quick manoeuvre detection and find accurate manoeuvre step, and then to estimate the manoeuvre accurately using IE to avoid reinitialising for manoeuvre model in VD filter.

Therefore, this work proposes a new multiple-model filter for tracking manoeuvring targets. This new multiple model approach is assuming several manoeuvre models when the manoeuvre is detected, such as, one is straight-line acceleration motion model, one is curvilinear acceleration motion model, and the other is circular motion model. The switching decision from one model to another is made by the manoeuvre detection and the estimated accelerations. For any ambiguous manoeuvre, the tracking is provided by modified Singer's filter. The reinitialising for the switched model is provided by the modified Singer's filter. The procedure of the new MM filter is that each case tracking of non-manoeuvre, manoeuvre, and ambiguity is performed by the corresponding filter, producing a good performance. The operation of the new MM filter is simple like VD filter but without reconstructing the entire estimates of manoeuvre filters, and the new MM filter also considers all kind of motions like IMM filter but with a limited number

of models, reducing computational load. Section 6.2 presents the multiple-model filter in detail. Section 6.3 gives the simulation results. Section 6.4 gives the further discussions about a range change in accelerations. Finally, Section 6.5 summarizes this chapter.

A further improvement for the multiple-model tracking is provided by using fixed-lag smoothing technique. The fixed-lag smoothing technique will be described in Chapter 7.

6.2 Multiple-Model Filter

The multiple-model filter can be briefly outlined, as illustrated in Fig. 6.1, mainly consists of 3 stages as follows:

- Manoeuvre detection
- Manoeuvre decision
- State estimate selection

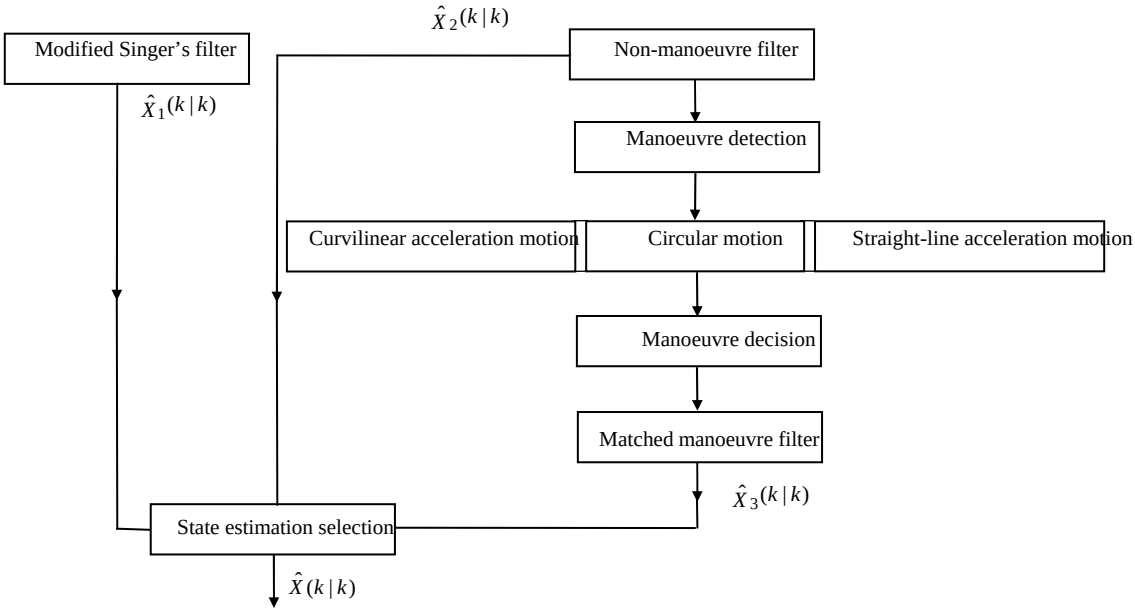


Fig. 6.1 Flowchart of the new multiple-model filter

The manoeuvre detection has been discussed in Chapter 4; this work will use the modified Weston-Norton manoeuvre detection to provide quick detection and find the manoeuvre steps in Section 6.3.

When the manoeuvre is detected, several possible manoeuvres are assumed, the inputs are estimated by using the possible manoeuvre models to distinguish the manoeuvre, and the modified Weston-Norton detection is used to find the step of manoeuvre. In practice, the target manoeuvre can be classified into straight-line acceleration motion, circular motion and curvilinear acceleration motion. According to the 3 kinds of motion, the multiple-model filter contains the 3 kinds of motion model. In order that each model represents a kind of motion, the straight-line acceleration motion model is constructed by an augmented state model with position, speed, and along-track acceleration states; the circular motion model is constructed by an augmented state model with position, speed, and cross-track acceleration states; and curvilinear acceleration motion model is constructed by an augmented state model with position, speed, cross- and along-track accelerations states. Thus, after the manoeuvre has been detected, the three model filters are run in parallel to estimate the corresponding manoeuvre, and in the meanwhile, the manoeuvre detection is processing along with each filter. The quicker the new manoeuvre is detected, the more unreliable the assumed manoeuvre filter is, so the estimated input is also used to make the manoeuvre decision. The models have been discussed in Chapter 3.

During distinguishing the manoeuvre, the tracking is provided by the modified Singer's filter. Once the manoeuvre is certain and the manoeuvre step is found, the tracking is provided by the matched manoeuvre model with process noise standard deviation which

is taken to be 5% of the estimated accelerations. The non-manoeuvre filter is constructed by constant-velocity model with unaugmented state model in Fig. 6.1.

6.3 Simulations

The simulations were done on a Intel 82371AB Pentium II processor and written in MATLAB with version 5.3 as in Chapter 5.

The accuracy is evaluated through the root-mean-square (rms) estimation error for position, speed and acceleration as well.

$$\text{rms position error}_{\text{sample } k} \equiv \tilde{p}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{x}_k^i - x_k^i)^2 + (\hat{y}_k^i - y_k^i)^2)}{N}}$$

$$\text{rms speed error}_{\text{sample } k} \equiv \tilde{v}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{\dot{x}}_k^i - \dot{x}_k^i)^2 + (\hat{\dot{y}}_k^i - \dot{y}_k^i)^2)}{N}}$$

$$\text{rms acceleration error}_{\text{sample } k} \equiv \tilde{a}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{a}_{nk}^i - a_{nk}^i)^2 + (\hat{a}_{tk}^i - a_{tk}^i)^2)}{N}}$$

$$\text{or rms acceleration error}_{\text{sample } k} \equiv \tilde{a}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{a}_{x(k|k)}^i - a_{xk}^i)^2 + (\hat{a}_{y(k|k)}^i - a_{yk}^i)^2)}{N}}$$

where $\hat{x}_k^i$ is the estimate of x at time k in simulation run number i; the actual value is x_k^i.

The computational load is evaluated using the CPU executing time.

For each test, a Monte Carlo test of N=100 runs was performed.

The following uses the same three examples as in Chapter 5 to compare the new multiple-model filter with the various target tracking algorithms mentioned in the preceding chapters in performance and computation load.

6.3.1 Comparison of trackers using a simple target with rectilinear acceleration

motion

The target's trajectory is at first a constant-velocity, and then has an acceleration motion to a new velocity, and finally executes the new constant-velocity motion, as in the first example in Chapter 5. The manoeuvre is the result of a lateral constant acceleration input. This case is the most frequently used.

Six trackers are tested here,

tracker	details of tracker
CV	constant-velocity-based Kalman filter
VDF	models: constant-velocity motion and rectilinear-acceleration manoeuvre detection: testing the statistic of normalised squared innovations window size: 5 samples for manoeuvre detection 3 samples for the end of manoeuvre detection process noise standard deviation of the manoeuvre filter: 5% of the estimated acceleration initial values of manoeuvre filter: provided by measurements at the start of the sliding window
Singer's Filter	manoeuvre correction time: 10s manoeuvre acceleration variance for target model: 3^2
Modified Singer's Filter	manoeuvre level estimation: input estimation manoeuvre detection: testing the statistic of normalised squared innovations and using modified Weston and Norton's detection with fixed-lag smoothing window size: 5 samples; fixed-lag: 3 samples target accelerations around estimated manoeuvre level: modelled by Singer's model manoeuvre correction time and manoeuvre acceleration variance: the same as for Singer's filter.
IMM	x-direction accelerations used: a_x={±50.0,±25.0,0.0}m/s^2 y-direction accelerations used: a_y={±50.0,±25.0,0.0}m/s^2 with a probability of no change in manoeuvre of 0.9.
Proposed Filter	models: constant-velocity motion, modified Singer's model and augmented state model (with accelerations) manoeuvre detection: using modified Weston and Norton's detection with fixed-lag smoothing fixed-lag: 3 samples initial values of manoeuvre: provided by measurements within the sliding window with modified Singer's filter decision of switch model: made by manoeuvre detection and estimated acceleration

Table 6.1 Trackers used in simulation comparison

For IMM, CV, VD and proposed filters, the initial estimates of the states (Bar-Shalom and Birmiwal, 1982) are

$$\hat{x}(0 \mid 0) = Z_x(0) = x(0) + V_x(0) = r_m(0)\cos\theta_m(0)$$

$$\hat{y}(0 \mid 0) = Z_y(0) = y(0) + V_y(0) = r_m(0)\sin\theta_m(0)$$

$$\hat{\dot{x}}(0 \mid 0) = \frac{Z_x(0) - Z_x(-1)}{T} = \frac{r_m(0)\cos\theta_m(0) - r_m(-1)\cos\theta_m(-1)}{T}$$

$$\hat{\dot{y}}(0 \mid 0) = \frac{Z_y(0) - Z_y(-1)}{T} = \frac{r_m(0)\sin\theta_m(0) - r_m(-1)\sin\theta_m(-1)}{T}$$

where $Z_x(-1) = x(0) - \dot{x}(0)T + V_x(-1)$ and $Z_y(-1) = y(0) - \dot{y}(0)T + V_y(-1)$

$$P(0 \mid 0) = \begin{bmatrix} R_L^{11}(0) & R_L^{12}(0) & \frac{R_L^{11}(0)}{T} & \frac{R_L^{12}(0)}{T} \\ R_L^{12}(0) & R_L^{22}(0) & \frac{R_L^{12}(0)}{T} & \frac{R_L^{22}(0)}{T} \\ \frac{R_L^{11}(0)}{T} & \frac{R_L^{12}(0)}{T} & \frac{R_L^{11}(0) + R_L^{11}(-1)}{T^2} & \frac{R_L^{12}(0) + R_L^{12}(-1)}{T^2} \\ \frac{R_L^{12}(0)}{T} & \frac{R_L^{22}(0)}{T} & \frac{R_L^{12}(0) + R_L^{12}(-1)}{T^2} & \frac{R_L^{22}(0) + R_L^{22}(-1)}{T^2} \end{bmatrix}$$

For Singer's filter and modified Singer's filter, the initial estimates of the states (Singer, 1970) are

$$\hat{x}(0 \mid 0) = Z_x(0) = x(0) + V_x(0) = r_m(0)\cos\theta_m(0)$$

$$\hat{y}(0 \mid 0) = Z_y(0) = y(0) + V_y(0) = r_m(0)\sin\theta_m(0)$$

$$\hat{\dot{x}}(0 \mid 0) = \frac{Z_x(0) - Z_x(-1)}{T} = \frac{r_m(0)\cos\theta_m(0) - r_m(-1)\cos\theta_m(-1)}{T}$$

$$\hat{\dot{y}}(0 \mid 0) = \frac{Z_y(0) - Z_y(-1)}{T} = \frac{r_m(0)\sin\theta_m(0) - r_m(-1)\sin\theta_m(-1)}{T}$$

$$\hat{\ddot{x}}(0 \mid 0) = 0$$

$$\hat{\ddot{y}}(0 \mid 0) = 0$$

$$P(0\mid 0)=\begin{bmatrix} R_L^{11}(0) & R_L^{12}(0) & \dfrac{R_L^{11}(0)}{T} & \dfrac{R_L^{12}(0)}{T} & 0 & 0 \\ R_L^{12}(0) & R_L^{22}(0) & \dfrac{R_L^{12}(0)}{T} & \dfrac{R_L^{22}(0)}{T} & 0 & 0 \\ \dfrac{R_L^{11}(0)}{T} & \dfrac{R_L^{12}(0)}{T} & \dfrac{R_L^{11}(0)+R_L^{11}(-1)}{T^2} & \dfrac{R_L^{12}(0)+R_L^{12}(-1)}{T^2} & 0 & 0 \\ \dfrac{R_L^{12}(0)}{T} & \dfrac{R_L^{22}(0)}{T} & \dfrac{R_L^{12}(0)+R_L^{12}(-1)}{T^2} & \dfrac{R_L^{22}(0)+R_L^{22}(-1)}{T^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where $R_L^{11}=r_m^2\sigma_\theta^2\sin^2\theta_m+\sigma_r^2\cos^2\theta_m$

$R_L^{22}=r_m^2\sigma_\theta^2\cos^2\theta_m+\sigma_r^2\sin^2\theta_m$

and $R_L^{12}=(\sigma_r^2-r_m^2\sigma_\theta^2)\sin\theta_m\cos\theta_m$ (presented in Section 3.8)

The simulation results are shown in Figs 6.2-18. Figs 6.2 and 6.3 present one-run results of manoeuvre detection by using the modified Weston and Norton's detection with fixed-lag smoothing. The detection of the start of manoeuvre and the detection of the end of manoeuvre are shown in Fig. 6.2 and Fig. 6.3, respectively. Figs 6.4-6.6 show the performance of the proposed filter and constant-velocity filter in position, speed and acceleration. Figs 6.7-6.9 show the performance of the proposed filter and variable-dimension filter in position, speed and acceleration. Figs 6.10-6.12 show the performance of the proposed filter and Singer's filter in position, speed and acceleration. Figs 6.13-6.15 show the performance of the proposed filter and modified Singer's filter in position, speed and acceleration. Figs 6.16-6.18 show the performance of the proposed filter and IMM filter in position, speed and acceleration.

Table 6.2 shows the computation loads of trackers used.

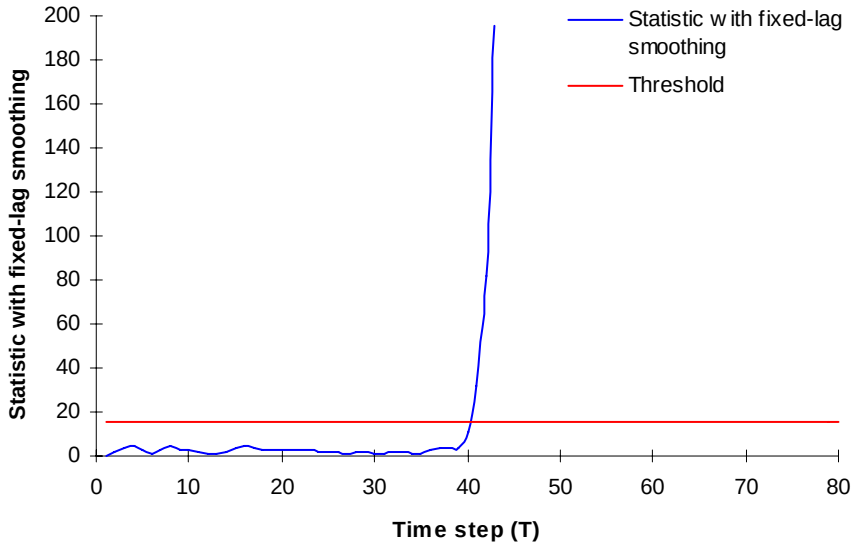


Fig. 6.2 Statistic with fixed-lag smoothing for the detection of manoeuvre start

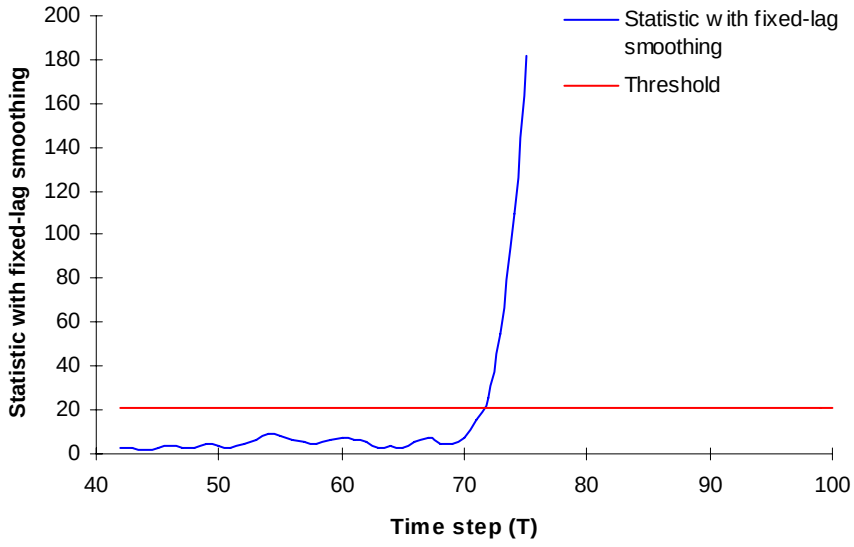


Fig. 6.3 Statistic with fixed-lag smoothing for the detection of manoeuvre end

Fig. 6.2 shows that the manoeuvre is detected at step 44 including the fixed lag. Fig. 6.3 shows that the end of manoeuvre is detected at step 75 including the fixed lag.

From the Figures and analysis above, the manoeuvre start and end are detected at times 44 and 75 respectively. The start of manoeuvre step is not found until at step 46, and the end of manoeuvre step is not found until at step 77. Thus the proposed multiple-model filter is that the unaugmented state model filter is used from the beginning to step 43, the modified Singer's filter is used from step 44 to 46 where the manoeuvre is not sure, and the augmented state model filter with rectilinear accelerations is used from step 47 to 74. After the manoeuvre end is certain by using manoeuvre detection from steps 75 to 77 where the modified Singer's filter is used, the unaugmented state model is used until the end of tracking.

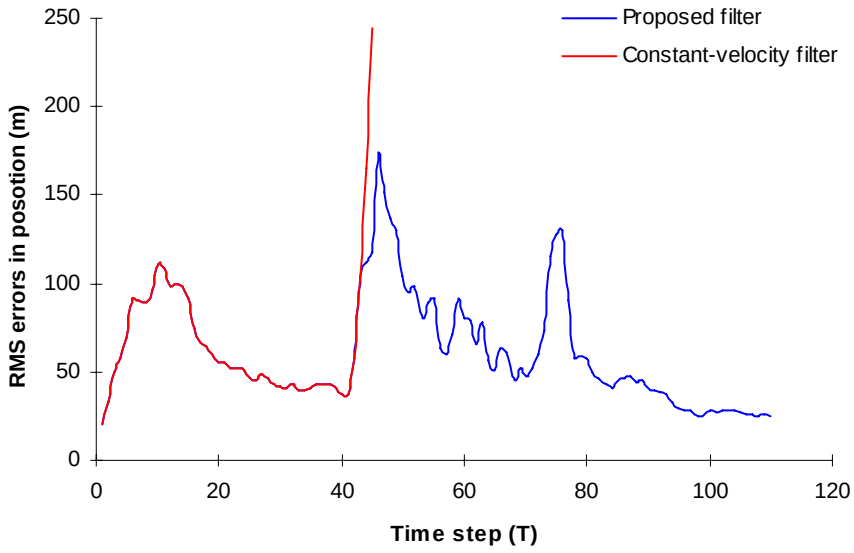


Fig. 6.4 RMS errors of proposed filter and constant-velocity filter in position

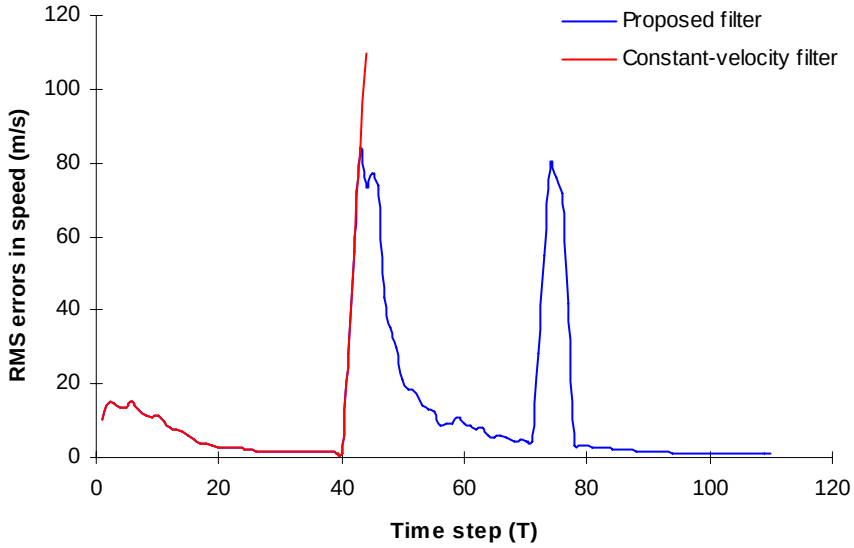


Fig. 6.5 RMS errors of proposed filter and constant-velocity filter in speed

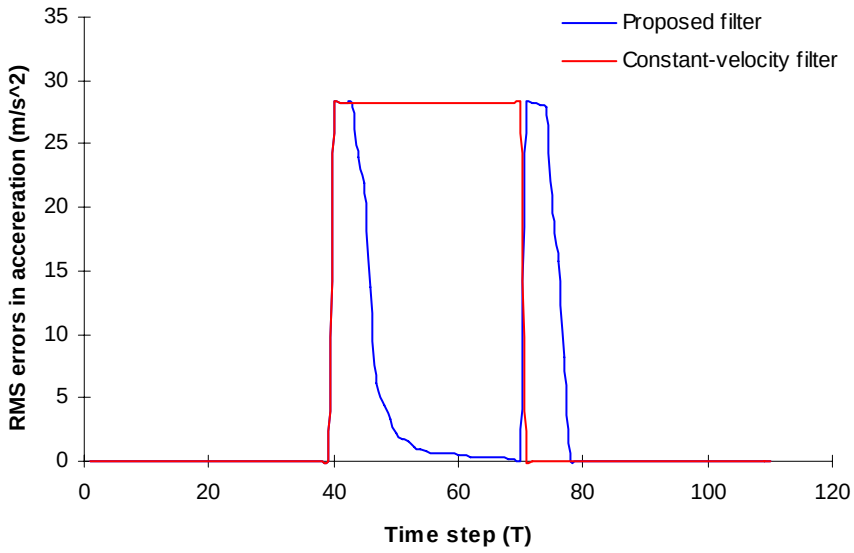


Fig. 6.6 RMS errors of proposed filter and constant-velocity filter in acceleration

Figs 6.4-6.6 show that the performance of proposed multiple-model filter is much better than that of constant-velocity filter. In the presence of manoeuvre, the constant-velocity model is not able to match the target motion. However, the proposed multiple-model filter tries to use the matched target motion model for tracking, even though in the case

of uncertain manoeuvres, the modified Singer's filter is used to provide safeguards for tracking.

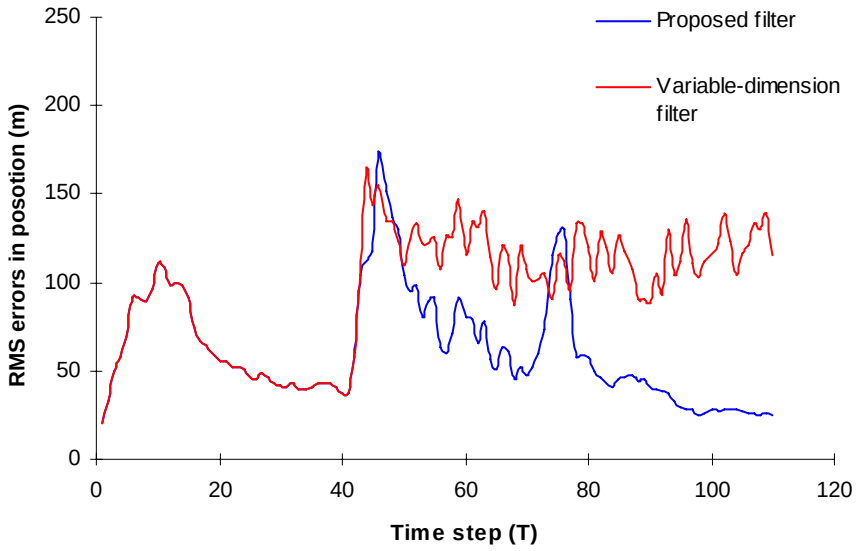


Fig. 6.7 RMS errors of proposed filter and variable-dimension filter in position

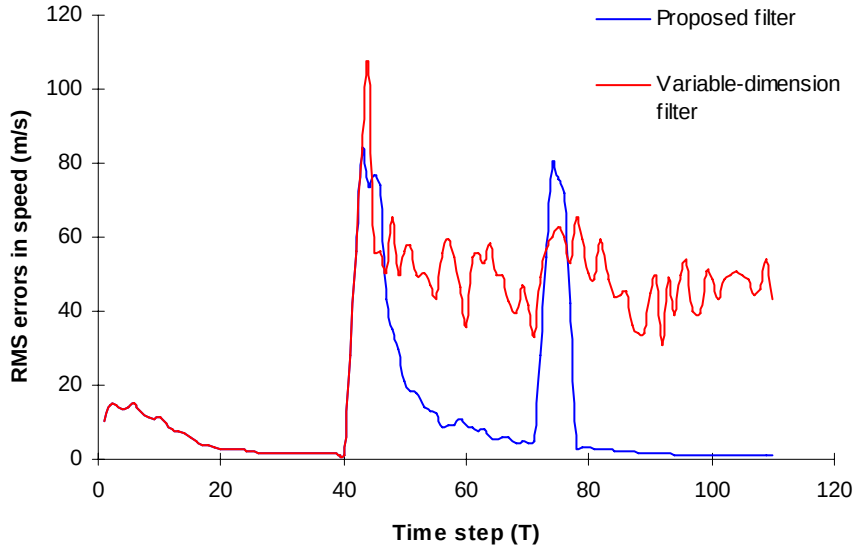


Fig. 6.8 RMS errors of proposed filter and variable-dimension filter in speed

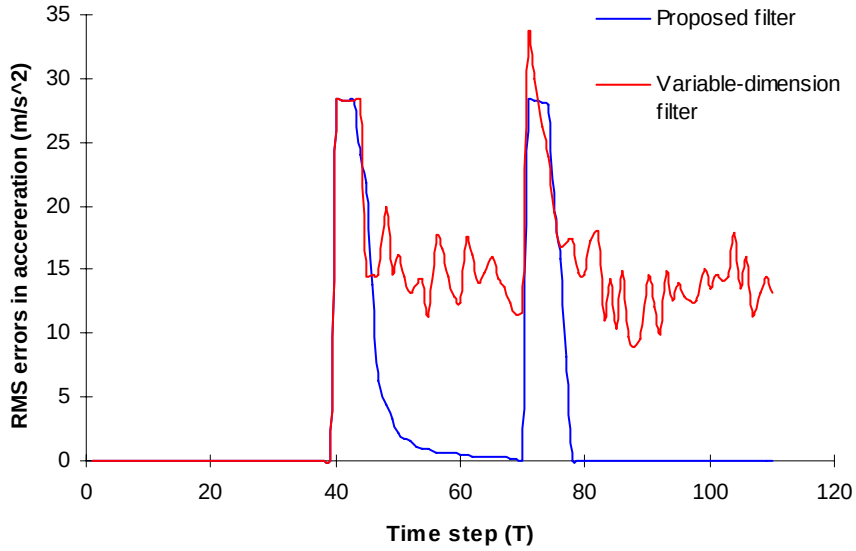


Fig. 6.9 RMS errors of proposed filter and variable-dimension filter in acceleration

Figs 6.7-6.9 show that the performance of the proposed multiple-model is much better than that of the variable-dimension filter. When the manoeuvre is detected, the manoeuvre model in variable-dimension filter is reconstructed only by the measurements at the start of the sliding window. Figs above show that the unsuitable reconstruction of manoeuvre model and the delay of manoeuvre detection in variable-dimension filter cause tracking error.

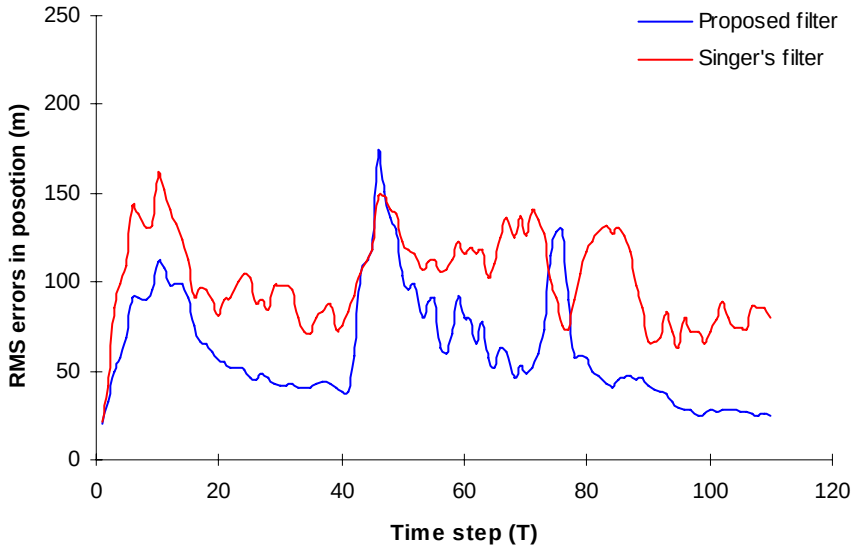


Fig. 6.10 RMS errors of proposed filter and Singer's filter in position

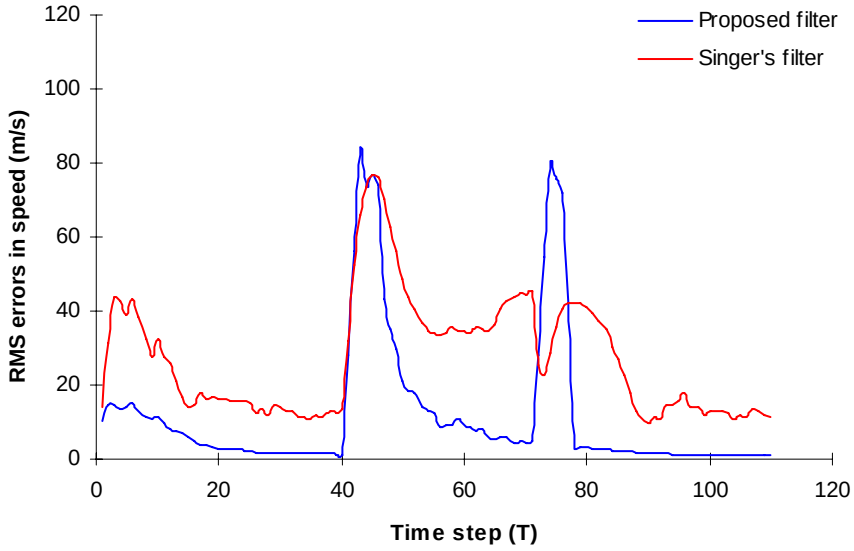


Fig. 6.11 RMS errors of proposed filter and Singer's filter in speed

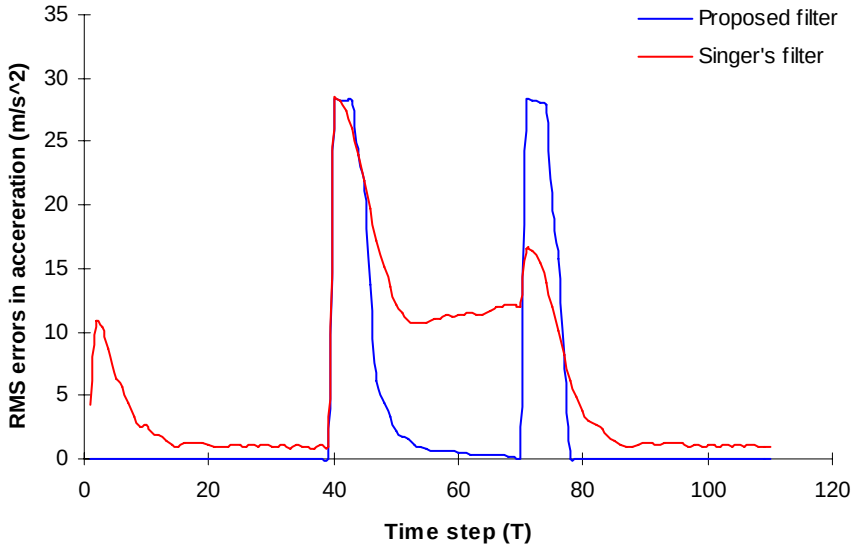


Fig. 6.12 RMS errors of proposed filter and Singer's filter in acceleration

Figs 6.10-6.12 show that the performance of proposed multiple-model filter is much better than that of Singer's filter at most times. The increased process noise covariance in Singer's filter is not able to cover the manoeuvre, causing tracking error. In very few steps around the steps of manoeuvre and of the end of manoeuvre, the performance of the proposed multiple-model filter is little worse than that of Singer's filter because the manoeuvre detection has a few steps delay and for some steps the tracking in proposed multiple-model filter is from the initial estimate of forcing estimates of the modified Singer's filter.

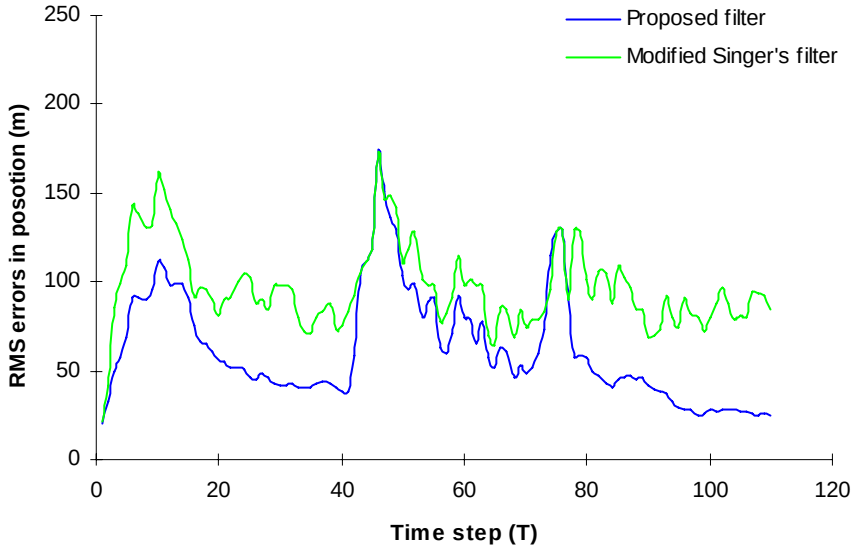


Fig. 6.13 RMS errors of proposed filter and modified Singer's filter in position

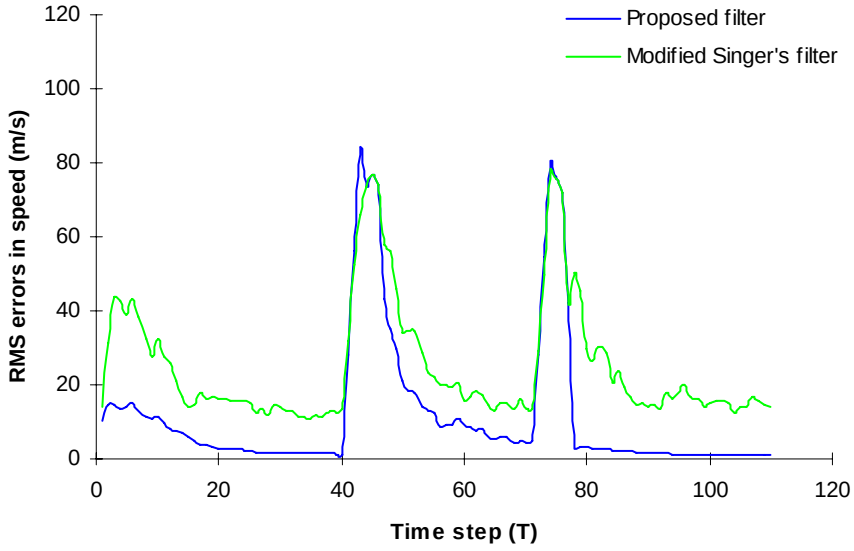


Fig. 6.14 RMS errors of proposed filter and modified Singer's filter in speed

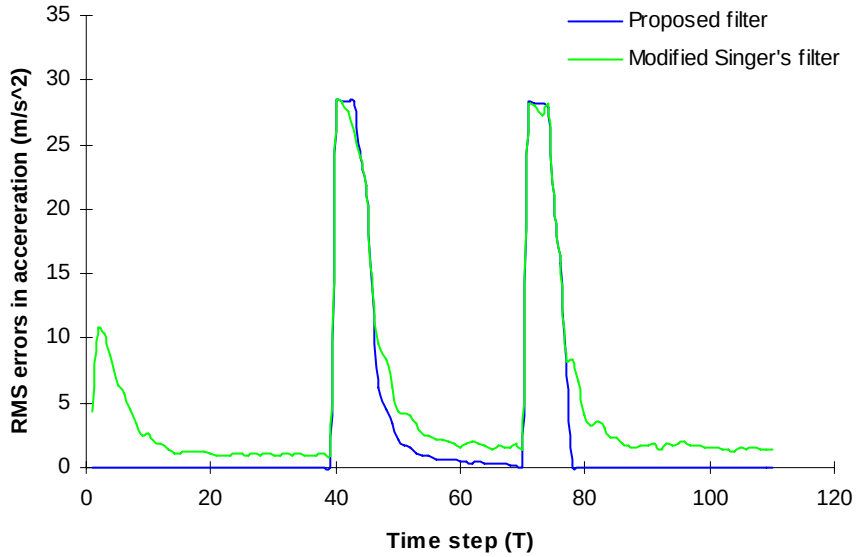


Fig. 6.15 RMS errors of proposed filter and modified Singer's filter in acceleration

Figs 6.13-6.15 show that the performance of proposed multiple-model is much better than that of modified Singer's filter during the certain manoeuvre. Around the start of the manoeuvre and around the end of manoeuvre, the modified Singer's filter gives the better performance than the multiple-model filter. In the case of certain manoeuvre, the proposed multiple-model filter uses the matched the filter for tracking, but around the manouvre and around the end of manoeuvre, because the manoeuvre detection has several steps delay, the original matched filter is not suitable for the manoeuvre, but the modified Singer's filter covers manoeuvre better by increasing its process noise covariance.

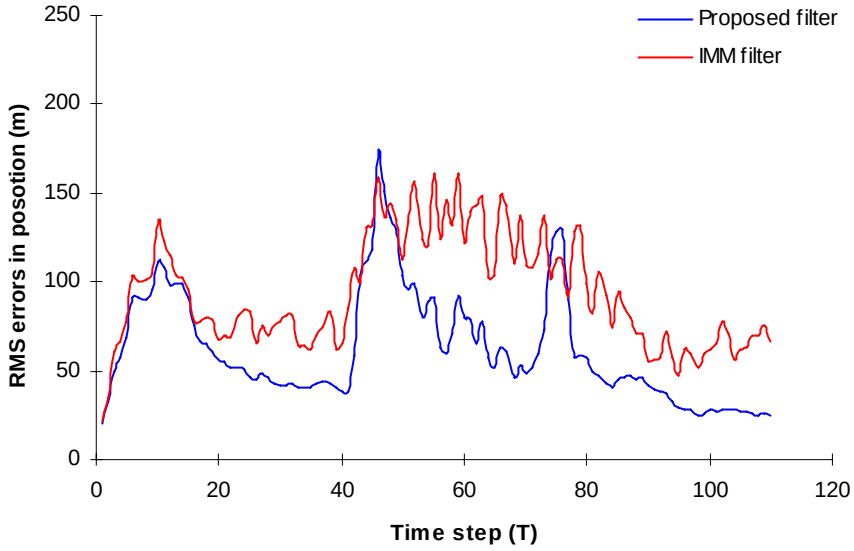


Fig. 6.16 RMS errors of proposed filter and IMM filter in position

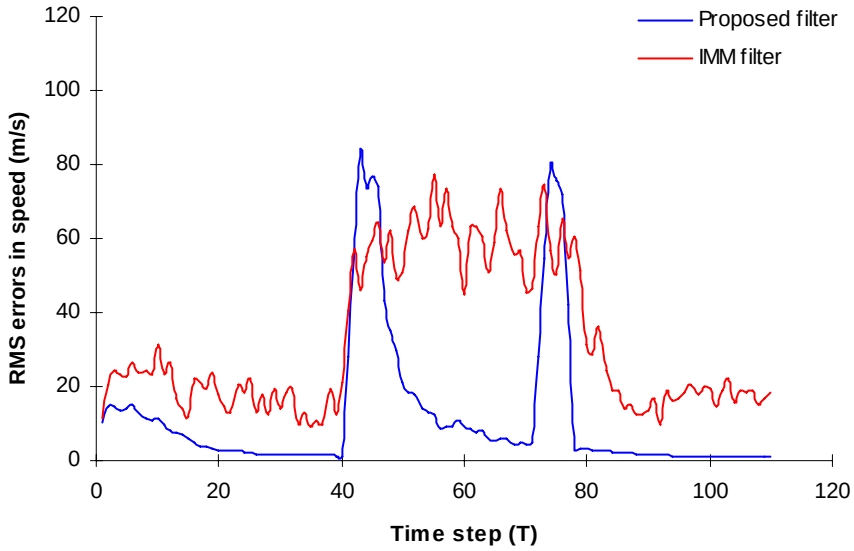


Fig. 6.17 RMS errors of proposed filter and IMM filter in speed

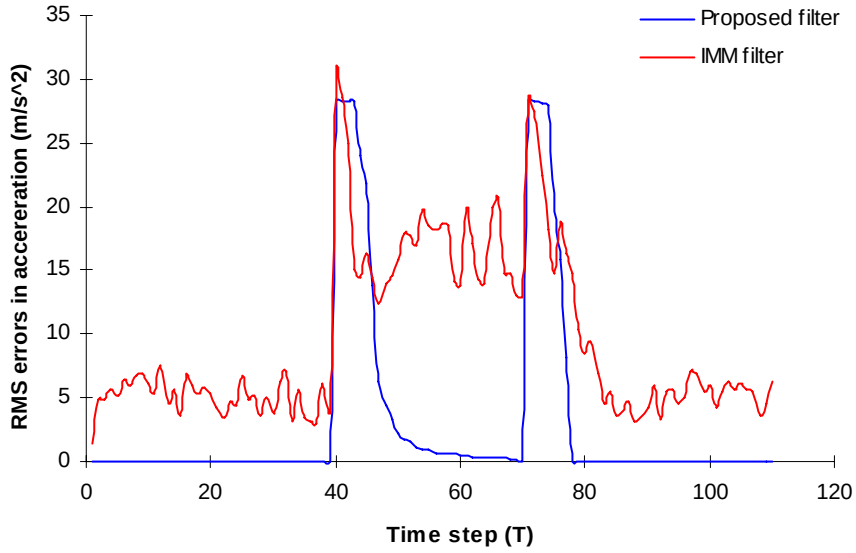


Fig. 6.18 RMS errors of proposed filter and IMM filter in acceleration

Figs 6.16-6.18 show that at few steps around the start of the manoeuvre and around the end of manoeuvre, the IMM filter gives the better performance than the multiple-model filter. The performance of proposed multiple-model filter is much better than that of IMM filter during certain manoeuvre. Around the start of the manouvre and around the end of manoeuvre, because the manoeuvre detection has several steps delay, the original matched filter is not suitable for the manoeuvre; but the IMM filter has the ability of self-adjusting to "switch" from one model to another and make it respond more quickly to the future manoeuvre. However, at most times, the proposed multiple-model filter uses the matched the filter for tracking in the certain case of manoeuvre.

Comparing the computation times of trackers

	Computation time(s)
Constant-velocity filter	0.120
Variable-dimension filter	0.274
Singer's Filter	0.164
Modified Singer's Filter	1.022
IMM	14.520
Proposed Filter	1.792

Table 6.2 Computation loads of trackers used in simulation comparison

6.3.2 Comparison of Trackers with a circular motion target

The target's trajectory is kept in a constant-speed motion, the track contains a constant-speed turn and two constant-velocity motions as the second example in Chapter 5.

Six trackers are tested here,

tracker	details of tracker
CV	constant-velocity-based Kalman filter
VDF	models: constant-velocity motion and rectilinear-acceleration manoeuvre detection: testing the statistic of normalised squared innovations window size: 5 samples for manoeuvre detection 3 samples for the end of manoeuvre detection process noise standard deviation of the manoeuvre filter: 5% of the estimated acceleration initial values of manoeuvre filter: provided by measurements at the start of the sliding window
Singer's Filter	manoeuvre correction time: 10s manoeuvre acceleration variance for target model: 3^2
Modified Singer's Filter	manoeuvre level estimation: input estimation manoeuvre detection: testing the statistic of normalised squared innovations and using modified Weston and Norton's detection with fixed-lag smoothing window size: 5 samples; fixed-lag: 3 samples target accelerations around estimated manoeuvre level: modelled by Singer's model manoeuvre correction time and manoeuvre acceleration variance: the same as for Singer's filter.
IMM	along-track accelerations used: a_t={±15.0,±7.5.0,0.0}m/s^2 cross-track accelerations used: a_n={±50.0,±25.0,0.0}m/s^2 with a probability of no change in manoeuvre of 0.9.
Proposed Filter	models: constant-velocity motion, modified Singer's model, straight line motion with an augmented state (with along-track acceleration), circular motion with an augmented state (with cross-track acceleration), and curvilinear motion with augmented states (cross- and along-track accelerations) manoeuvre detection: using modified Weston and Norton's detection with fixed-lag smoothing fixed-lag: 3 samples initial values of manoeuvre: provided by measurements within the sliding window with modified Singer's filter decision of switch model: made by manoeuvre detection and estimated accelerations.

Table 6.3 Trackers used in simulation comparison

The initialization of the filters are the same as in the first example. The detail of initialization is not reiterated here.

The simulation results are shown in Figs 6.19-40. Figs 6.19, 6.20, 6.22, and 6.24 present one-run results of manoeuvre detection by using the modified Weston and Norton's detection with fixed-lag smoothing. Figs 6.20, 6.22 and 6.24 are produced by assuming the manoeuvre be curvilinear acceleration motion, circular motion, and straight-line acceleration motion respectively. Figs 6.21, 6.23, and 6.25 are shown the estimated accelerations. Figs 6.21, 6.23, and 6.25 are created by using augmented state models with cross- and along-track accelerations, with cross-track acceleration, and with along-track acceleration, respectively. Figs 6.26-6.28 show the performance of the proposed filter and constant-velocity filter in position, speed and acceleration. Figs 6.29-6.31 show the performance of the proposed filter and variable-dimension filter in position, speed and acceleration. Figs 6.32-6.34 show the performance of the proposed filter and Singer's filter in position, speed and acceleration. Figs 6.35-6.37 show the performance of the proposed filter and modified Singer's filter in position, speed and acceleration. Figs 6.38-6.40 show the performance of the proposed filter and IMM filter in position, speed and acceleration.

Table 6.4 shows the computation loads of trackers used.

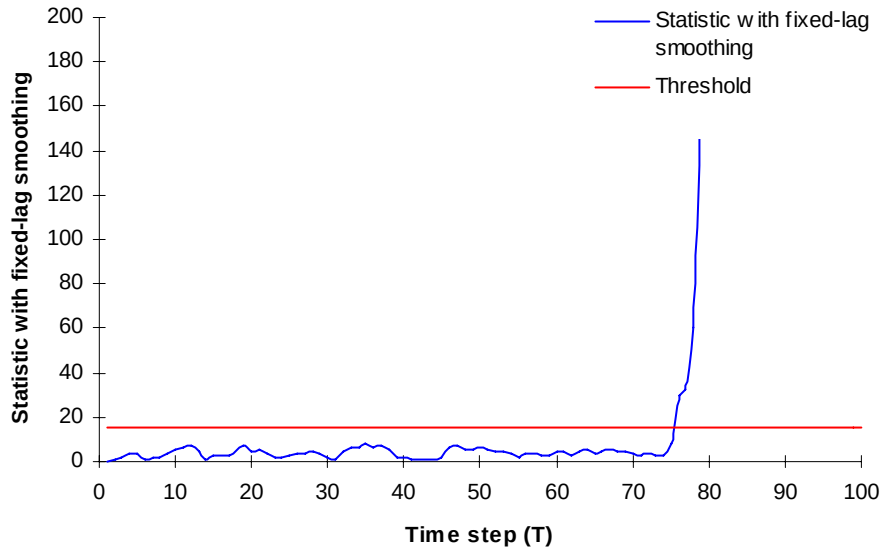


Fig. 6.19 Statistic with fixed-lag smoothing for the detection of manoeuvre start

The manoeuvre is detected at step 79 including the fixed lag.

The three manoeuvre filters are run and check to distinguish the manoeuvre by using manoeuvre detection and estimating the accelerations in parallel, as follows.

1) Running the augmented state model with along- and cross-track accelerations

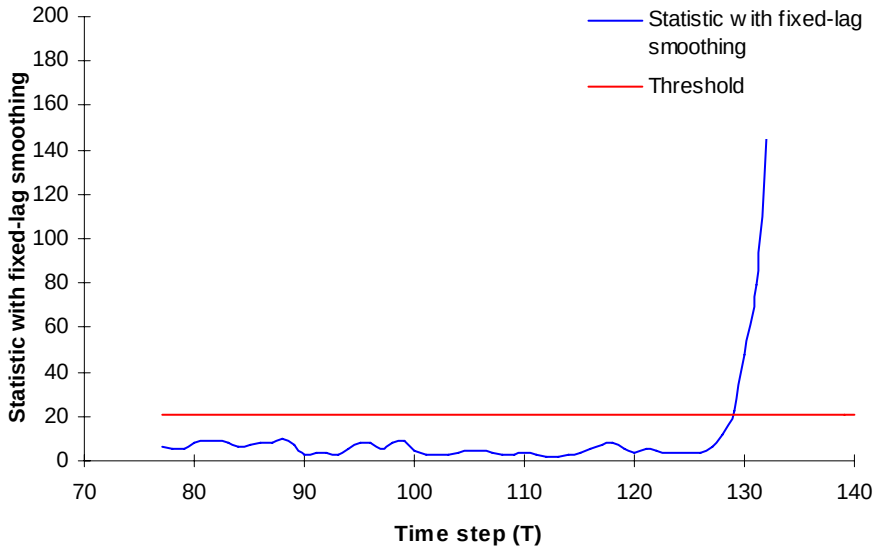


Fig. 6.20 Statistic with fixed-lag smoothing for the detection of manoeuvre end

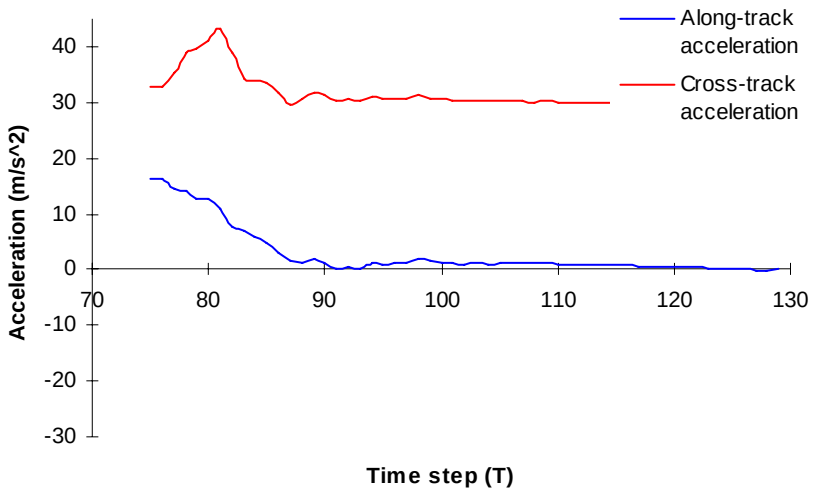


Fig. 6.21 Estimated along- and cross-track accelerations

From Figs 6.20 and 6.21, the end of manoeuvre is detected at step 132 including the fixed lag, and the along-track acceleration approaches towards to zero and the cross-track acceleration approaches towards to a constant around step 84.

2) Running the augmented state model with cross-track acceleration

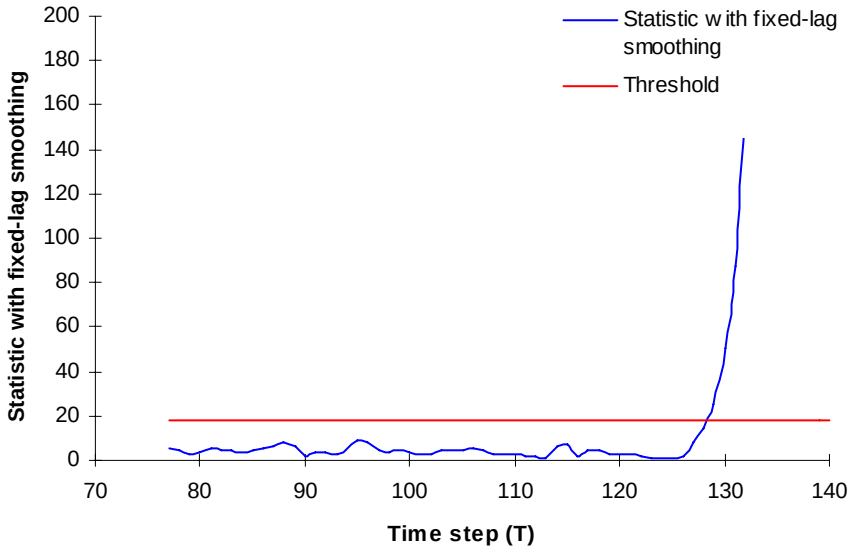


Fig. 6.22 Statistic with fixed-lag smoothing for the detection of manoeuvre end

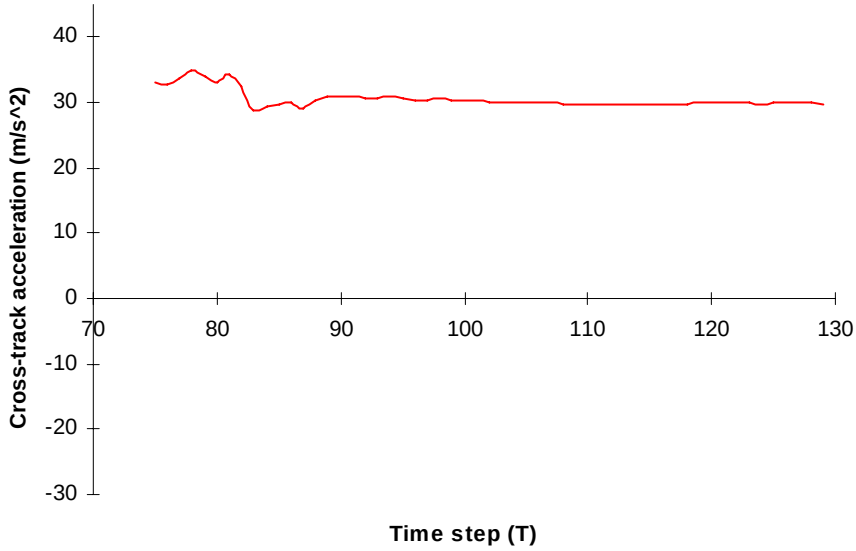


Fig. 6.23 Estimated cross-track acceleration

From Fig. 6.22 and 6.23, the manoeuvre ends at step 132 including the fixed lag, and the cross-track acceleration approaches near-constancy around step 84.

3) Running the augmented state model with along-track acceleration

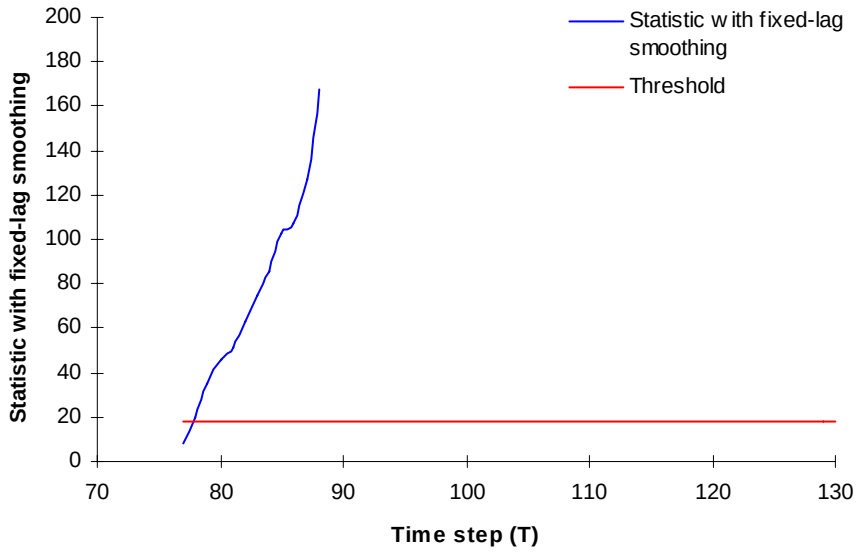


Fig. 6.24 Statistic with fixed-lag smoothing for the detection of manoeuvre end

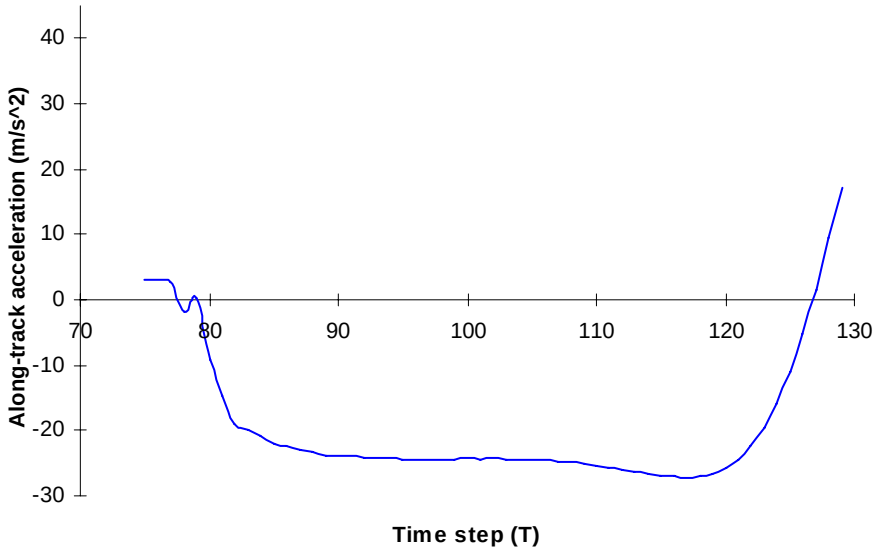


Fig. 6.25 Estimated along-track acceleration

From Figs 6.24, the manoeuvre detection shows that the assumed manoeuvre model is not fit for target motion immediately.

From Figs and analysis above, the manoeuvre start and end are detected at step 79 and 132 respectively, and the manoeuvre is decided to be circular motion at step 84. Thus the proposed multiple-model filter is that the unaugmented state model filter is used from the beginning to step 78, the modified Singer's filter is used from step 79 to 83, and the augmented state model filter with cross-track acceleration is used from step 84 to 131. After the manoeuvre end is sure by using manoeuvre detection from step 132 to 134 where the modified Singer's filter is used, the unaugmented state model is used until the end of tracking.

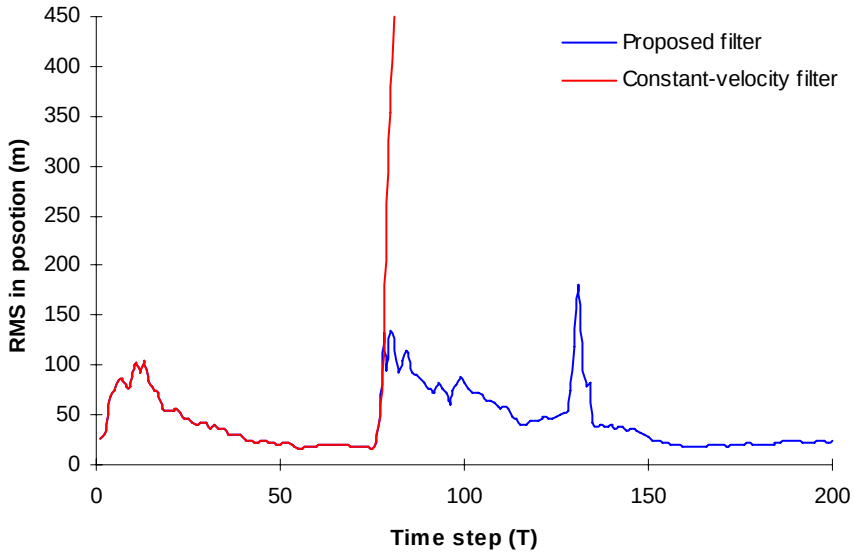


Fig. 6.26 RMS errors of proposed filter and constant-velocity filter in position

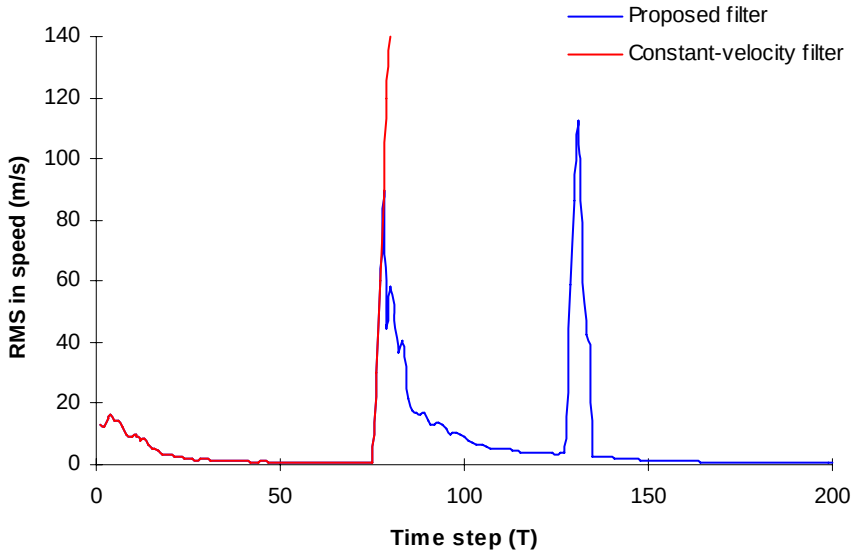


Fig. 6.27 RMS errors of proposed filter and constant-velocity filter in speed

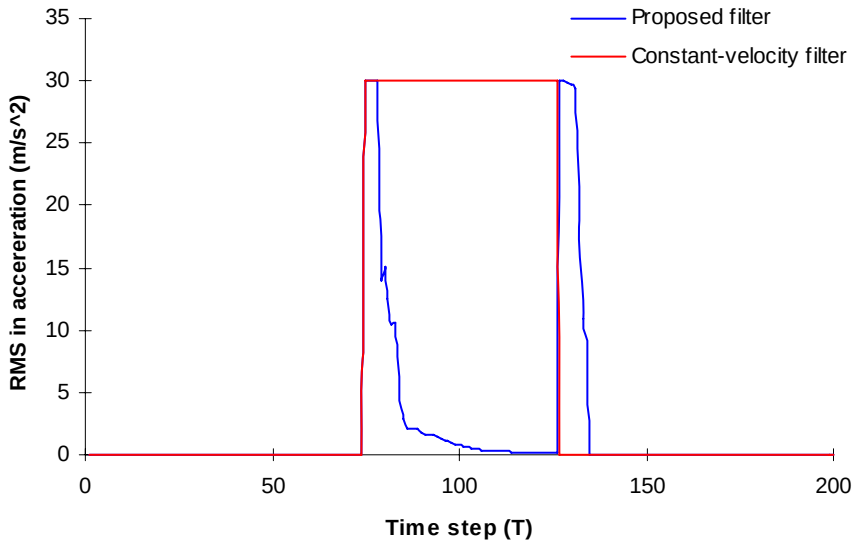


Fig. 6.28 RMS errors of proposed filter and constant-velocity filter in acceleration

Figs 6.26-6.28 show that the performance of proposed multiple-model filter is much better than that of constant-velocity filter, which is similar to the case with rectilinear acceleration motion mentioned before. In the present of manoeuvre, the constant-velocity model is not able to match the target motion. However, the proposed multiple-model filter uses matched target motion model at most times, even though in few steps of uncertain manoeuvres, the modified Singer's filter which is more suitable for target motion is used to provide safeguards for tracking.

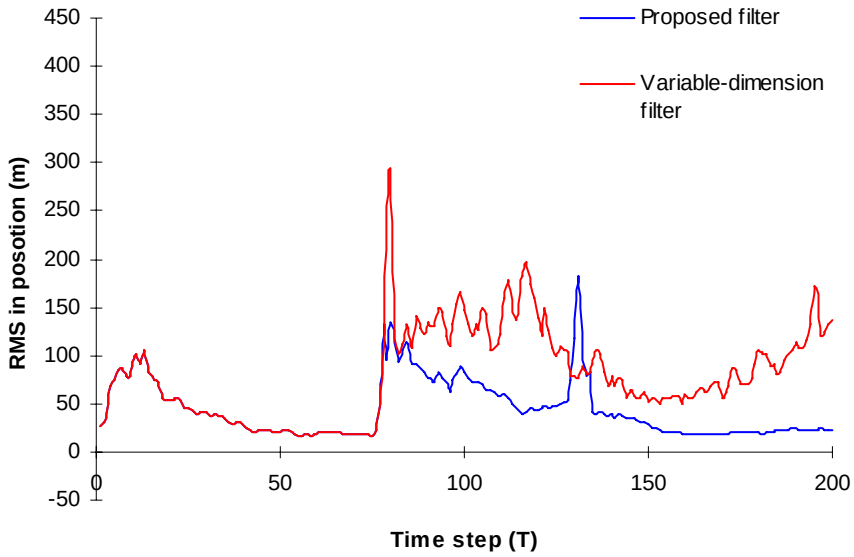


Fig. 6.29 RMS errors of proposed filter and variable-dimension filter in position

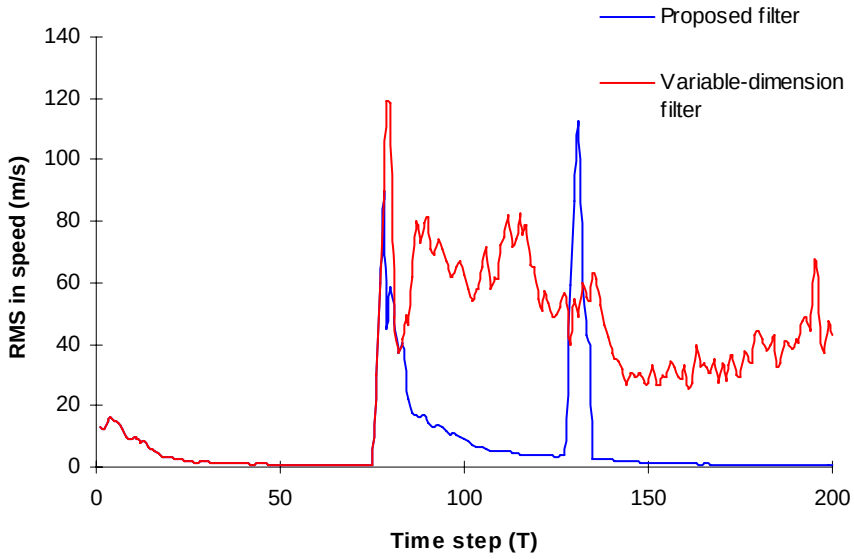


Fig. 6.30 RMS errors of proposed filter and variable-dimension filter in speed

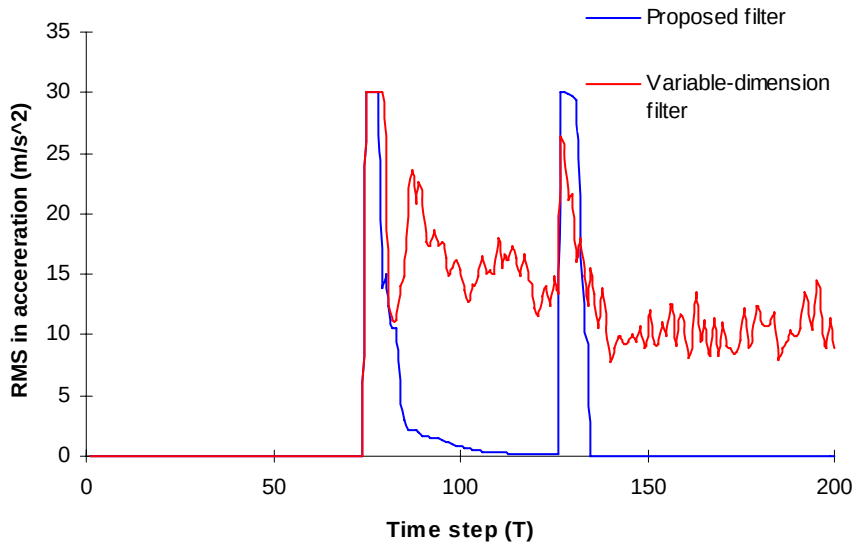


Fig. 6.31 RMS errors of proposed filter and variable-dimension filter in acceleration

Figs 6.29-6.31 show that the performance of proposed multiple-model filter is much better than that of variable-dimension filter. When the manoeuvre is detected, the manoeuvre model in variable-dimension is reconstructed only by the measurements at the start of the sliding window. Figs above show that the unsuitable reconstruction of manoeuvre model, the delay of manoeuvre detection in variable-dimension filter, and the linearisation errors of the variable-dimension filter with augmented rectilinear-acceleration states during the turn cause tracking error.

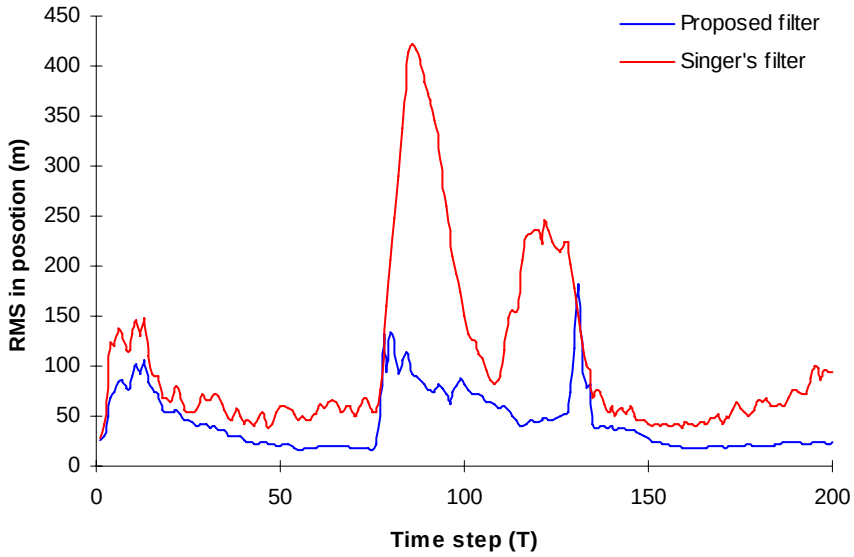


Fig. 6.32 RMS errors of proposed filter and Singer's filter in position

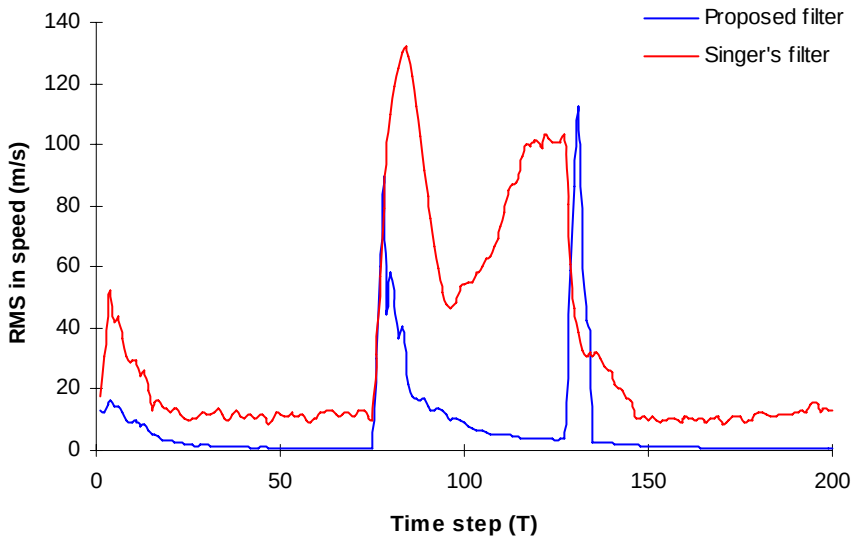


Fig. 6.33 RMS errors of proposed filter and Singer's filter in speed

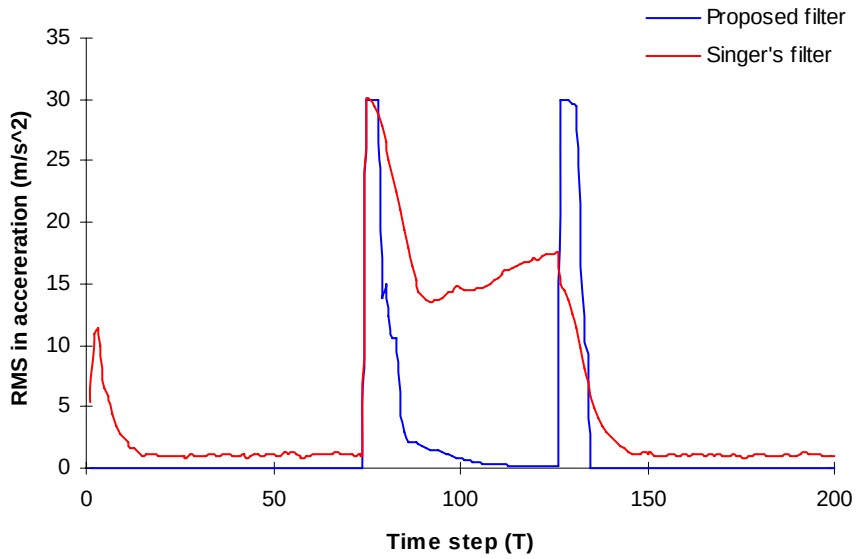


Fig. 6.34 RMS errors of proposed filter and Singer's filter in acceleration

Figs 6.32-6.34 show that the performance of proposed multiple-model filter is much better than that of Singer's filter during certain manoeuvre. The increased process noise covariance in Singer's filter is not able to cover the manoeuvre, causing tracking error. In very few steps around the steps of manoeuvre and of the end of manoeuvre, the performance of the proposed multiple-model filter is little worse than that of Singer's filter because the manoeuvre detection has a few steps delay and for some steps the tracking in proposed multiple-model filter comes from the initial steps estimates of the modified Singer's filter like in the case with rectilinear acceleration motion mentioned before.

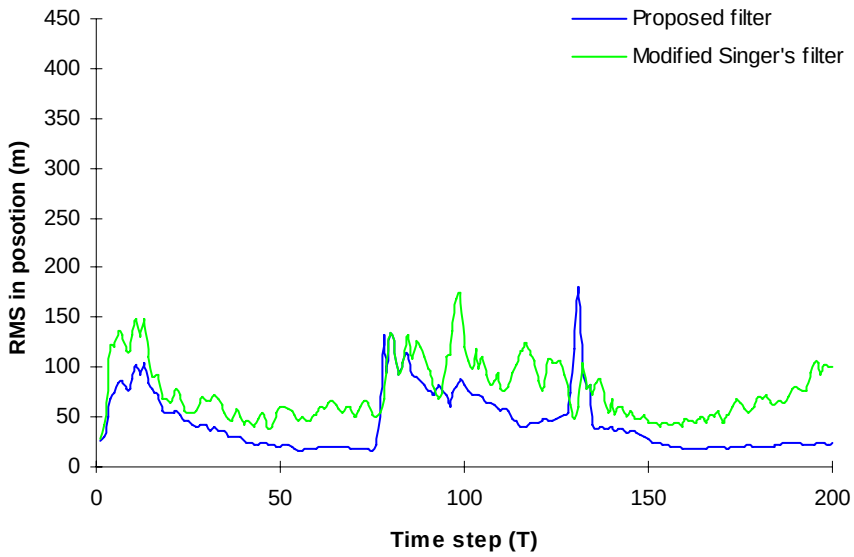


Fig. 6.35 RMS errors of proposed filter and modified Singer's filter in position

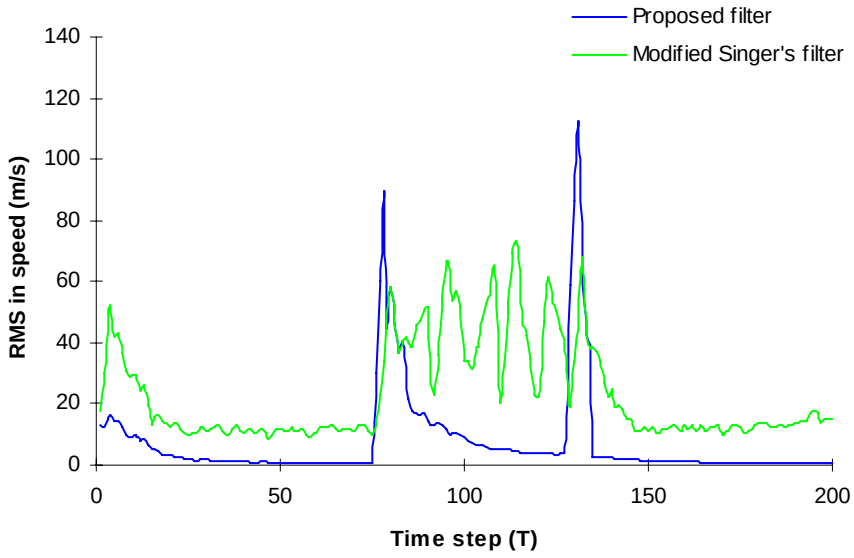


Fig. 6.36 RMS errors of proposed filter and modified Singer's filter in speed

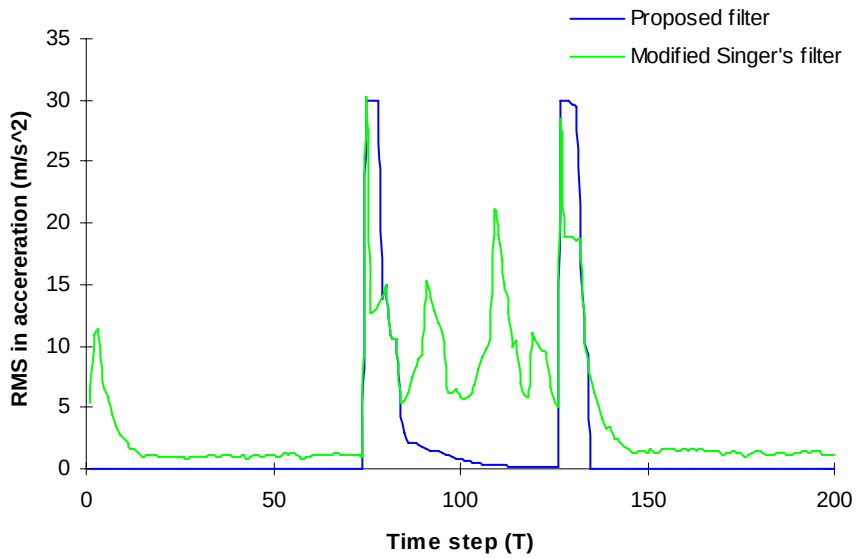


Fig. 6.37 RMS errors of proposed filter and modified Singer's filter in acceleration

Figs 6.35-6.37 show that the performance of proposed multiple-model filter is much better than that of modified Singer's filter during certain manoeuvre. Around the manoeuvre and around the end of manoeuvre, the modified Singer's filter gives the better performance than the multiple-model filter. In the case of certain manoeuvre, the proposed multiple-model filter uses the matched the filter for tracking, but around the manouvre and around the end of manoeuvre, because the manoeuvre detection has several steps delay, the original matched filter is not suitable for the manoeuvre, however the modified Singer's filter covers manoeuvre better because of increasing its process noise covariance.

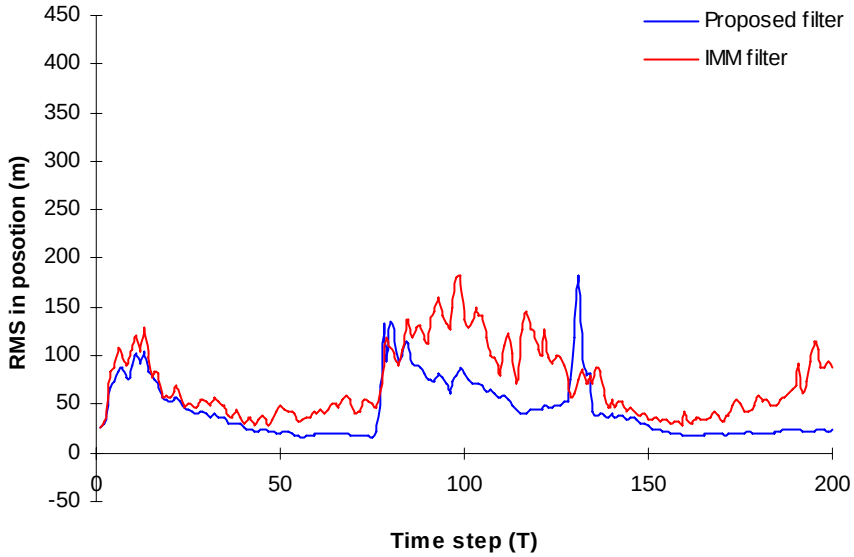


Fig. 6.38 RMS errors of proposed filter and IMM filter in position

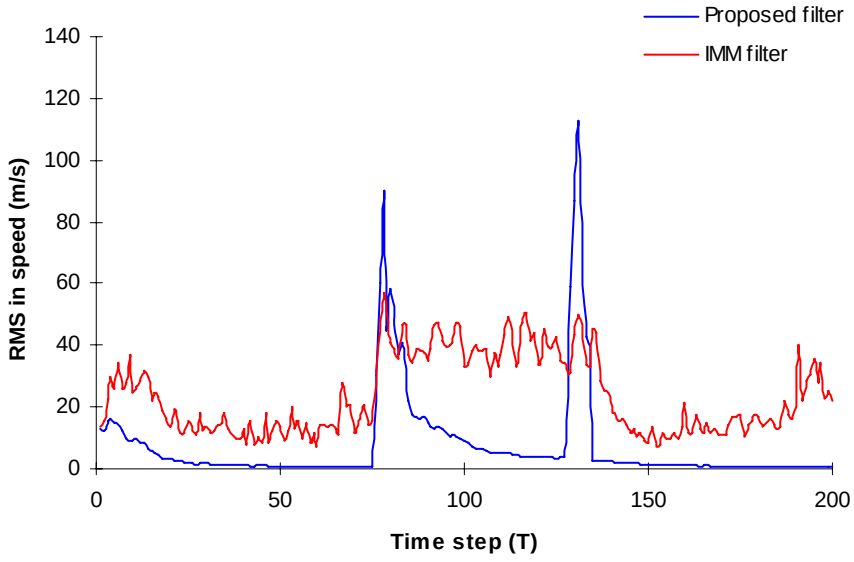


Fig. 6.39 RMS errors of proposed filter and IMM filter in speed

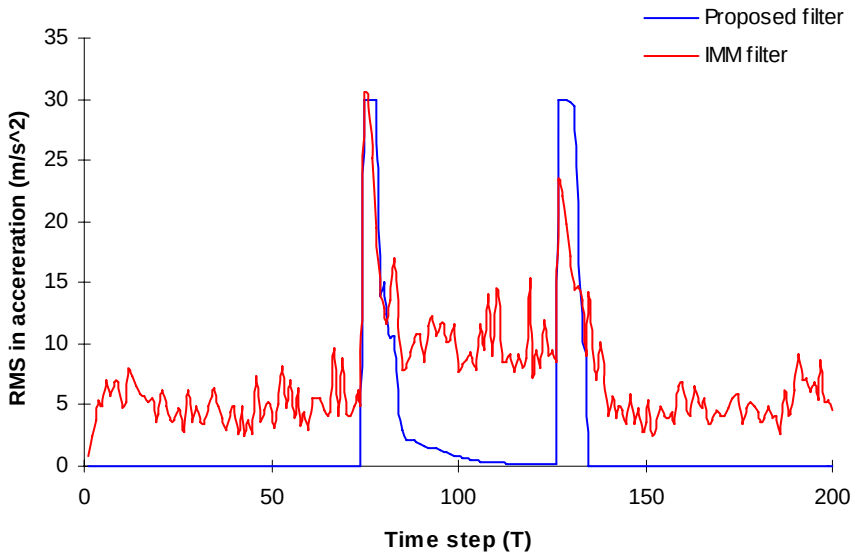


Fig. 6.40 RMS errors of proposed filter and IMM filter in acceleration

Figs 6.38-6.40 show that at few steps around the start of the manoeuvre and around the end of the manoeuvre, the IMM filter gives the better performance than the multiple-model filter. However, at most times, the performance of proposed multiple-model filter is better than that of IMM filter during the certain manoeuvre. Around the start of the manouvre and around the end of the manoeuvre, because the manoeuvre detection has several steps delay, the original matched filter is not suitable for the manoeuvre; but the IMM filter has the ability of self-adjusting to "switch" from one model to another and make it respond more quickly to the future manoeuvre. In the case of certain manoeuvre, the proposed multiple-model filter uses the matched filter for tracking; but the IMM filter is performed by several suboptimal filters.

Comparing the computation times of trackers

	Computation time(s)
Constant-velocity filter	0.296
Variable-dimension filter	0.856
Singer's Filter	0.494
Modified Singer's Filter	2.504
IMM	29.080
Proposed Filter	4.760

Table 6.4 Computation loads of trackers used in simulation comparison

6.3.3 Comparison of trackers with an agile target

The target's trajectory consists of several constant velocities, constant-speed turns and an non-constant-speed turn motion as the third example in Chapter 5.

Six trackers are tested here,

tracker	details of tracker
CV	constant-velocity-based Kalman filter
VDF	models: constant-velocity motion and rectilinear-acceleration manoeuvre detection: testing the statistic of normalised squared innovations window size: 5 samples for manoeuvre detection 3 samples for the end of manoeuvre detection process noise standard deviation of the manoeuvre filter: 5% of the estimated acceleration initial values of manoeuvre filter: provided by measurements at the start of the sliding window
Singer's Filter	manoeuvre correction time: 10s manoeuvre acceleration variance for target model: 3^2
Modified Singer's Filter	manoeuvre level estimation: input estimation manoeuvre detection: testing the statistic of normalised squared innovations and using modified Weston and Norton's detection with fixed-lag smoothing window size: 5 samples; fixed-lag: as in proposed filter target accelerations around estimated manoeuvre level: modelled by Singer's model manoeuvre correction time and manoeuvre acceleration variance: the same as for Singer's filter.
IMM	along-track accelerations used: $a_t=\{\pm 30.0,\pm 15.0,0.0\}\text{m/s}^2$ cross-track accelerations used: $a_n=\{\pm 100.0,\pm 50.0,0.0\}\text{m/s}^2$ with a probability of no change in manoeuvre of 0.9.
Proposed Filter	models: constant-velocity motion, modified Singer's model, straight line motion with an augmented state (with along-track acceleration), circular motion with an augmented state (cross-track acceleration), and curvilinear motion with augmented states (cross- and along-track accelerations) manoeuvre detection: using modified Weston and Norton's detection with fixed-lag smoothing the end of the third manoeuvre detection, fixed-lag is 10 samples; the others, fixed-lag is 5 samples initial values of manoeuvre: provided by measurements within the sliding window with modified Singer's filter decision of switch model: made by manoeuvre detection and estimated accelerations.

Table 6.5 Trackers used in simulation comparison

The initialization of the filters are the same as in the first example. The detail of initialization is not reiterated here.

The simulation results are shown in Figs 6.41-76. Figs 6.41, 6.42, 6.44, 6.46, 6.48, 6.49, 6.51, 6.53, 6.55, 6.56, 6.58, and 6.60 present one-run results of manoeuvre detection by using the modified Weston and Norton's detection with fixed-lag smoothing supposing the manoeuvre be curvilinear acceleration motion, circular motion, or straight-line acceleration motion. In the same way, Figs 6.43, 6.45, 6.47, 6.50, 6.52, 6.54, 6.57, 6.59, and 6.61 are shown the estimated accelerations assuming the manoeuvre be curvilinear acceleration motion by using augmented state models with cross- and along-track accelerations, circular motion by using augmented state models with cross-track acceleration, or straight-line acceleration motion by using augmented state models with along-track acceleration. Figs 6.62-6.64 show the performance of the proposed filter and constant-velocity filter in position, speed and acceleration. Figs 6.65-6.67 show the performance of the proposed filter and variable-dimension filter in position, speed and acceleration. Figs 6.68-6.70 show the performance of the proposed filter and Singer's filter in position, speed and acceleration. Figs 6.71-6.73 show the performance of the proposed filter and modified Singer's filter in position, speed and acceleration. Figs 6.74-6.76 show the performance of the proposed filter and IMM filter in position, speed and acceleration.

Table 6.6 shows the computation loads of trackers used.

A) For the first manoeuvre

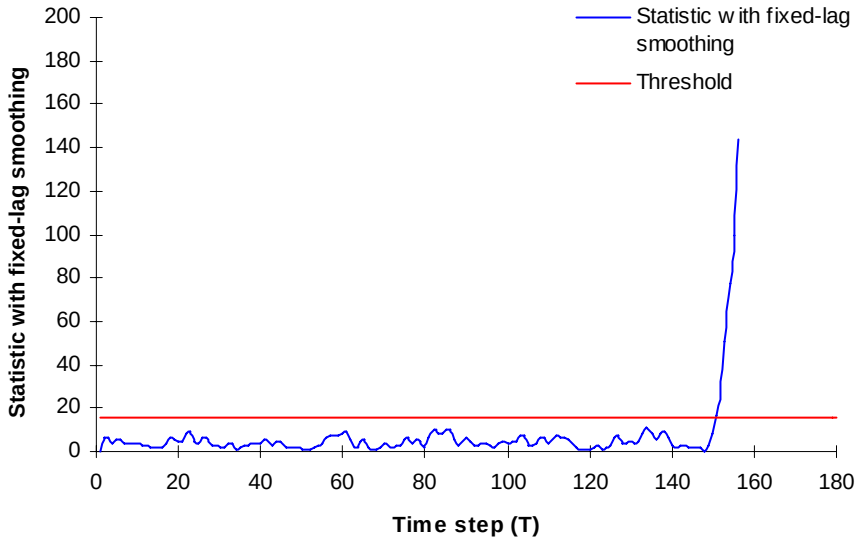


Fig. 6.41 Statistic with fixed-lag smoothing for the detection of manoeuvre start

The manoeuvre is detected at step 156 including the fixed lag.

The three manoeuvre filters are run and check to distinguish which manoeuvre occurs by using manoeuvre detection and estimating the accelerations in parallel, as follows.

A.1) Running the augmented state model with along- and cross-track accelerations

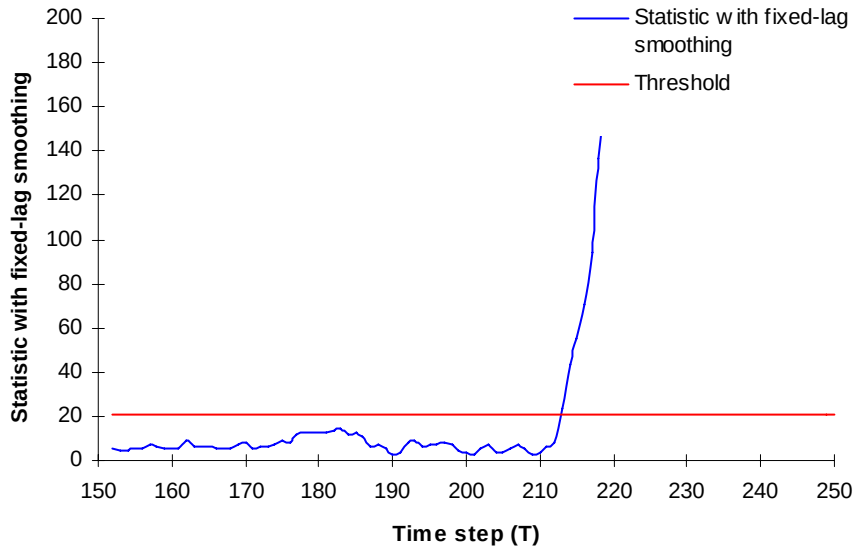


Fig. 6.42 Statistic with fixed-lag smoothing for the detection of manoeuvre end

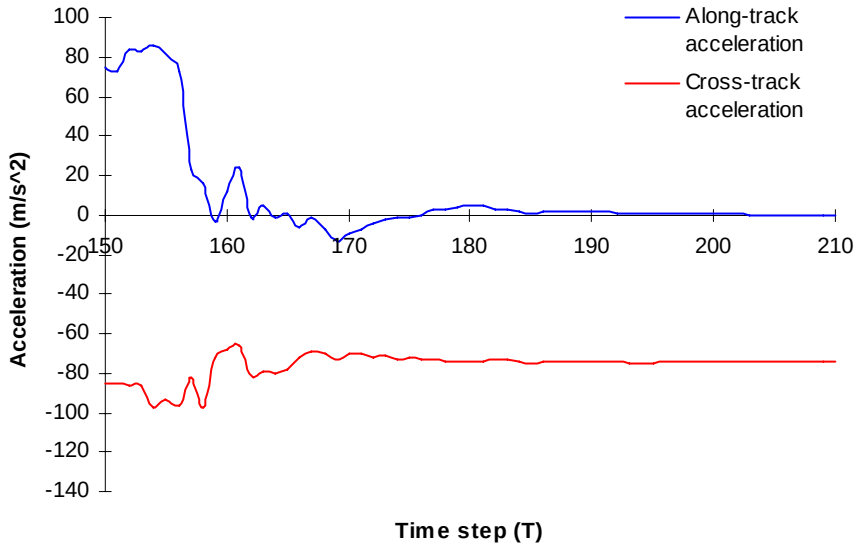


Fig. 6.43 Estimated along- and cross-track accelerations

From Figs 6.42 and 6.43, the end of manoeuvre is detected at step 218 including the fixed lag, and the along-track acceleration approaches zero and the cross-track acceleration approaches a constant around step 172.

A.2) Running the augmented state model with cross-track acceleration

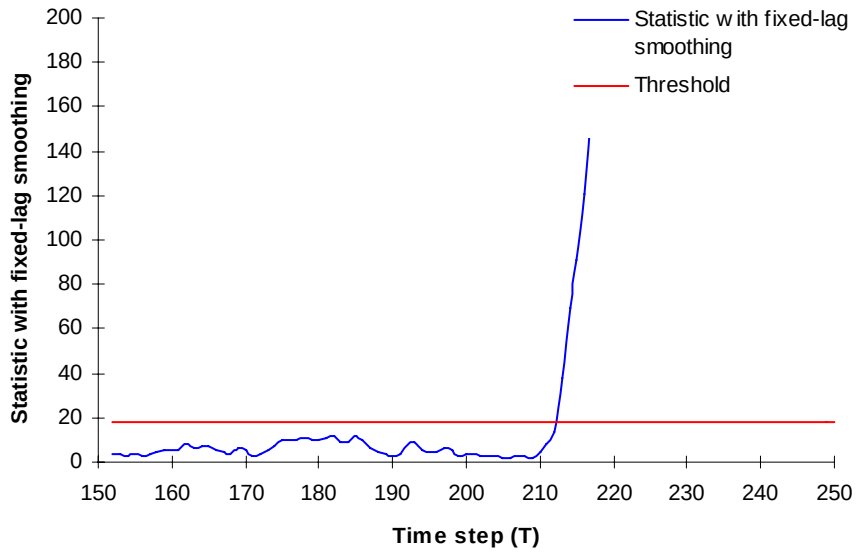


Fig. 6.44 Statistic with fixed-lag smoothing for the detection of manoeuvre end

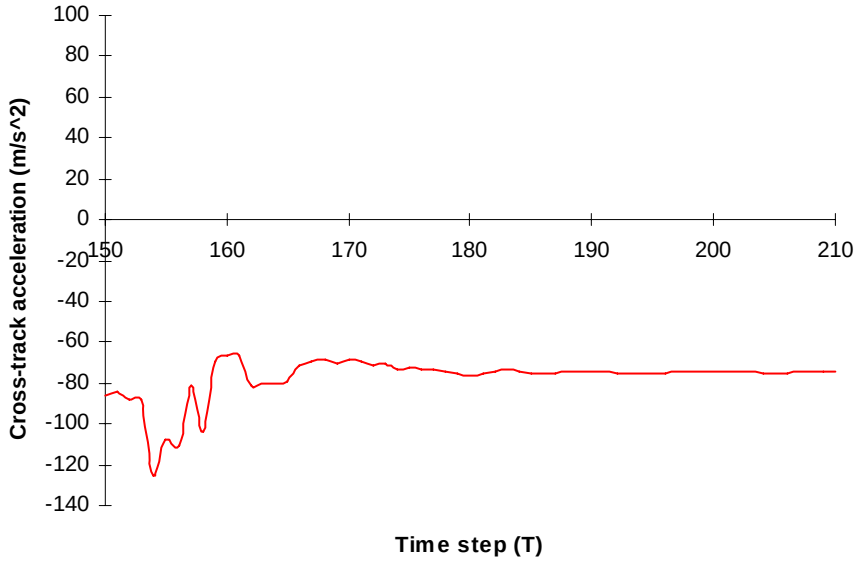


Fig. 6.45 Estimated cross-track acceleration

From Figs 6.44 and 6.45, the end of manoeuvre is detected at step 218 including the fixed lag, and the cross-track acceleration approaches towards to stable around step 172.

A.3) Running the augmented state model with along-track accelerations

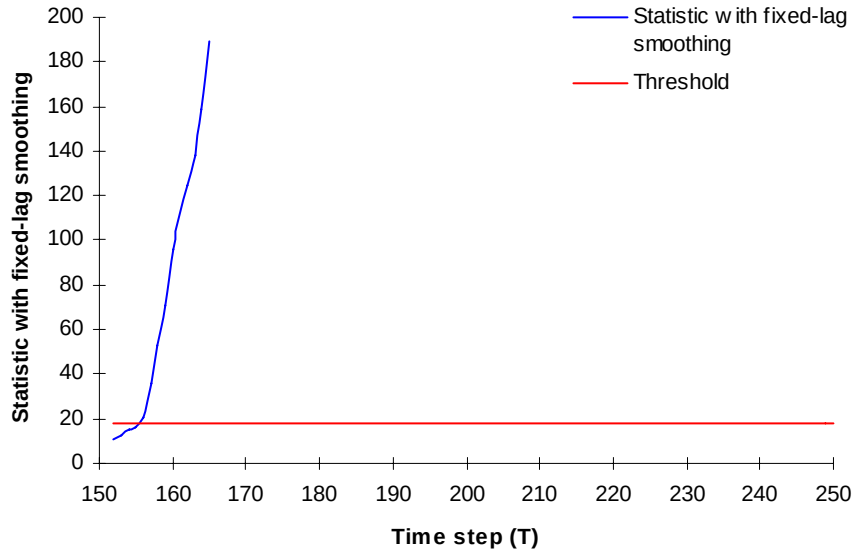


Fig. 6.46 Statistic with fixed-lag smoothing for the detection of manoeuvre end

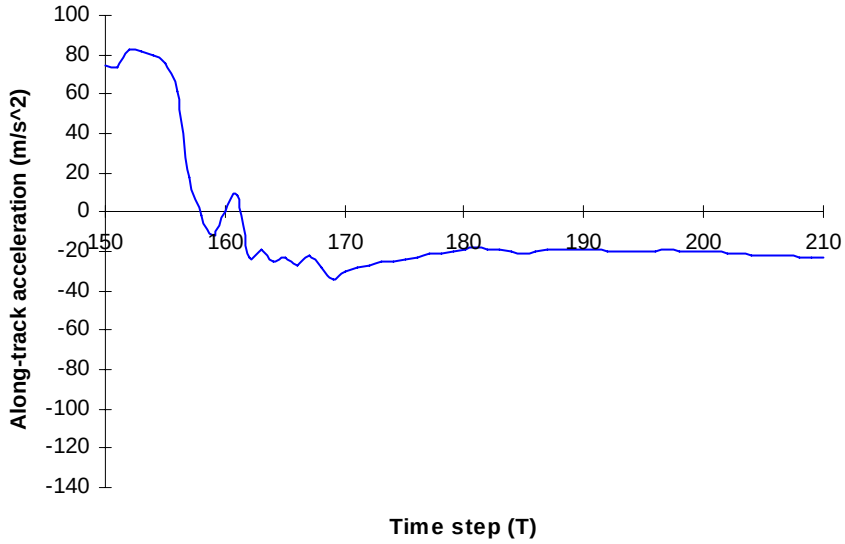


Fig.6.47 Estimated along-track acceleration

From Fig. 6.46, the manoeuvre detection shows that the assumed manoeuvre model is not fit for target motion immediately.

B) For the second manoeuvre

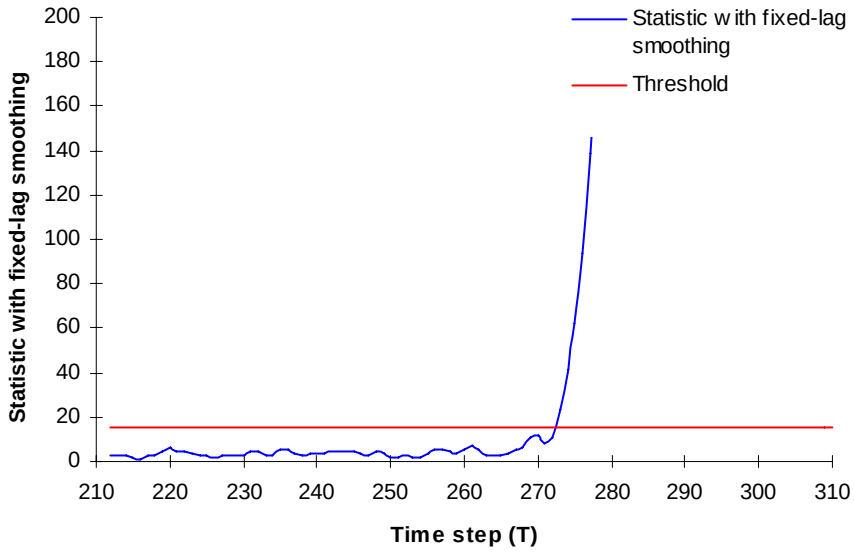


Fig. 6.48 Statistic with fixed-lag smoothing for the detection of manoeuvre start

The manoeuvre is detected at step 278 including the fixed lag.

The three manoeuvre filters are run and check to distinguish which manoeuvre happens by using manoeuvre detection and estimating the accelerations in parallel, as follows.

B.1) Running the augmented state model with along- and cross-track accelerations

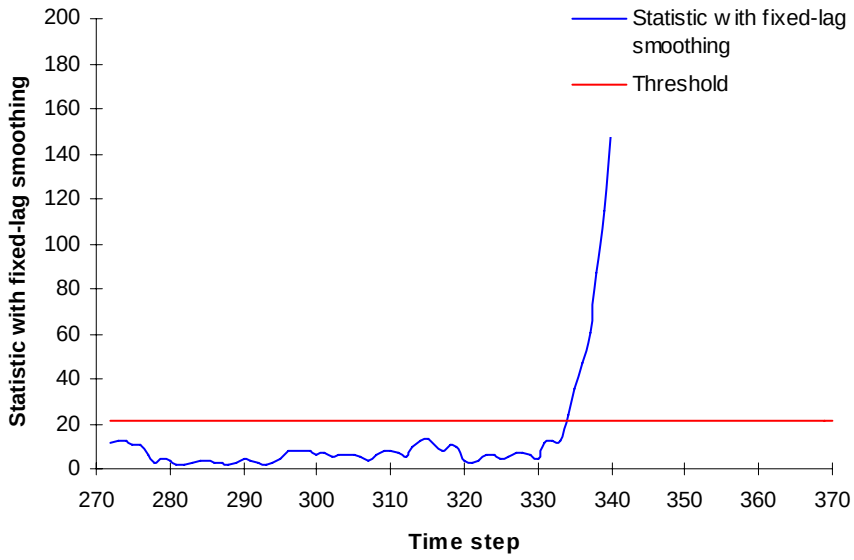


Fig. 6.49 Statistic with fixed-lag smoothing for the detection of manoeuvre end

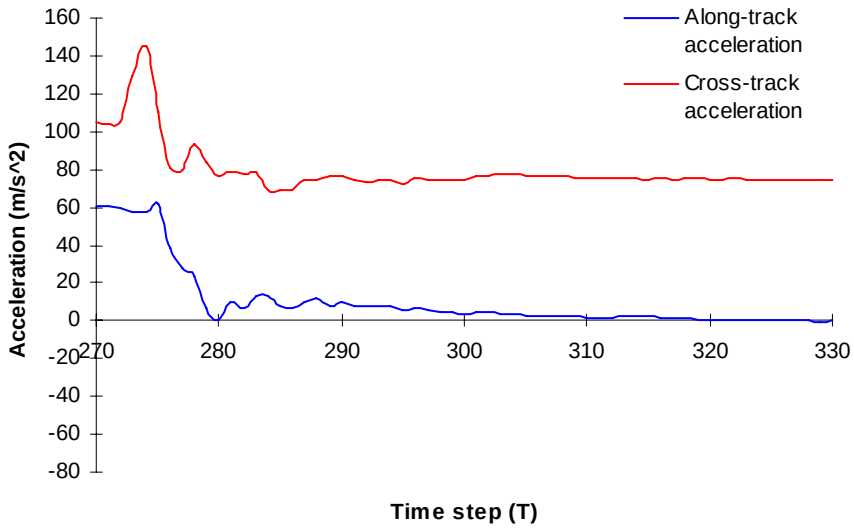


Fig. 6.50 Estimated along- and cross-track accelerations

From Figs 6.49 and 6.50, the end of manoeuvre is detected at step 339 including the fixed lag, and the along-track acceleration approaches zero and the cross-track acceleration approaches a constant around step 297.

B.2) Running the augmented state model with cross-track acceleration

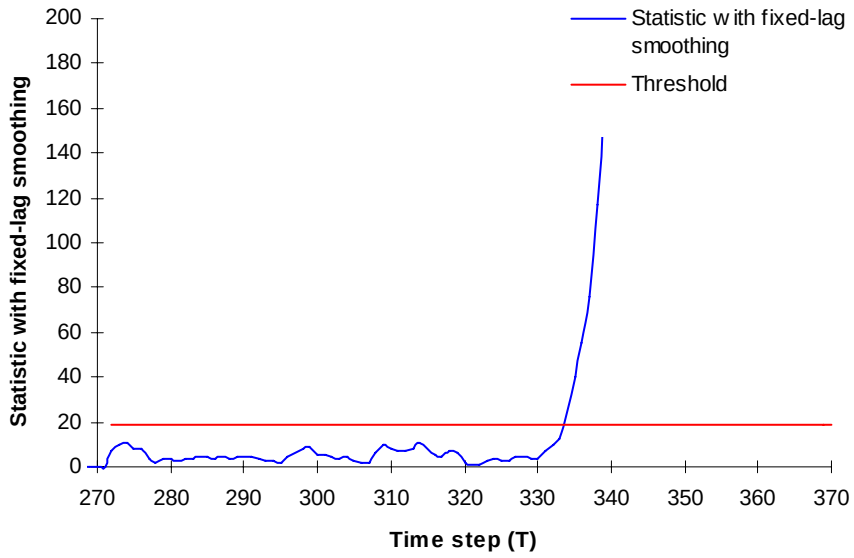


Fig. 6.51 Statistic with fixed-lag smoothing for the detection of manoeuvre end

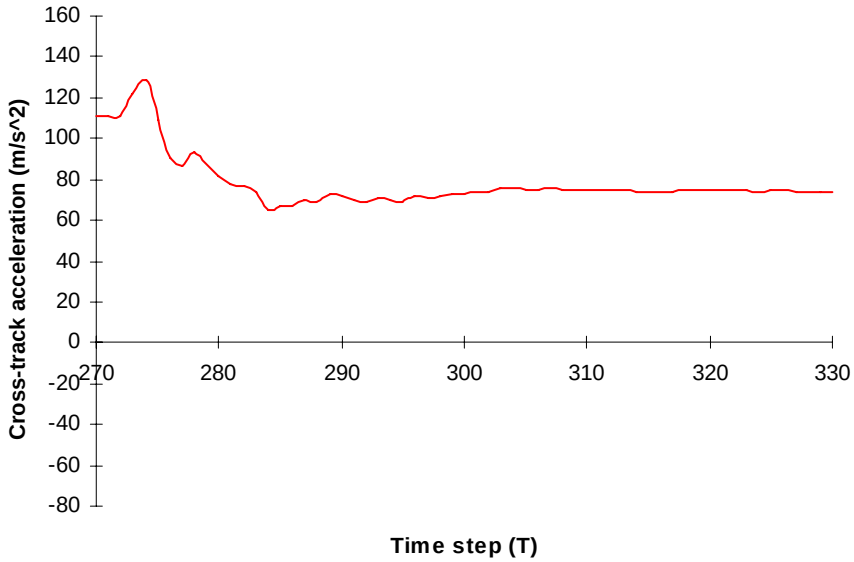


Fig. 6.52 Estimated cross-track acceleration

From Figs 6.51 and 6.52, the end of manoeuvre is detected at step 339 including the fixed lag, and the cross-track acceleration approaches a constant around step 297.

B.3) Running the augmented state model with along-track acceleration

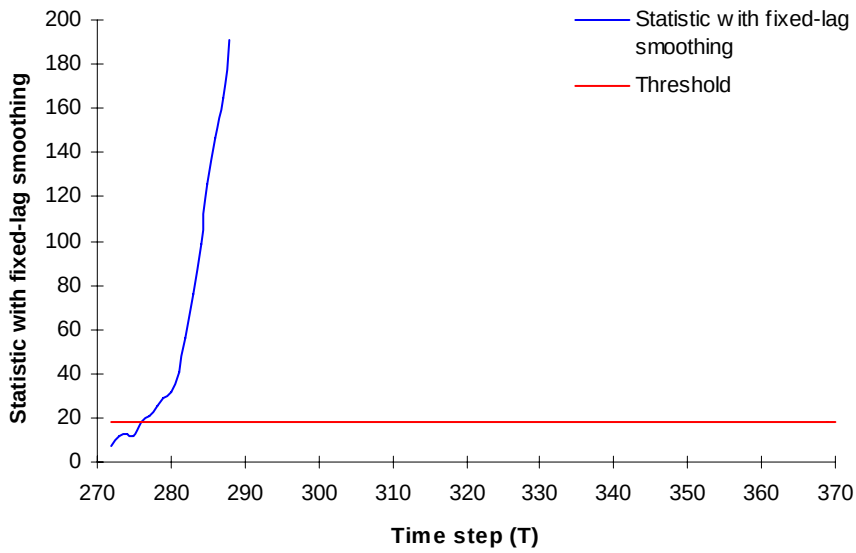


Fig. 6.53 Statistic with fixed-lag smoothing for the detection of manoeuvre end

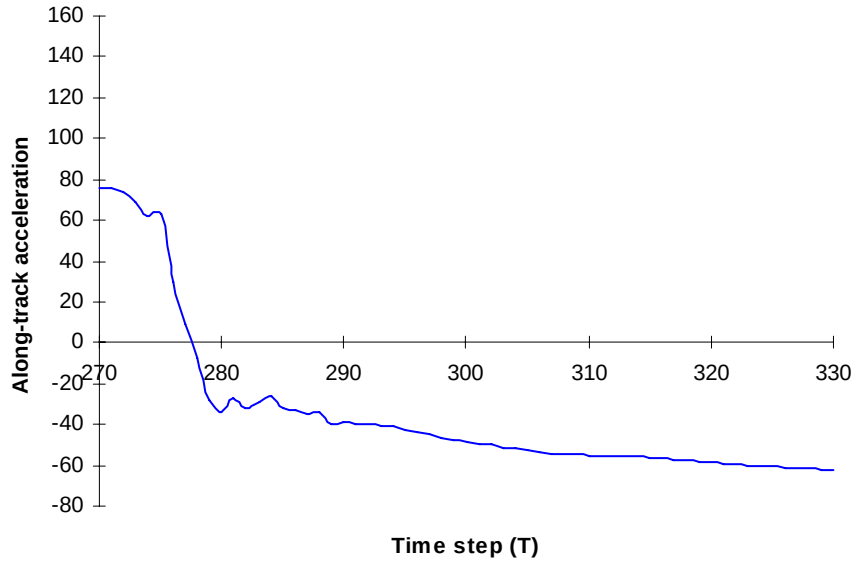


Fig. 6.54 Estimated along-track acceleration

From Fig. 6.53, the manoeuvre detection shows that the assumed manoeuvre model is not fit for target motion immediately.

C) For the third manoeuvre

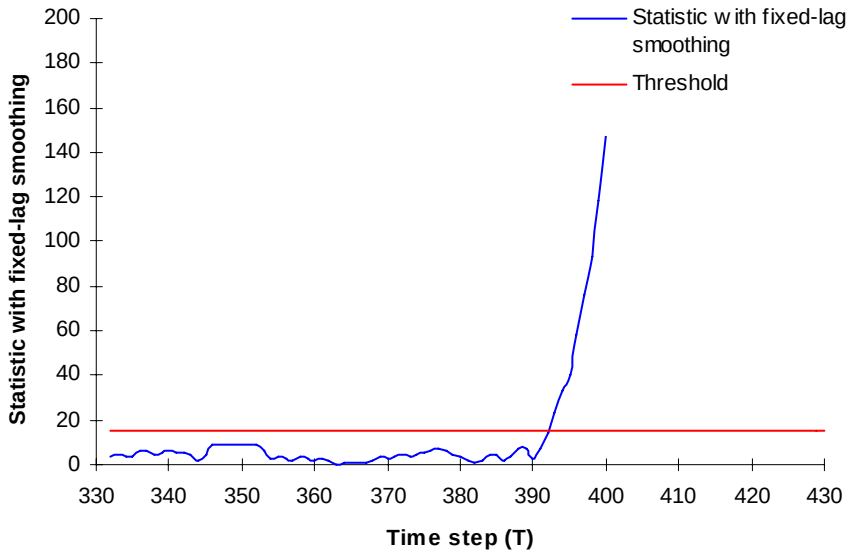


Fig. 6.55 Statistic with fixed-lag smoothing for the detection of manoeuvre start

The manoeuvre is detected at step 398 including the fixed lag.

The three manoeuvre filters are run and check to distinguish the manoeuvre by using manoeuvre detection and estimating the accelerations in parallel, as follows.

C.1) Running the augmented state model with along- and cross-track accelerations

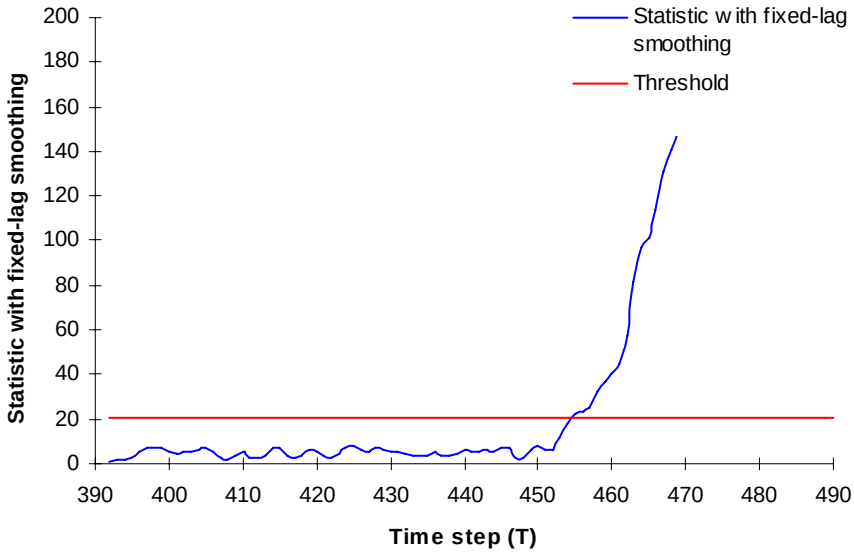


Fig. 6.56 Statistic with fixed-lag smoothing for the detection of manoeuvre end

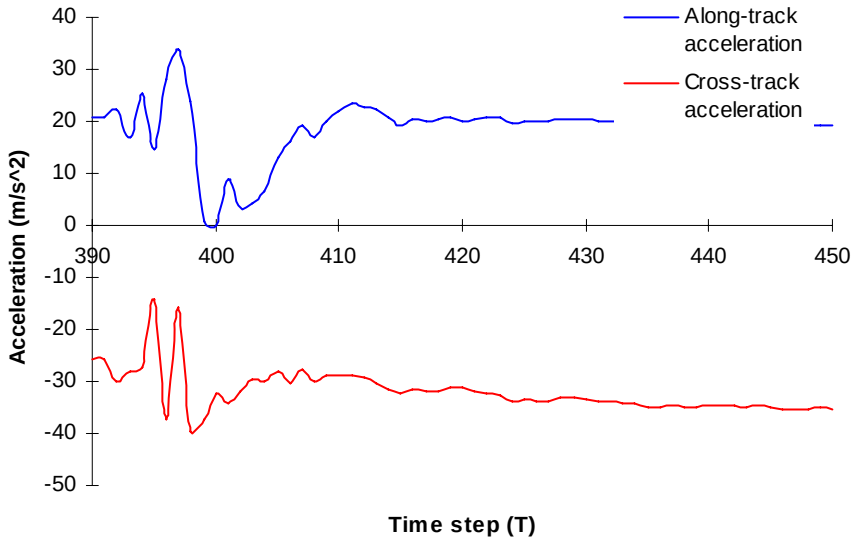


Fig.6.57 Estimated along- and cross-track accelerations

From Figs 6.56 and 6.57, the end of manoeuvre is detected at step 465 including the fixed lag, and the along- and cross-track accelerations approach constants around step 430

C.2) Running the augmented state model with cross-track acceleration

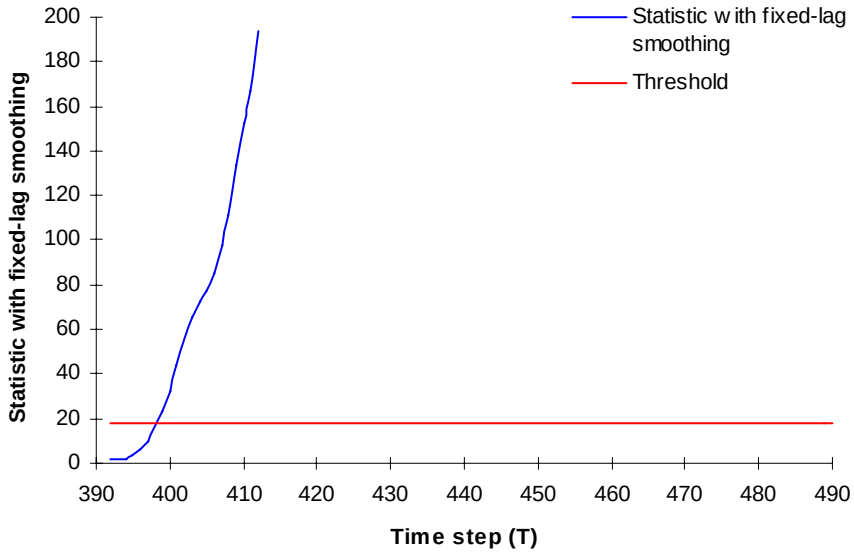


Fig. 6.58 Statistic with fixed-lag smoothing for the detection of manoeuvre end

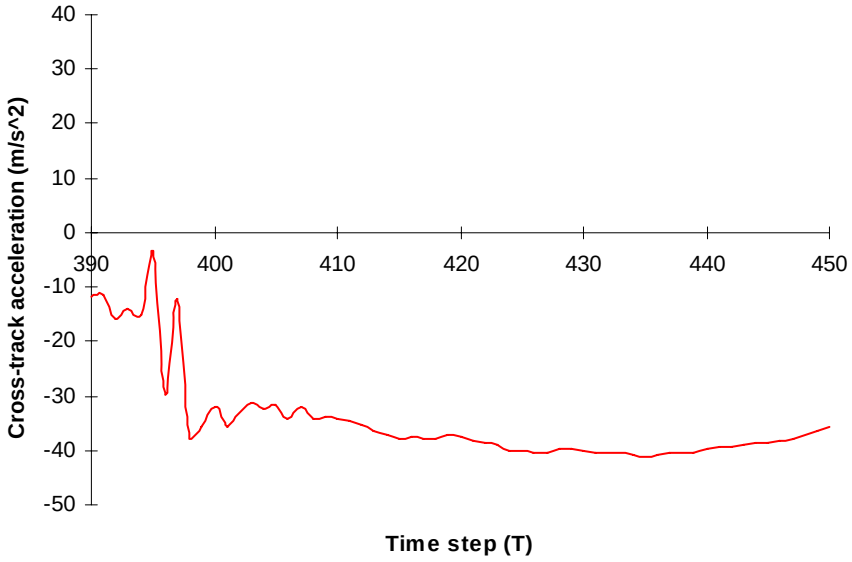


Fig. 6.59 Estimated cross-track acceleration

From Fig. 6.58, the manoeuvre detection shows that the assumed manoeuvre model is not fit for target motion immediately after step 409 including the fixed lag.

C.3) Running the augmented state model with along-track accelerations

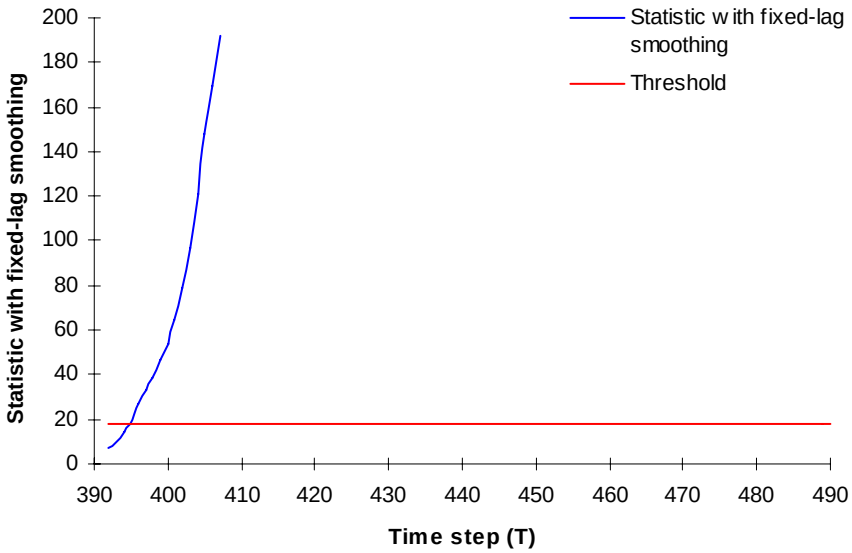


Fig. 6.60 Statistic with fixed-lag smoothing for the detection of manoeuvre end

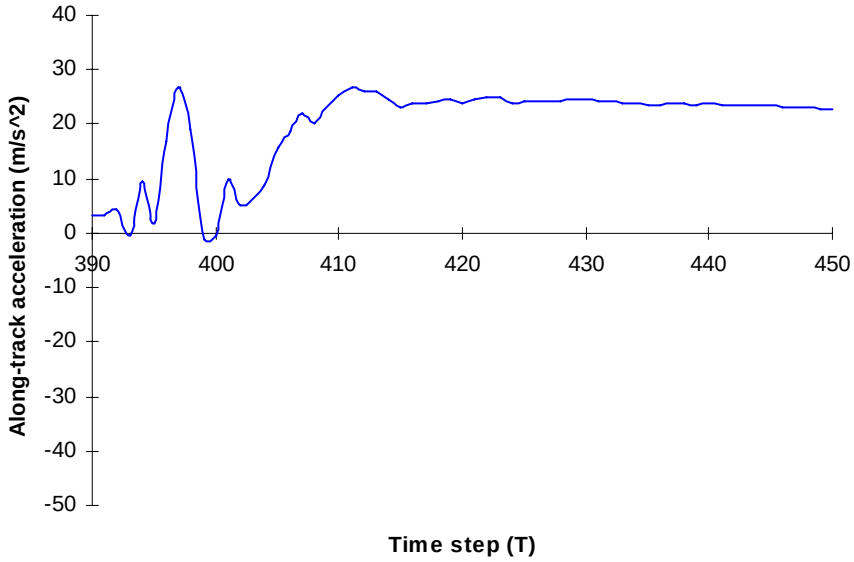


Fig. 6.61 Estimated along-track acceleration

From Fig. 6.60, the manoeuvre detection shows that the assumed manoeuvre model is not fit for target motion immediately after step 406 including the fixed lag.

From Figs and analysis above, the first manoeuvre starts and ends at steps 156 and 218 respectively, and the manoeuvre is decided to be circular motion at step 172. Thus the proposed multiple-model filter is that the unaugmented state model filter is used from the beginning to step 155, the modified Singer's filter is used from step 156 to 171, and the augmented state model filter with cross-track acceleration is used from step 172 to 217. After the manoeuvre end is sure by using manoeuvre detection from step 218 to 220 where the modified Singer's filter is used, the unaugmented state model is used until the new manoeuvre detected.

The second manoeuvre starts and ends at step 278 and 339 respectively, and the manoeuvre is decided to be circular motion at step 297. Thus the proposed multiple-model filter is that the unaugmented state model filter is used from the beginning to step

277, the modified Singer's filter is used from step 278 to 296, and the augmented state model filter with cross-track acceleration is used from step 297 to 338. After the manoeuvre end is sure by using manoeuvre detection from step 339 to 341 where the modified Singer's filter is used, the unaugmented state model is used until the third manoeuvre detected.

The third manoeuvre starts and ends at step 398 and 465 respectively, and the manoeuvre is decided to be curvilinear acceleration motion at step 409. Thus the proposed multiple-model filter is that the unaugmented state model filter is used from the beginning to step 397, the modified Singer's filter is used from step 398 to 408, and the augmented state model filter with cross-track and along-track accelerations is used from step 409 to 464. After the manoeuvre end is sure by using manoeuvre detection from step 465 to 467 where the modified Singer's filter is used, the unaugmented state model is used until the end of tracking.

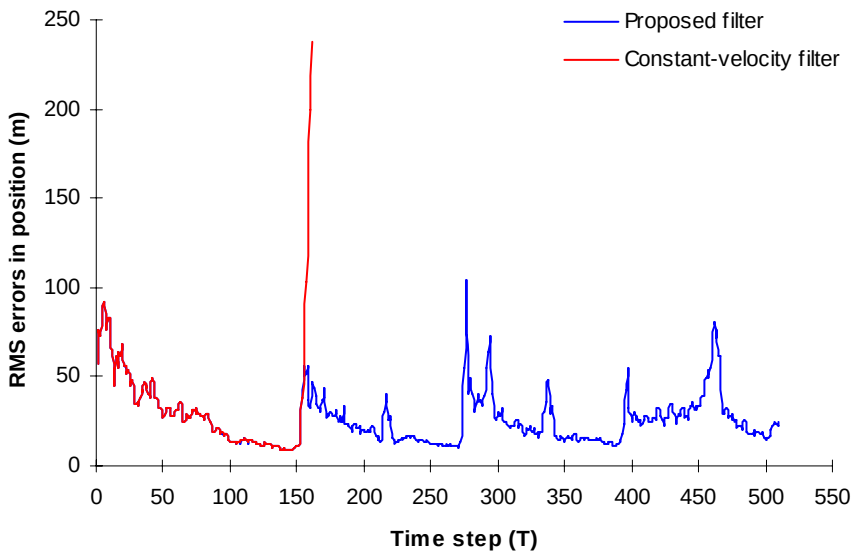


Fig. 6.62 RMS errors of proposed filter and constant-velocity filter in position

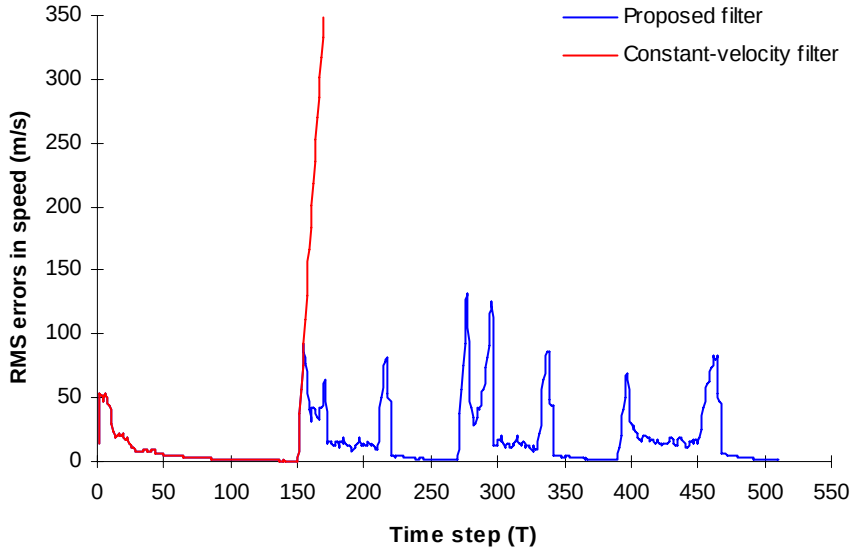


Fig. 6.63 RMS errors of proposed filter and constant-velocity filter in speed

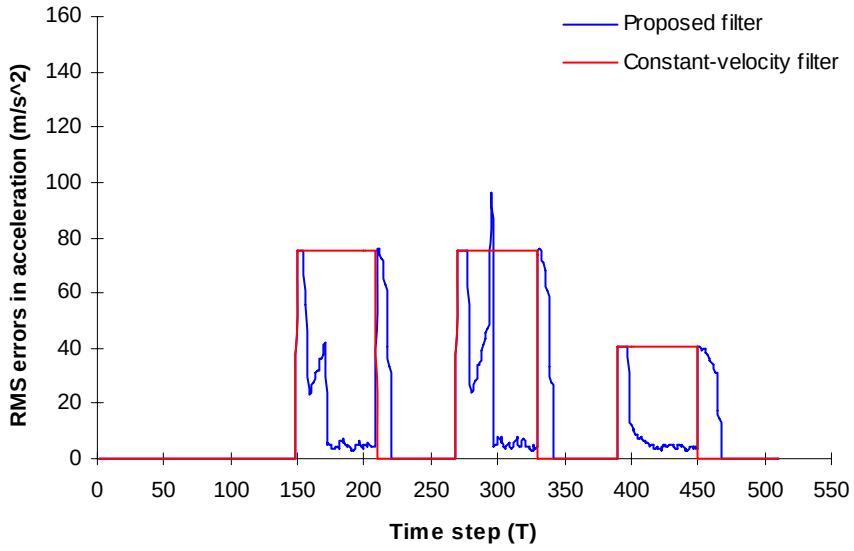


Fig. 6.64 RMS errors of proposed filter and constant-velocity filter in acceleration

Figs 6.62-6.64 show that the performance of proposed multiple-model filter is much better than that of constant-velocity filter, which is similar to the case with rectilinear acceleration motion mentioned before. In the present of manoeuvre, the constant-velocity model is not able to match the target motion, however, the proposed multiple-model filter uses matched target motion model in most steps, even though in the other

few steps of uncertain manoeuvres, the modified Singer's filter is used to provide safeguards for tracking.

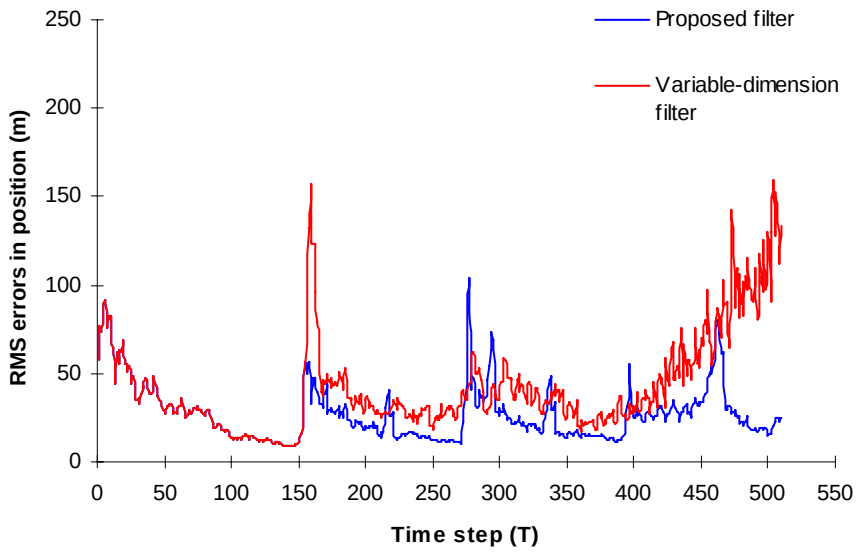


Fig. 6.65 RMS errors of proposed filter and variable-dimension filter in position

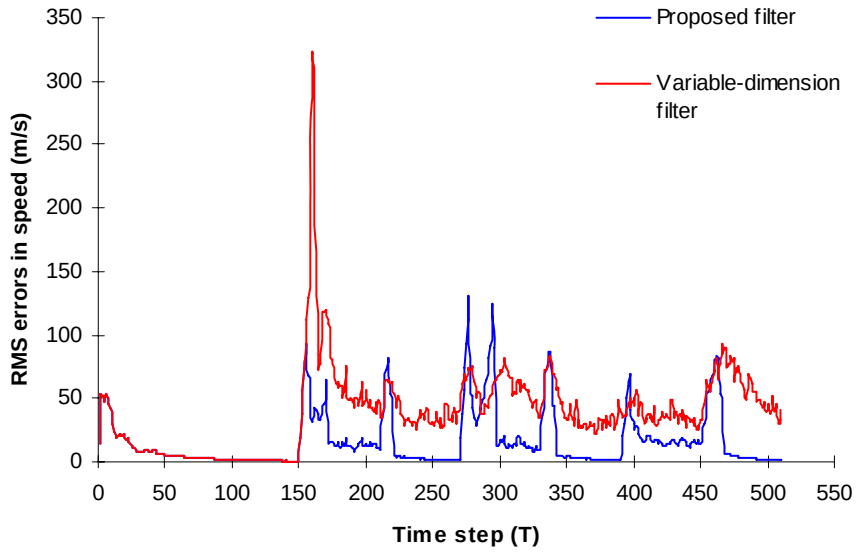


Fig. 6.66 RMS errors of proposed filter and variable-dimension filter in speed

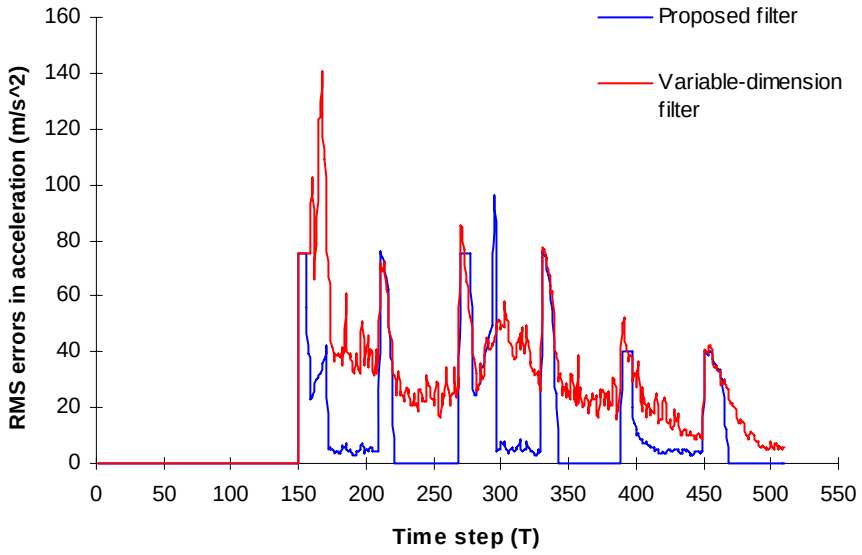


Fig. 6.67 RMS errors of proposed filter and variable-dimension filter in acceleration

Figs 6.65-6.67 show that the performance of proposed multiple-model filter is much better than that of variable-dimension filter, which is similar to the case with circular motion mentioned before. When the manoeuvre is detected, the manoeuvre model in variable-dimension filter is reconstructed only by the measurements at the start of the sliding window. Figs above show that the unsuitable reconstruction of manoeuvre model cause tracking error, and also the linearisation errors of the variable-dimension filter with augmented rectilinear-acceleration states during the turns might increase tracking error.

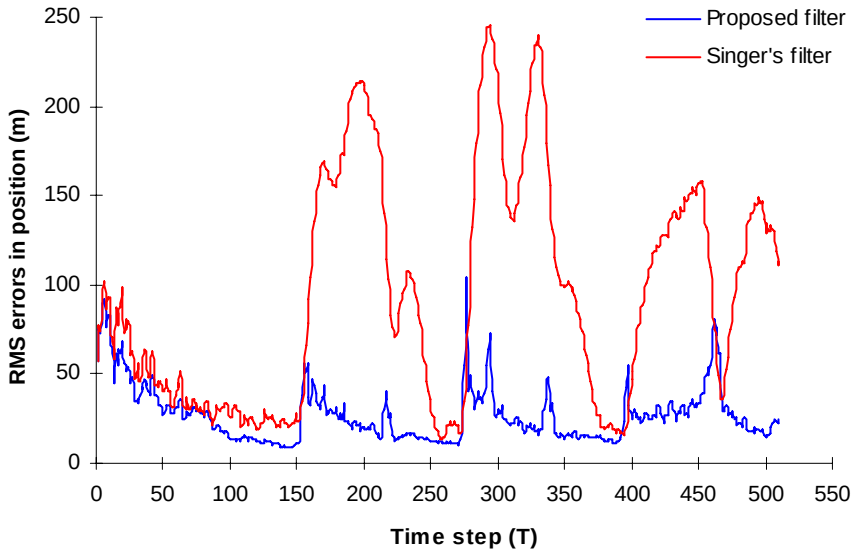


Fig. 6.68 RMS errors of proposed filter and Singer's filter in position

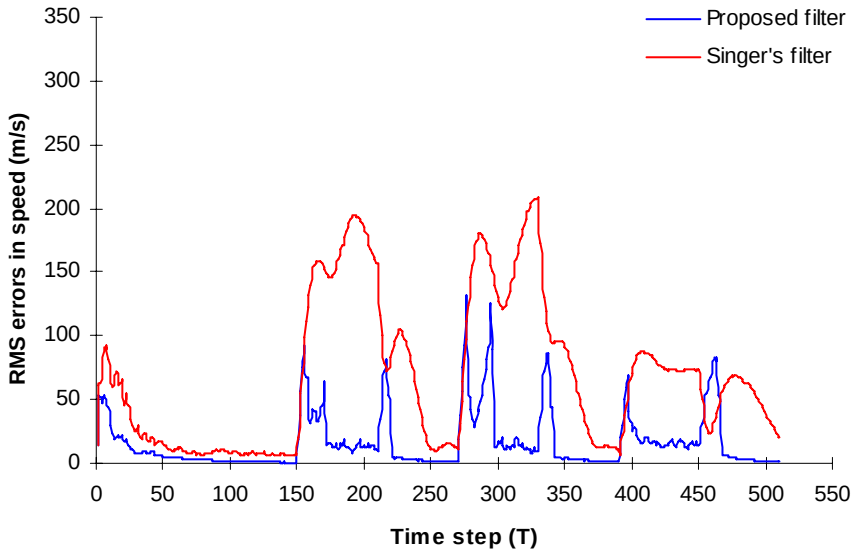


Fig. 6.69 RMS errors of proposed filter and Singer's filter in speed

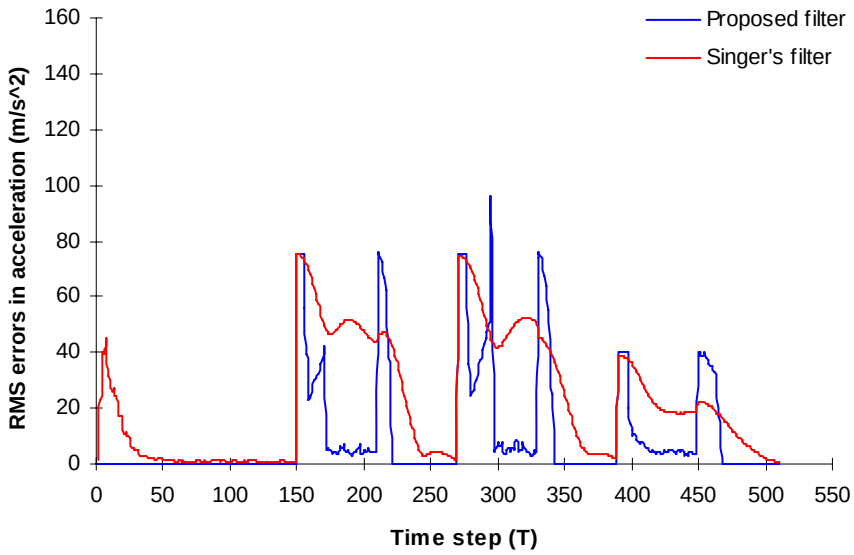


Fig. 6.70 RMS errors of proposed filter and Singer's filter in acceleration

Figs 6.68-6.70 show that the performance of proposed multiple-model filter is much better than that of Singer's filter at most times. The increased process noise covariance in Singer's filter is not able to cover the manoeuvre, causing tracking error. In very few steps around the steps of manoeuvre and of the end of manoeuvre, the performance of the proposed multiple-model filter is little worse than that of Singer's filter because the manoeuvre detection has few steps delay and for some steps the tracking in proposed multiple-model filter comes from the initial estimate of forcing estimates of the modified Singer's filter.

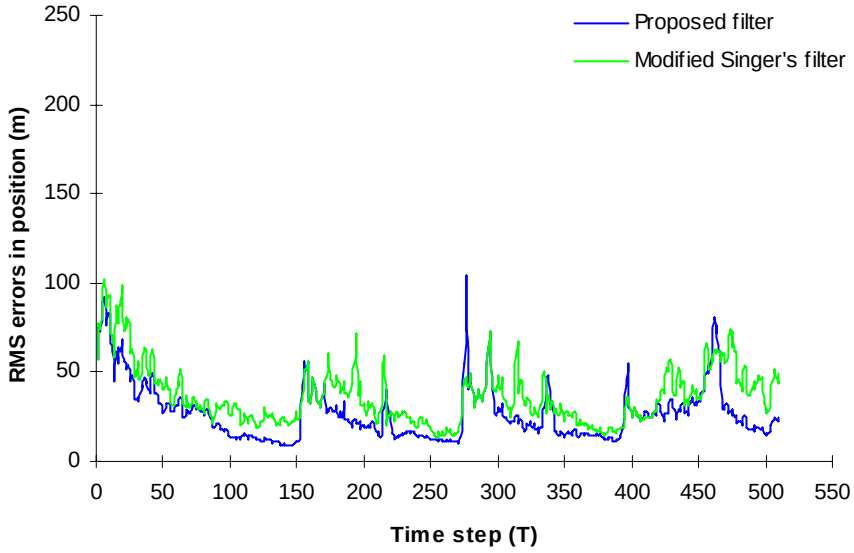


Fig. 6.71 RMS errors of proposed filter and modified Singer's filter in position

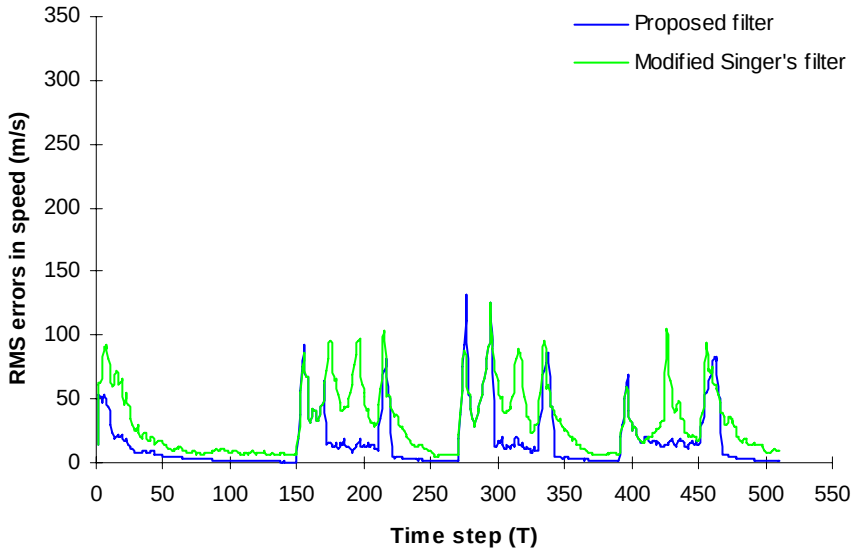


Fig. 6.72 RMS errors of proposed filter and modified Singer's filter in speed

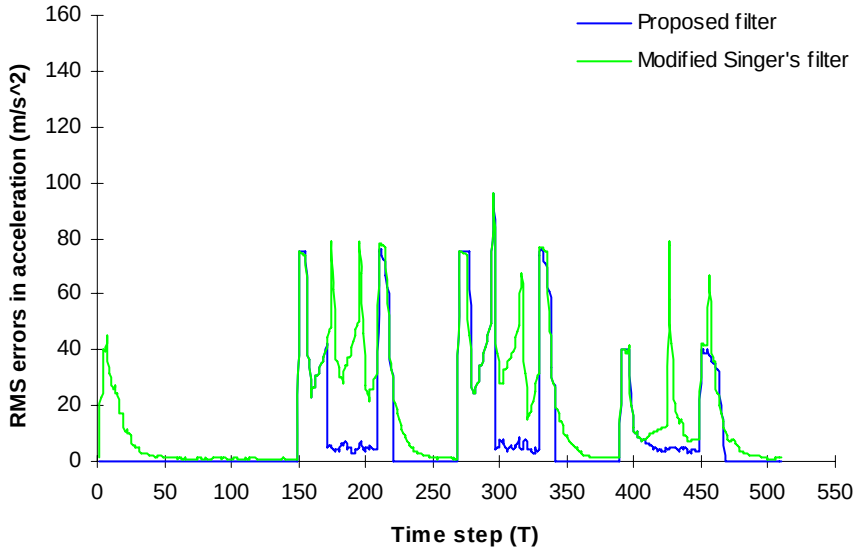


Fig. 6.73 RMS errors of proposed filter and modified Singer's filter in acceleration

Figs 6.71-6.73 show that the performance of proposed multiple-model is much better than that of modified Singer's filter during the certain manoeuvre. Around the manoeuvre and around the end of manoeuvre, the modified Singer's filter gives the better performance than the multiple-model filter. In the case of certain manoeuvre, the proposed multiple-model filter uses the matched filter for tracking, but around the manouvre and around the end of manoeuvre, because the manoeuvre detection has several steps delay, the original matched filter is not suitable for the manoeuvre, but the modified Singer's filter covers manoeuvre better because of increasing its process noise covariance.

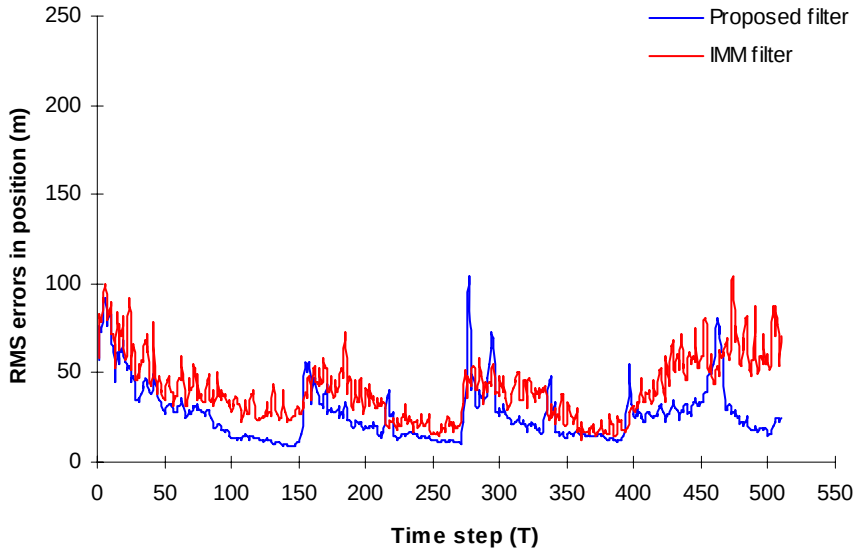


Fig. 6.74 RMS errors of proposed filter and IMM filter in position

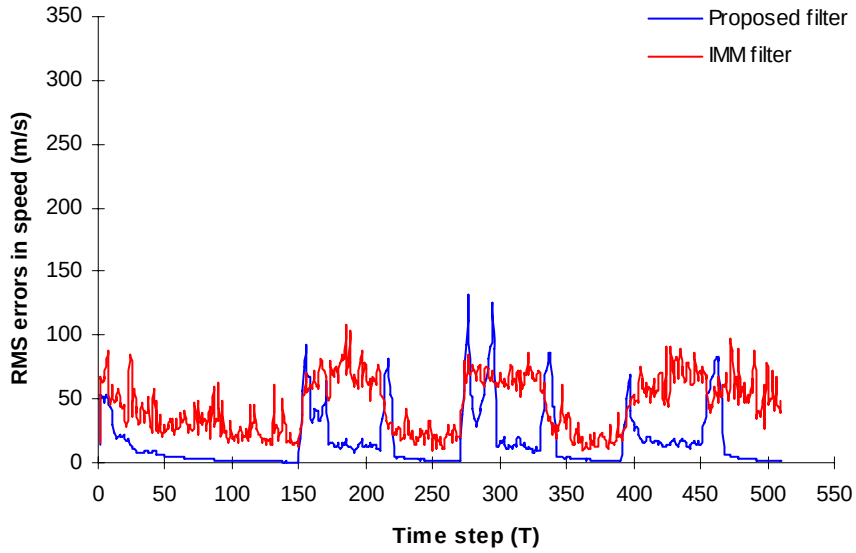


Fig. 6.75 RMS errors of proposed filter and IMM filter in speed

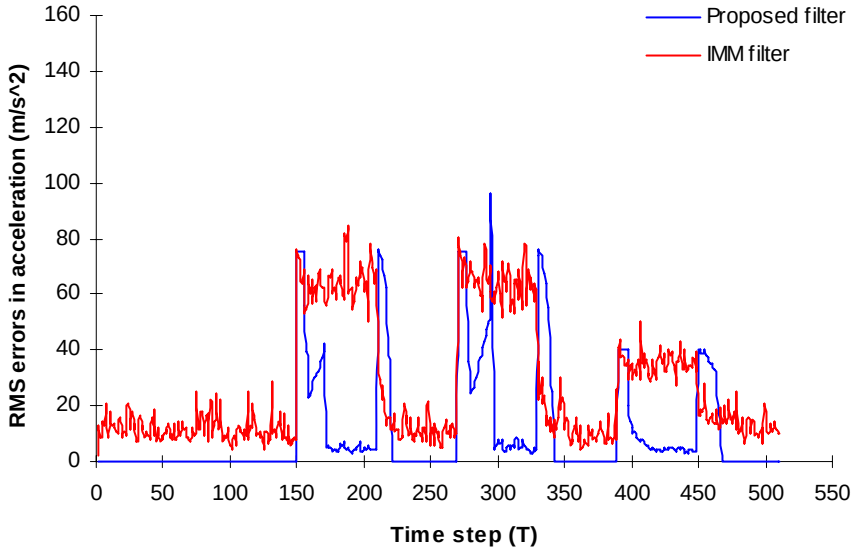


Fig. 6.76 RMS errors of proposed filter and IMM filter in acceleration

Figs 6.74-6.76 show that the performance of proposed multiple-model filter is much better than that of IMM filter during the certain manoeuvre. Around the start of the manoeuvre and around the end of the manoeuvre, the IMM filter gives the better performance than the multiple-model filter. Around the start of the manouvre and around the end of the manoeuvre, because the manoeuvre detection has several steps delay, the original matched filter is not suitable for the manoeuvre, but the IMM filter has the ability of self-adjusting to "switch" from one model to another and make it fast respond to the future manoeuvre. However, in most times where the manoeuvre is certain, the proposed multiple-model filter uses the matched the filter for tracking, but the IMM filter is performed by several suboptimal filters.

Comparing the computation times of trackers

	Computation time(s)
Constant-velocity filter	1.374
Variable-dimension filter	3.602
Singer's Filter	2.310
Modified Singer's Filter	8.808
IMM	74.040
Proposed Filter	19.583

Table 6.6 Computation loads of trackers used in simulation comparison

6.4 Discussions

The simulations have shown the performance of new multiple-model filter in three typical cases (rectilinear acceleration motion with constant accelerations in *x*- and *y*-direction, circular motion with a constant cross-track acceleration, and curvilinear motion with constant cross- or/and along-track accelerations). The manoeuvre is very complex, not just constant changes in accelerations. If the acceleration changes are in some range during the manoeuvre, how about the performance of the new multiple-model filter? In fact, for the second case (circular motion with a constant cross-track acceleration) and the third case (curvilinear motion with constant cross- or/and along-track accelerations), the changes of acceleration in *x*-direction and *y*-direction are in some range. The acceleration changes in *x*-direction and *y*-direction for the second case are shown in Fig. 6.77 and 6.78 as follows:

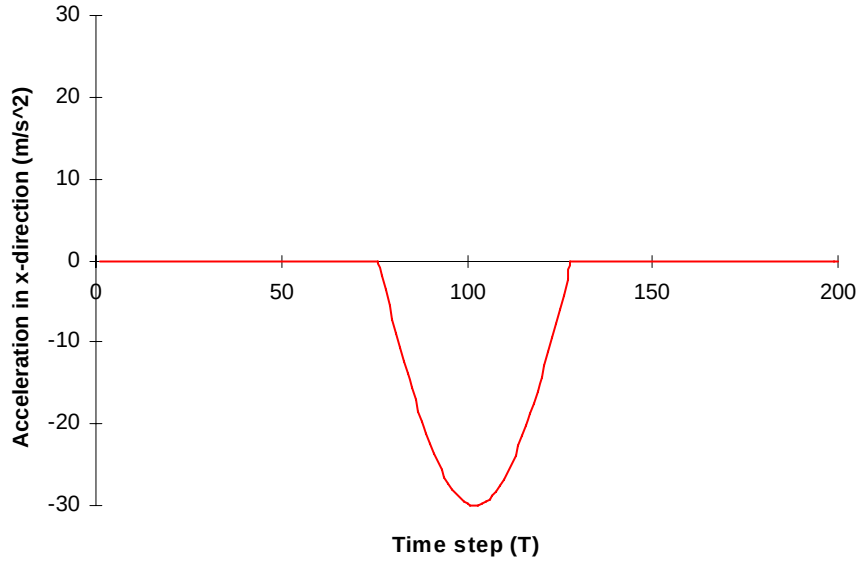


Fig. 6.77 Acceleration in *x*-direction for circular motion

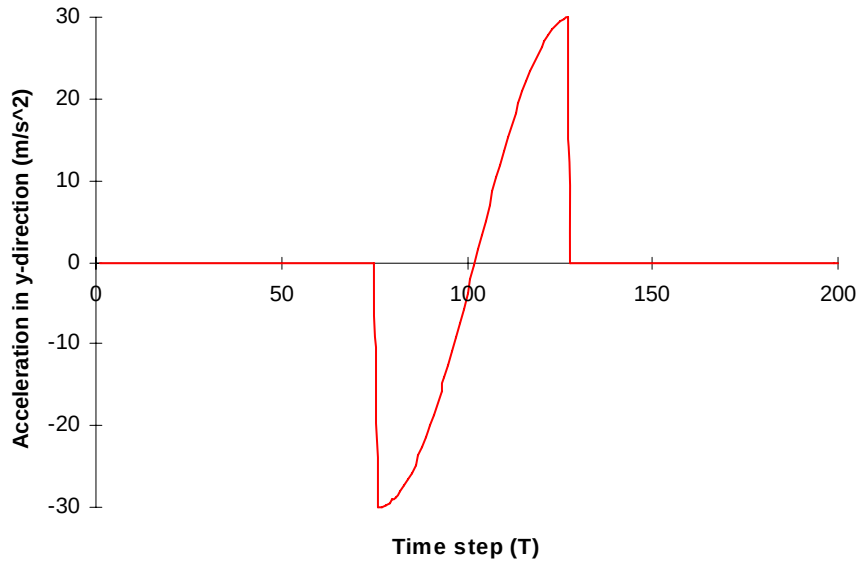


Fig. 6.78 Acceleration in *y*-direction for circular motion

Thus, the new multiple-model filter is suitable for the range change in acceleration. People may have a doubt about construction of manoeuvre models. If the manoeuvre models are constructed improperly, the manoeuvre detection would be difficult. Now

we imagine that the manoeuvre models are not fit for actual motion and the manoeuvre has not been detected, then the tracking is provided by modified Singer's filter. Thus for the worst case, the tracking can be safeguarded by modified Singer's filter at least, the performance of which is comparable to IMM filter but with significantly less computational load.

6.5 Summary

This chapter gave the detailed new multiple-model filter, and also compared the performance of the proposed filter with trackers in preceding chapters via three typical examples.

Constant-velocity filter is the simplest, and also has the worst performance but the lowest computational load. During the manoeuvre, the constant-velocity model is not able to match the target motion, causing the worst performance.

Variable-dimension filter gives a poor performance, because the manoeuvre model in variable-dimension is reconstructed only by the measurements at the start of the sliding window but there usually exists a difference between the actual and the assumed manoeuvre and also the linearisation errors of the variable-dimension filter with augmented rectilinear-acceleration states during the turn might increase tracking error.

Singer's filter is simple, and its computation load is low, but when the constructed process noise covariance is not able to cover the manoeuvre, the track would be lost during manoeuvres.

Modified Singer's filter improves Singer’s filter, its computation load is heavier than Singer’s filter but significantly less than IMM filter. During manoeuvre changes or low

level manoeuvre, the modified Singer's filter tracks manoeuvring targets well. Thus, this work uses the modified Singer's filter as a part of new multiple-model filter to provide the tracking during uncertain manoeuvres.

IMM filter has been very popular for tracking and is thought be the most robust, however its computation is heaviest. The construction of IMM filter could also affect its performance, such as increasing the number of filters to increase the possibility of the real manoeuvre model being included, causing more computation; decreasing the number of filters to make the correct filter to be dominated risks an incorrect filter being dominant, causing loss of track.

Proposed multiple-model filter provides best performance but has a modest computational load. The quick manoeuvre detection and reasonable induction via the estimated acceleration make it possible for the multiple-model filter to use matched target motion models for tracking at most times. In the few steps where the manoeuvre is uncertain, the modified Singer's filter is used to provide safeguards for tracking.

CHAPTER 7

FURTHER IMPROVEMENT FOR THE MULTIPLE-MODEL TRACKING FILTER

The preceding chapters have focused on linear motion models and optimal filtering, *i.e.*, estimation of the process at the same time as the current measurement. However, the smoothing algorithm uses additional "future" measurements to improve estimation accuracy. This chapter presents the further improvement for the new multiple-model filter by using the fixed-lag smoothing technique.

7.1 Introduction

Optimal smoothing (retrospective state estimation) has a long history. The smoothing problem was originally solved by Wiener in the 1940s. However, it receives less attention than optimal filtering because its recursive implementation is more complicated than that of filtering. But, optimal smoothing is potentially useful to help identify target type, resolve manoeuvre ambiguities and reconstruct tracks. Smoothing has been applied since the early 1960s. In 1965, Rauch et al. (1965) treated an orbit determination problem involving estimation of track position for a satellite in a nominally circular orbit. The smoothed position error variance was factors of 5-8 times less than that obtained via filtering. In 1968, Bryson and Mehra (1968) examined filtering and smoothing of data from a ship inertial navigation system. The most significant result in their study was that for continuous velocity data and a position fix every 1.5 hours over a 3 hours data span, the RMS error in estimating gyro drift using smoothing was roughly half of the RMS error using filtering. In 1971, Nash et al.

(1971) examined the use of smoothing in the testing and evaluation of inertial navigation systems. This has been done at both the component and system level, and includes (1) gyro testing, (2) system testing under laboratory conditions, (3) identification of component failure during a mission, and (4) post-mission analysis. These works are based on fixed-interval or fixed-point smoothing technique with offline processing.

More recently, Chen et al. (2000) and Ronald et al. (1996) use fixed-lag smoothing in conjunction with an IMM filter and achieve significant improvement, comparing with IMM filter alone. Chen et al. (2000) also quote a example to demonstrate that the performance of IMM fixed-lag smoothing is comparable to that of IMM fixed-interval smoothing. Chen et al. indicate that the fixed-lag smoothing can be implemented in real time with a small delay, compared with fixed-interval smoothing. In 1999, Chen et al (1999) applied the fixed-lag smoothing for a manoeuevring target in cluttered environments. The fixed-lag smoothing algorithm is developed by applying the basic Interacting Multiple Model (IMM) technique to a state-augmented system. Compared with an existing IMM PDA filtering algorithm, the fixed-lag smoothing algorithm obtains much better performance.

This work will use fixed-lag smoothing to improve the multiple-model tracking filter and process online by introducing a small time lag between the instants of estimation and latest measurements. Section 7.2 presents the fixed-lag smoothing in detail. Section 7.3 give the simulation results to demonstrate the improvement for the multiple-model filter with fixed-lag smoothing. Finally, Section 7.4 give the summary of this chapter.

7.2 Fixed-Lag Smoothing

Several authors derived solutions to the fixed-lag smoothing problem (Meditch, 1969; Sage and Melsa, 1971; Priemer and Vaeroux, 1971; Biswas and Mahalanablis, 1972; Meditch, 1973; Moore, 1973; Bierman, 1974; Anderson and Moore,1979; Maybeck, 1982 and the references therein). Schemes for the fixed-lag smoothing solutions are considerably complicated. Here they are not reiterated. Brown and Hwang (1992) recommend a simpler approach, which is by using fixed-interval smoothing to resolve the problem of fixed-lag smoothing. We can always do fixed-lag smoothing by first filtering up to the measurement and then sweeping back a fixed number of steps with the RTS algorithm (Rauch, 1963; Rauch et al., 1965). If the number of backward steps is small, this is a simple and effective way of doing fixed-lag smoothing. If, however, the number of backward steps is large, this method becomes cumbersome. This work only considers a small fixed lag of d intervals, estimating $X(k-d)$ from $\{Z(1), Z(2), \ldots, Z(k)\}$. The procedure for fixed-lag smoothing is shown in Fig. 7.1.

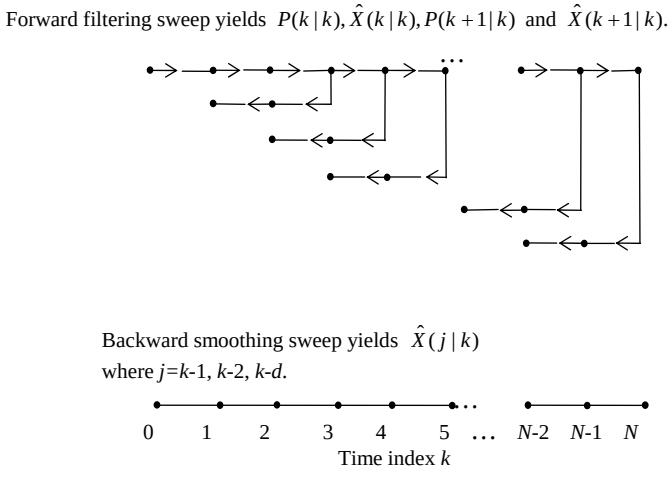


Fig. 7.1 Procedure for fixed-lag smoothing

Fig. 7.1 only illustrates procedure of fixed-lag smoothing a fixed lag of 2. The optimal fixed-lag smoothed estimate of the state at time j is based on measurement data up to time k, where $k>j$; namely, $\hat{X}(j \mid k) = E\{X(j) \mid Z^k\}$. Conceptually, the smoothing problem can be decomposed into two filtering problems: one using initial conditions and the "past" history of data $\{Z(1), Z(2), \ldots, Z(j)\}$ and the other incorporating only "future" measurements $\{Z(j+1), Z(j+2), \ldots, Z(k-1), Z(k)\}$, with the smoothed estimate being the optimal combination of these two filter outputs. The fixed-lag smoothed estimate is the estimation $\hat{X}(j \mid k) + E\{X(j) \mid Z^k\}$ at time j, where $j = k - d$ and $Z^k \equiv \{Z(i), i = 1,2,...,k\}$.

Consider a discrete-time model:

$$X(k+1) = FX(k) + GW(k)$$

$$Z(k) = HX(k) + V(k)$$

Thus, the fixed-lag smoothed estimate can be provided by using the fixed-interval smoothing with equations (7.1)-(7.3).

$$\hat{X}(j \mid k) = \hat{X}(j \mid j) + A(j)[\hat{X}(j+1 \mid k) - \hat{X}(j+1 \mid j)] \tag{7.1}$$

where the smoothing gain $A(j)$ is given by

$$A(j) = P(j \mid j) F^{\mathrm{T}} P^{-1}(j+1 \mid j) \tag{7.2}$$

and $j = k-1, k-2, ..., k-d$, where d is the desired fixed lag.

The error covariance matrix for the smoothed estimates is given by the recursive equation

$$P(j \mid k) = P(j \mid j) + A(j)[P(j+1 \mid k) - P(j+1 \mid j)]A^{\mathrm{T}}(j) \tag{7.3}$$

Maybeck (1982) generated the fixed-lag smoothing from the relation (Kailath, 1975; Meditch, 1967; Meditch, 1969; Meditch, 1973) as followings for an *d*-step time lag.

$$\hat{X}(j+1 \mid j+d+1) = F\hat{X}(j \mid j+d) + C(j+d+1)[Z(j+d+1) -$$

$$H\hat{X}(j+d+1 \mid j+d)] + U(j+1)[\hat{X}(j \mid j+d) - \hat{X}(j \mid j)] \tag{7.4}$$

$$C(j+d+1) = \prod_{k=j+1}^{j+d} A(k) = A^{-1}(j)C(j+d)A(j+d) \tag{7.5}$$

$$U(j+1) = GQ(j)G^{\mathrm{T}}F^{\mathrm{T}}P^{-1}(j \mid j) \tag{7.6}$$

and

$$\begin{aligned} &P(j+1 \mid j+d+1) = P(j+1 \mid j) - \\ &C(j+d+1)K(j+d+1)HP(j+d+1 \mid j+\Delta)C^{\mathrm{T}}(j+d+1) - \\ &A^{-1}(j)[P(j \mid j) - P(j \mid k+d)][A^{-1}(j)]^{\mathrm{T}} \end{aligned} \tag{7.7}$$

For a big fixed lag, we prefer to use equations (7.4)-(7.7) to solve the fixed-lag smoothing problem.

The details about the fixed-lag smoothing can be found in (Brown and Hwang, 1992; Maybeck, 1982).

7.3 Simulation Results

The simulations were done on a Intel 82371AB Pentium II processor and written in MATLAB with version 5.3 like in Chapter 5.

The accuracy is evaluated through the root-mean-square (rms) estimation error for position, speed and acceleration as well.

$$\text{rms position error}_{\text{sample } k} \equiv \tilde{p}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{x}^i_{k|k+d} - x^i_k)^2 + (\hat{y}^i_{k|k+d} - y^i_k)^2)}{N}}$$

$$\text{rms speed error}_{\text{sample } k} \equiv \tilde{v}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{\dot{x}}^i_{k|k+d} - \dot{x}^i_k)^2 + (\hat{\dot{y}}^i_{k|k+d} - \dot{y}^i_k)^2)}{N}}$$

$$\text{rms acceleration error}_{\text{sample } k} \equiv \tilde{a}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{a}^i_{n(k|k+d)} - a^i_{nk})^2 + (\hat{a}^i_{t(k|k+d)} - a^i_{tk})^2)}{N}}$$

$$\text{or rms acceleration error}_{\text{sample } k} \equiv \tilde{a}_k = \sqrt{\frac{\sum_{i=1}^{N}((\hat{a}^i_{x(k|k+d)} - a^i_{xk})^2 + (\hat{a}^i_{y(k|k+d)} - a^i_{yk})^2)}{N}}$$

where $\hat{x}^i_{k|k+d}$ is the smoothed estimate of x at time k in simulation run number i when the forward filter is up to at time $k+d$, and the actual value is x^i_k; a^i_{xk} is the actual acceleration in x-direction; a^i_{nk} is the actual cross-track acceleration; and a^i_{tk} is the actual along-track acceleration.

The computational load is evaluated using the CPU executing time

For each test, a Monte Carlo test of N=100 runs was performed.

The following uses the same three examples as in Chapters 5 and 6 to compare the new fixed-lag smoothing multiple-model filter with the new multiple-model filter in performance and computation load.

The manoeuvre detection and the construction of forward filters used below are the same as in Chapter 6.

1) A target with rectilinear acceleration motion

From Chapter 6, the proposed multiple-model tracking filter is that the unaugmented state model filter is used from the beginning to step 43, switched into the modified Singer's filter from step 44 to 46, changed into the augmented state model filter with rectilinear accelerations is used from step 47 to 74, switched back to the modified Singer's filter from steps 75 to 77 after the manoeuvre end is detected, and then the unaugmented state model is used until the end of tracking.

The procedure of proposed multiple-model filter with fixed-lag smoothing is that the switches of filter type occur d steps earlier, as compared with the proposed multiple-model filter tracking, where d is fixed lag.

The simulation results are shown in Figs 7.2-7.7. Figs 7.2-7.4 show the performance of the proposed filter with fixed-lag smoothing and the proposed filter without fixed-lag smoothing in position, speed and acceleration. Figs 7.5-7.7 show the performance of the proposed filter with fixed-lag smoothing and IMM filter in position, speed and acceleration.

Table 7.1 shows the computation loads of trackers used.

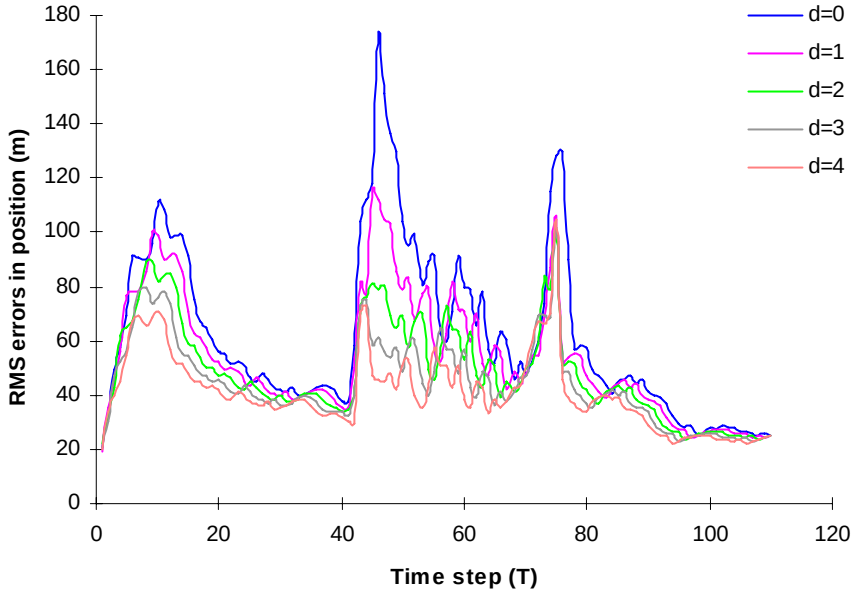


Fig. 7.2 RMS errors of proposed filter with fixed-lag smoothing and without fixed-lag smoothing in position

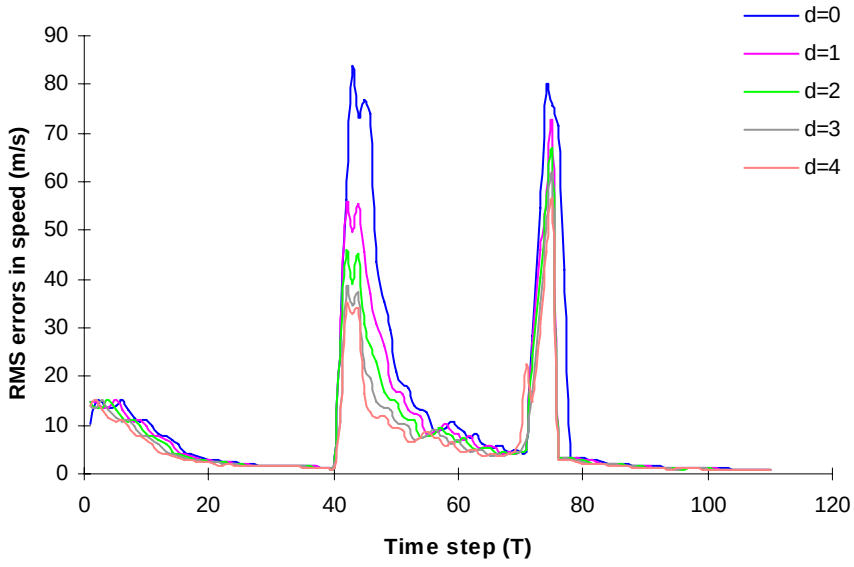


Fig. 7.3 RMS errors of proposed filter with fixed-lag smoothing and without fixed-lag smoothing in speed

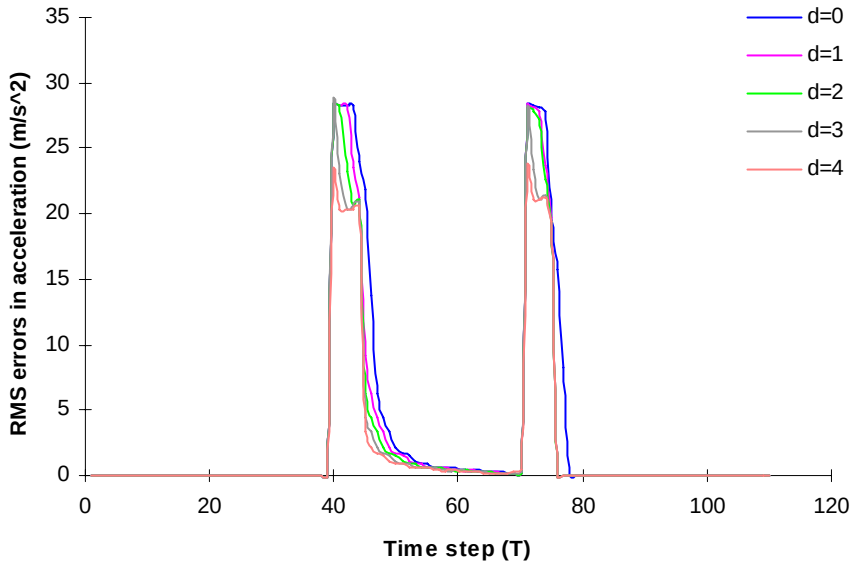


Fig. 7.4 RMS errors of proposed filter with fixed-lag smoothing and without fixed-lag smoothing in acceleration

Figs 7.2-7.4 show that the performance of proposed multiple-model filter is improved by using the fixed-lag smoothing at the cost of a time delay equal to the lag. Even with d=1, the performance improvement is significant. As the fixed-lag d increases, the optimal fixed-lag smoothed estimate of the state is based on more "future" measurements, and the state estimation accuracy improves almost monotonically.

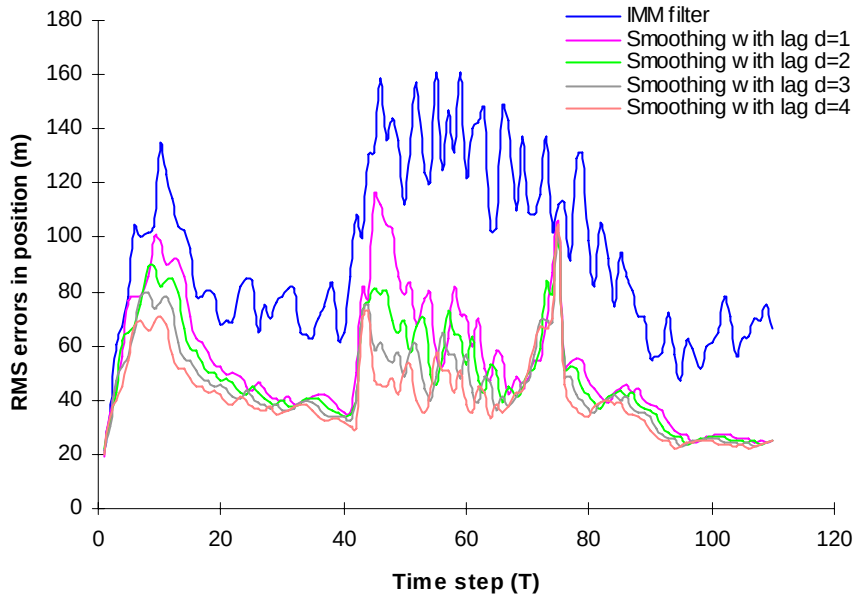


Fig. 7.5 RMS errors of proposed filter with fixed-lag smoothing and IMM filter in position

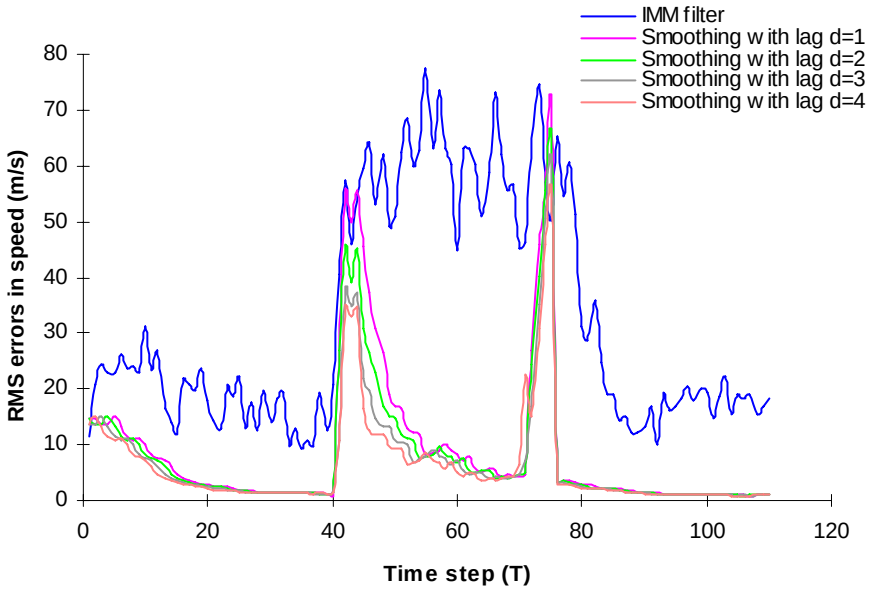


Fig. 7.6 RMS errors of proposed filter with fixed-lag smoothing and IMM filter in speed

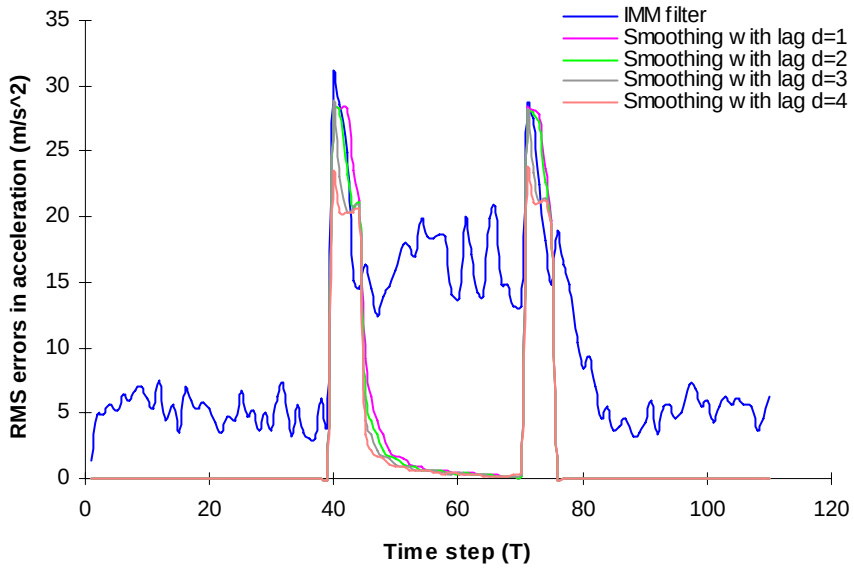


Fig. 7.7 RMS errors of proposed filter with fixed-lag smoothing and IMM filter in acceleration

Figs 7.5-7.7 show that the performance of proposed multiple-model filter with fixed-lag smoothing is much better than that of IMM filter, except for few steps around the start of the manoeuvre and the end of the manoeuvre.

Comparing the computation times of proposed filter without fixed-lag smoothing, proposed filter with fixed-lag smoothing, and IMM filter for averaging 100 runs.

Trackers	Computation time(s)
Proposed filter	1.792
Fixed-lag smoothing with lag d=1	2.144
Fixed-lag smoothing with lag d=2	2.352
Fixed-lag smoothing with lag d=3	2.575
Fixed-lag smoothing with lag d=4	2.755
IMM filter	14.520

Table 7.1 Computational loads of the multiple-model filter, the fixed-lag smoothing multiple-model filters, and IMM filter.

The computational load of fixed-lag smoothing with a lag of 1 interval increases 19.6%; the computational load of fixed-lag smoothing with a lag of 2 intervals increases 31.3%; the computational load of fixed-lag smoothing with a lag of 3 intervals increases 43.7%; and the computational load of fixed-lag smoothing with a lag of 4 intervals increases 53.7%; as compared with that of the multiple-model filter without fixed-lag smoothing.

The computational load of IMM filter is 6.77 times greater than that of fixed-lag smoothing with a lag of 1 interval; 6.17 times greater than that of fixed-lag smoothing with a lag of 2 intervals; 5.64 times greater than that of fixed-lag smoothing with a lag of 3 intervals; and 5.27 times greater than that of fixed-lag smoothing with a lag of 4 intervals.

2) A circular-motion target

From Chapter 6, the scheme of proposed multiple-model filter for tracking is that the unaugmented state model filter is used from the beginning to step 78, switched into the modified Singer's filter from step 79 to 83 before the manoeuvre is certain, changed into the augmented state model filter with cross-track acceleration is used from step 84 to 131, switched back to the modified Singer's filter from steps 132 to 134 after the manoeuvre end is detected, and then the unaugmented state model is used until the end of tracking.

The procedure of proposed multiple-model filter with fixed-lag smoothing is that the changes of filter type occur d steps earlier, as compared with the proposed multiple-model filter tracking, where d is fixed lag.

The simulation results are shown in Figs 7.8-7.13. Figs 7.8-7.10 show the performance of the proposed filter with fixed-lag smoothing and the proposed filter without fixed-lag smoothing in position, speed and acceleration. 7.11-7.13 show the performance of the proposed filter with fixed-lag smoothing and IMM filter in position, speed and acceleration.

Table 7.2 shows the computation loads of trackers used.

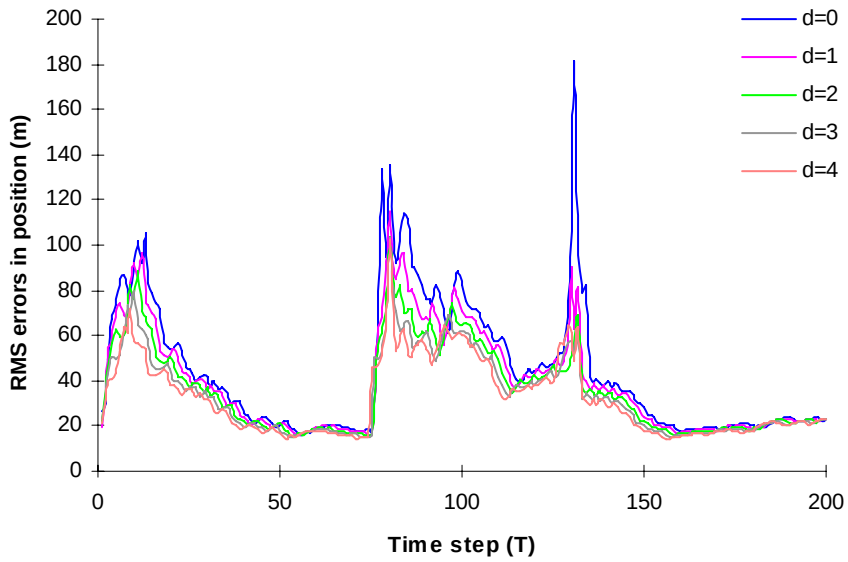


Fig. 7.8 RMS errors of proposed filter with fixed-lag smoothing and without fixed-lag smoothing in position

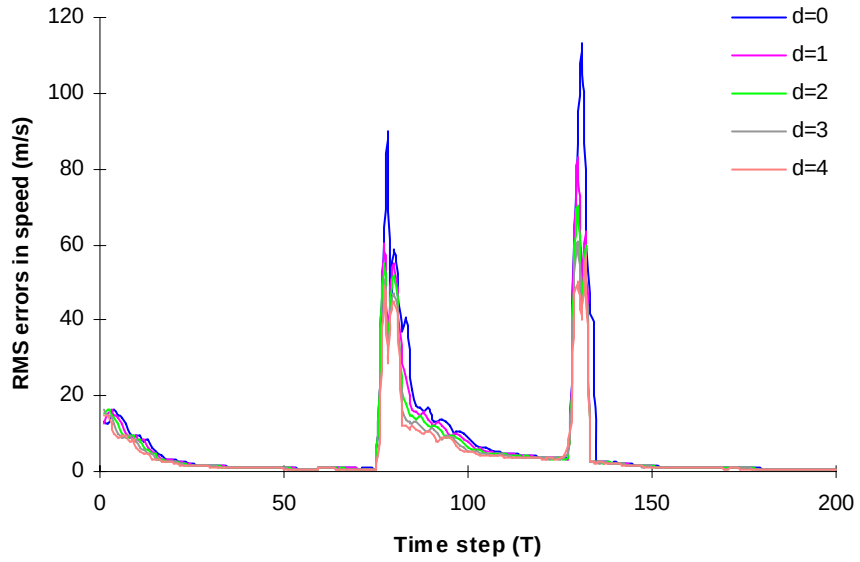


Fig. 7.9 RMS errors of proposed filter with fixed-lag smoothing and without fixed-lag smoothing in speed

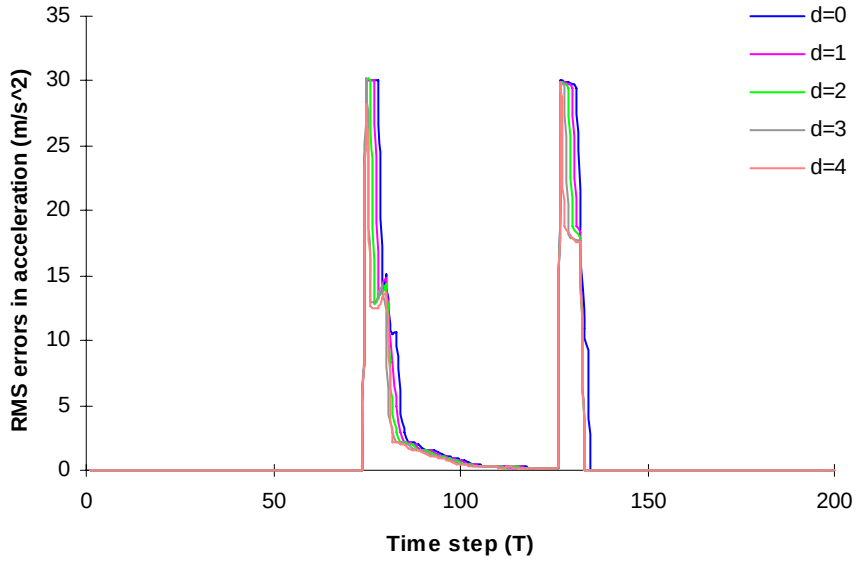


Fig. 7.10 RMS errors of proposed filter with fixed-lag smoothing and without fixed-lag smoothing in acceleration

Figs 7.8-7.10 show that the performance of proposed multiple-model filter with fixed-lag smoothing is much better than that of proposed multiple-model filter without fixed-lag smoothing. Even with d=1, the performance improvement of the fixed-lag smoothing is significant. As the fixed-lag d increases, the state estimation accuracy improves almost monotonically.

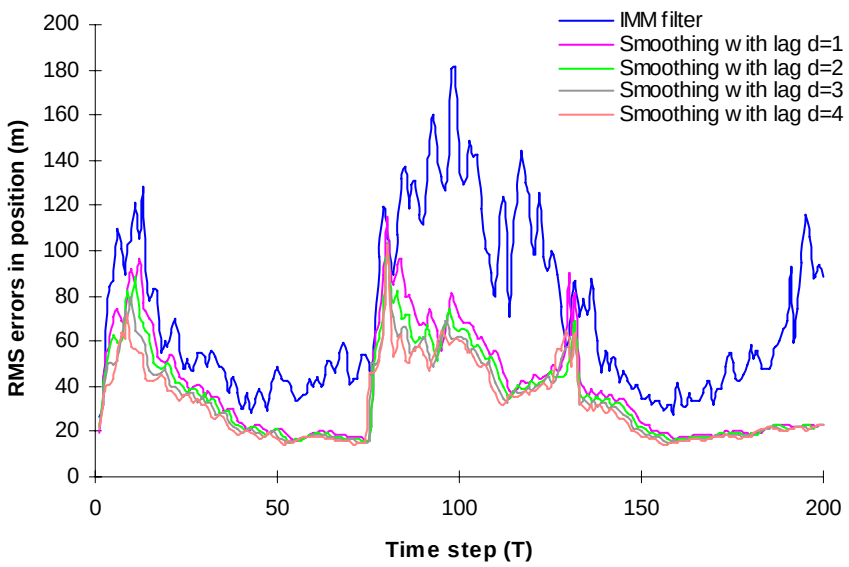


Fig. 7.11 RMS errors of proposed filter with fixed-lag smoothing and IMM filter in position

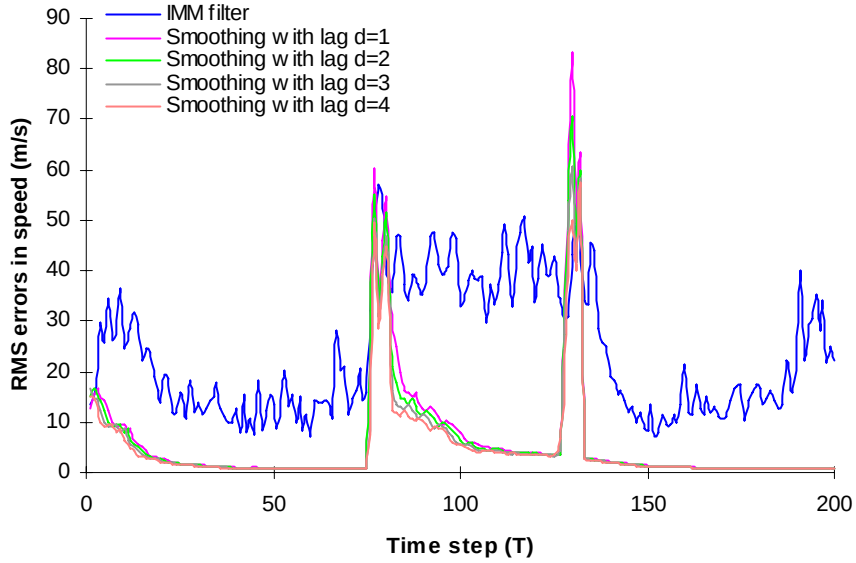


Fig. 7.12 RMS errors of proposed filter with fixed-lag smoothing and IMM filter in speed

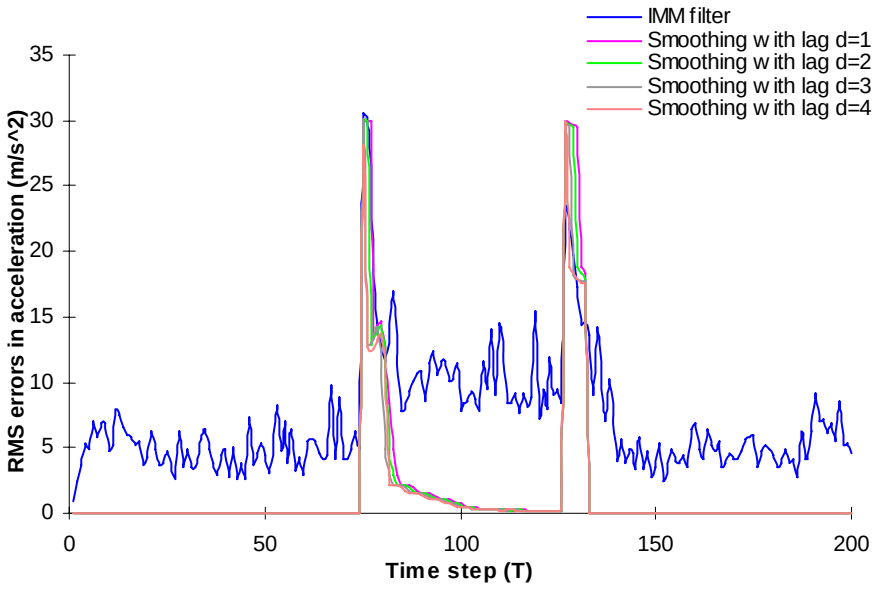


Fig. 7.13 RMS errors of proposed filter with fixed-lag smoothing and IMM filter in acceleration

Figs 7.11-7.13 show that the improvement of the performance of proposed multiple-model filter with fixed-lag smoothing is significant, as compared with IMM filter, except for few steps around the start of the manoeuvre and the end of the manoeuvre.

Comparing the computation times of proposed filter without fixed-lag smoothing, proposed filter with fixed-lag smoothing, and IMM filter for averaging 100 runs.

Trackers	Computation time(s)
Proposed filter	4.76
Fixed-lag smoothing with lag d=1	5.54
Fixed-lag smoothing with lag d=2	5.829
Fixed-lag smoothing with lag d=3	6.239
Fixed-lag smoothing with lag d=4	6.55
IMM filter	29.080

Table 7.2 Computational loads of the multiple-model filter, the fixed-lag smoothing multiple-model filters, and IMM filter.

The computational load of fixed-lag smoothing with a lag of 1 interval increases 16.4%; the computational load of fixed-lag smoothing with a lag of 2 intervals increases 22.5%; the computational load of fixed-lag smoothing with a lag of 3 intervals increases 31.1%; and the computational load of fixed-lag smoothing with a lag of 4 intervals increases 37.6%; as compared with that of the multiple-model filter without fixed-lag smoothing.

The computational load of IMM filter is 5.25 times greater than that of fixed-lag smoothing with a lag of 1 interval; 4.99 times greater than that of fixed-lag smoothing with a lag of 2 intervals; 4.66 times greater than that of fixed-lag smoothing with a lag of 3 intervals; and 4.44 times greater than that of fixed-lag smoothing with a lag of 4 intervals.

3) An agile target

From Section 6.3.3 in Chapter 6, the switching of filters in the proposed multiple-model filter for tracking is:

From steps the beginning to step 155, from steps 221 to step 277, from steps 342 to 397, and from steps 468 to 510: the unaugmented state model filter is used.

From steps 156 to 171, from steps 218 to 220, from steps 278 to 296, from steps 339 to 341, from steps 398 to 408, and from steps 465 to 467: the modified Singer's filter is used.

From steps 172 to 217, and from steps 297 to 338: the augmented state model filter with cross-track acceleration is used.

From steps 409 to 465: the augmented state model filter with along- and cross-track accelerations is used.

The procedure of proposed multiple-model filter with fixed-lag smoothing is that all changes in filter type occur d steps earlier, as compared with the proposed multiple-model filter tracking, where d is fixed lag.

The simulation results are shown in Figs 7.14-7.19. Figs 7.14-7.16 show the performance of the proposed filter with fixed-lag smoothing and the proposed filter without fixed-lag smoothing in position, speed and acceleration. Figs 7.17-7.19 show the performance of the proposed filter with fixed-lag smoothing and IMM filter in position, speed and acceleration.

Table 7.3 shows the computation loads of trackers used.

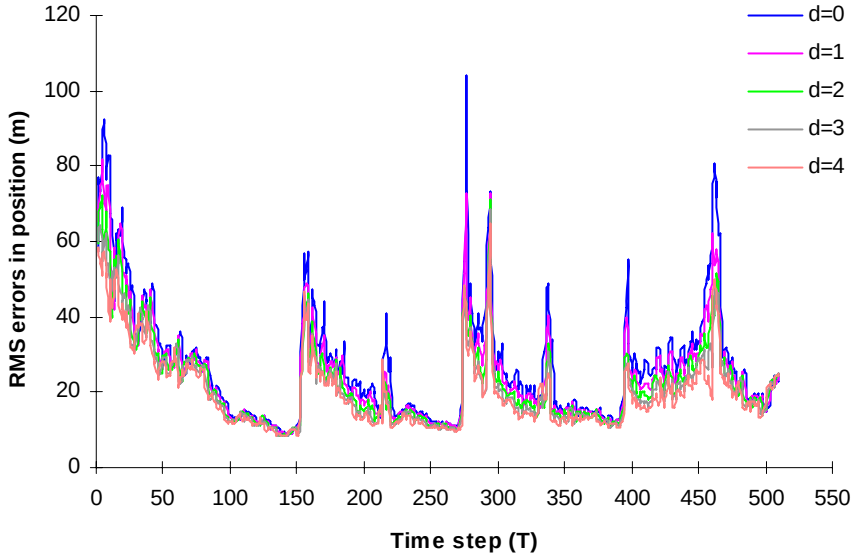


Fig. 7.14 RMS errors of proposed filter with fixed-lag smoothing and without fixed-lag smoothing in position

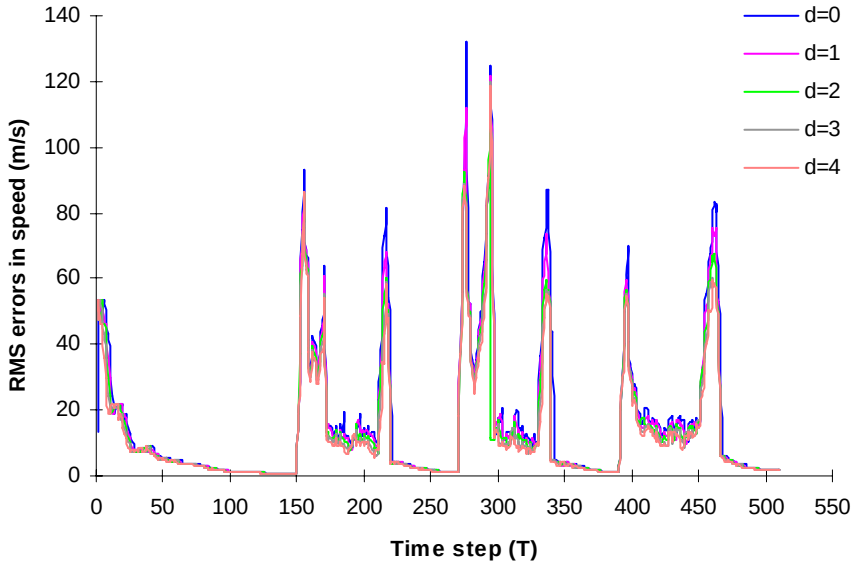


Fig. 7.15 RMS errors of proposed filter with fixed-lag smoothing and without fixed-lag smoothing in speed

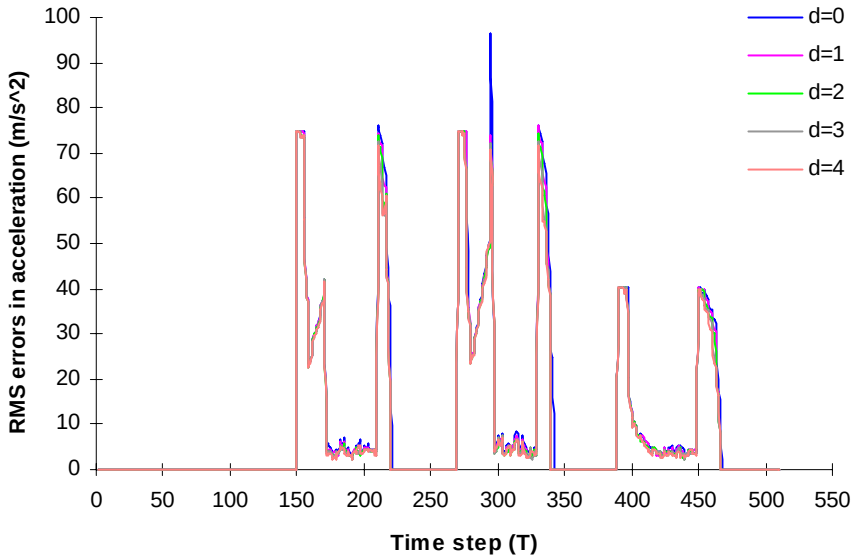


Fig. 7.16 RMS errors of proposed filter with fixed-lag smoothing and without fixed-lag smoothing in acceleration

Figs 7.14-7.16 show that the performance of proposed multiple-model filter with fixed-lag smoothing is significantly better than that of proposed multiple-model filter without fixed-lag smoothing, and give the similar results to the cases of a target with rectilinear acceleration motion and the other target with a circular motion mentioned before.

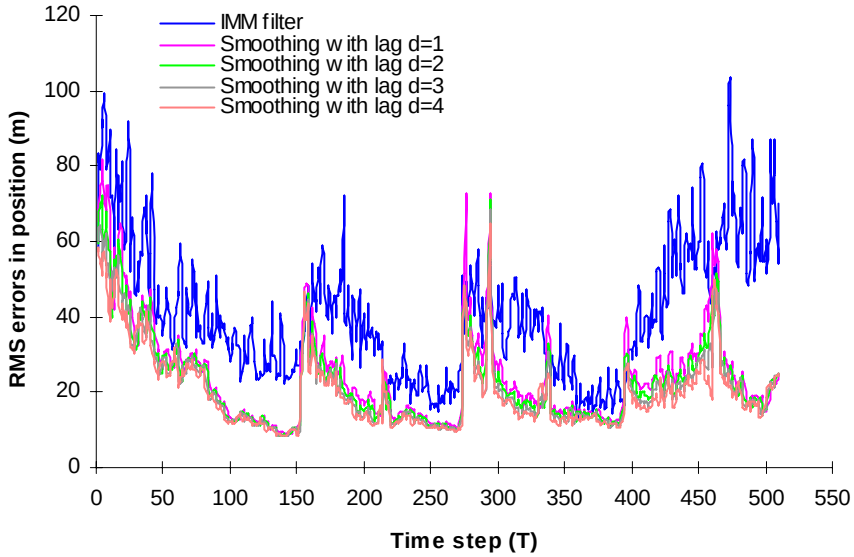


Fig. 7.17 RMS errors of proposed filter with fixed-lag smoothing and IMM filter in position

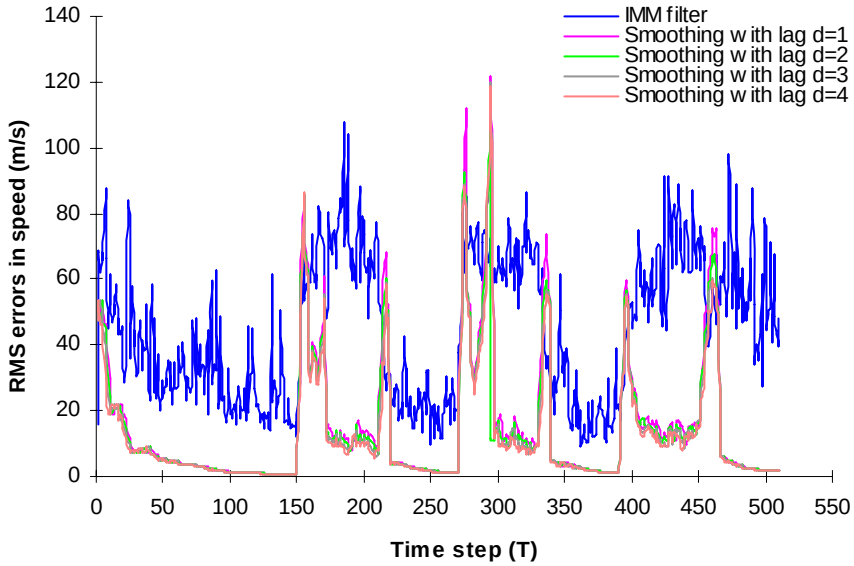


Fig. 7.18 RMS errors of proposed filter with fixed-lag smoothing and IMM filter in speed

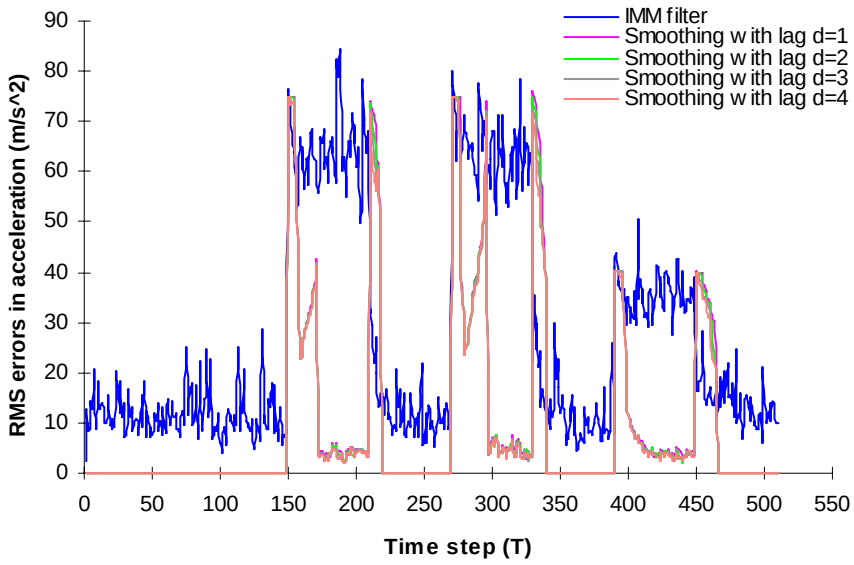


Fig. 7.19 RMS errors of proposed filter with fixed-lag smoothing and IMM filter in acceleration

Figs 7.17-7.19 show that the performance of proposed multiple-model filter with fixed-lag smoothing is much better than that of IMM filter except for few steps.

Comparing the computation times of proposed filter without fixed-lag smoothing, proposed filter with fixed-lag smoothing, and IMM filter for averaging 100 runs.

Trackers	Computation time(s)
Proposed filter	19.583
Fixed-lag smoothing with lag d=1	22.06
Fixed-lag smoothing with lag d=2	23.021
Fixed-lag smoothing with lag d=3	23.974
Fixed-lag smoothing with lag d=4	24.874
IMM filter	74.040

Table 7.3 Computational loads of the multiple-model filter, the fixed-lag smoothing multiple-model filters, and IMM filter.

The computational load of fixed-lag smoothing with a lag of 1 interval increases 12.6%; the computational load of fixed-lag smoothing with a lag of 2 intervals increases 17.6%; the computational load of fixed-lag smoothing with a lag of 3 intervals increases 22.4%; and the computational load of fixed-lag smoothing with a lag of 4 intervals increases 27.0%; as compared with that of the multiple-model filter without fixed-lag smoothing.

The computational load of IMM filter is 3.36 times greater than that of fixed-lag smoothing with a lag of 1 interval; 3.22 times greater than that of fixed-lag smoothing with a lag of 2 intervals; 3.09 times greater than that of fixed-lag smoothing with a lag of 3 intervals; and 2.98 times greater than that of fixed-lag smoothing with a lag of 4 intervals.

7.4 Summary

This chapter gave the details of the new multiple-model filter with fixed-lag smoothing, and also compared the performance of the fixed-lag smoothing multiple-model filter with the multiple-model filter proposed in Chapter 6 and IMM filter via three typical examples.

The simulation results show that the proposed multiple-model filter can be improved significantly by using the fixed-lag smoothing even with d=1, and also show that as the fixed-lag d increase, the state estimation accuracy improves almost monotonically.

The fixed-lag smoothing multiple-model filter slightly increases the computational load , as compared with the multiple-model filter. The computational load of the fixed-lag smoothing increases linearly with lag.

The performance of the multiple-model filter with the fixed-lag smoothing is much better than that of IMM filter at most times but with significant reduction in computational load.

All in all, the fixed-lag smoothing multiple-model filter provides much better performance, and can be implemented in real time at the costs of a small delay and slightly increase in computational load, compared to the multiple-model filter on its own. It provides much better performance than IMM filter but with significantly less computational load. It is useful for tracking reconstruction (e.g. to help in identifying target type).

CHAPTER 8

CONCLUSIONS AND FURTHER WORK

This work has developed a new multiple-model filter for tracking manoeuvring targets to achieve the aims of providing the better performance and economical computational load. The approach of new multiple-model filter is to assume one or more manoeuvre models when the manoeuvre is detected. The switching decision from one model to another is made by the manoeuvre detection and the estimated accelerations. The new tracking method has been shown to be superior to the common multiple model tracking, IMM filter, constant-velocity filter, Singer's filter, and variable-dimension filter. Further improvement for the multiple-model tracking is provided by using the fixed-lag smoothing technique.

The following sections give more detail about this work, draw conclusions, and propose further work.

8.1 Target Motion Models

The most general models are derived from mechanics and involve the forcing acting on the targets, such as thrust, lift and drag. However, these models employ unmeasurable and incompletely observable forcing, and are too complicated to implement in a practical system. The usual approach is to use kinematics to construct the motion models, based solely on position, velocity and acceleration.

Since the measurements usually come in the form of a range and bearing, the most obvious choice of modelling system is polar co-ordinates. However, the modelling system of polar co-ordinates means that even the simplest of target motions, straight-

line travel at constant speed, results in a non-linear target model. The most common choice of co-ordinates is Cartesian co-ordinates with the origin at the tracker's initial position. The non-linear measurements are overcome by the technique of Polar-to-Cartesian measurement conversion (Bar-Shalom and Li, 1995)..

Usually, the target trajectory model candidates are constant-velocity motion and rectilinear acceleration motion. For constant-velocity motion, the target trajectories are modelled as having straight-line constant velocity with slight changes in speed. For simplicity, the rectilinear-acceleration model only considers constant acceleration and constant rate of acceleration variation. The constant-acceleration motion is modelled as straight-line constant acceleration with slight changes in acceleration. In practice, target manoeuvre models are along-track acceleration (straight-line acceleration motion), cross-track acceleration (circular motion), and curvilinear acceleration models. For these practical manoeuvre motions, Best and Norton suggest a more general model with both cross- and along-track accelerations (Best and Norton, 1997). However, Best and Norton's model is only suitable for the target with known constant cross- and along-track accelerations. This work has modified Best and Norton's model by augmenting the state model to make the model able to track the manoeuvring target with unknown constant cross- and along-track accelerations.

The main problem with the rectilinear acceleration model and curvilinear acceleration model above is that accelerations are assumed constant in the x and y directions, and cross-track and along-track directions respectively over each sampling interval. Therefore, if these models are used in target tracking, significant modelling errors arise when the target accelerations suddenly change in the x, y, cross-track or along-track

directions. This thesis modified Singer's filter to handle these situations and broaden the application of Singer's filter.

The noise processes affecting the target motion must also be considered in tracker design. They are usually modelled as white noise with a zero-mean Gaussian distribution.

8.2 Manoeuvre Detection

The classical manoeuvre detection by testing the normalised squared innovations is simple but suffers much delay or false alarm. Bar-Shalom and Birmiwal (1982) suggest that the end of the manoeuvre be detected by testing a statistic for significance of the acceleration in the manoeuvring model. This detection technique is faster than the classical method. However, the effectiveness and reliability depend on a good reconstruction of process-noise covariance of the manoeuvre filter. If the process-noise covariance is too large, it causes false alarms, but if it too small, the end-of-manoeuvre detection is delayed. The process-noise covariance has to be reconstructed by the state estimates at the step at which the target is supposed to manoeuvre. However, it is difficult to reconstruct a good process-noise covariance within the sliding window when changing to the manoeuvre model, and so difficult to detect the end of a manoeuvre reliably by this method. Chan and Couture's manoeuvre detection (1993) by comparing the projections against measurements suffers much delay or false alarm as well. McAulay and Denlinger's manoeuvre detection (1973) by using matched filtering through noisy innovations; and Zhang and Li's (1997, 1998), and Isaksson and Gustafsson's manoeuvre detection (1995) by observing the probability of each model, these methods require the knowledge of the manoeuvre before detecting it, which is difficult to supply in real situations.

In order to overcome the problems in (Bar-Shalom and Birmiwal, 1982), the augmented model is used throughout the process with the same process noise covariance as the normal model, and fixed-lag smoothing is used to provide better estimates of the accelerations and their error covariance. The test based on normalised squared smoothed acceleration is reliable and quick for detecting the beginning and the end of manoeuvres.

Chen and Norton's rapid parameter changes detection (1987), based on Hägglund's method (1983), is used effectively in adaptive parameter estimation and adaptive detecting rapid parameter changes. The feature of manoeuvre is that the statistic of normalised squared innovations increases in size abruptly, so detecting the changes of the statistic of normalised squared-innovation gives effective manoeuvre detection and broadens the application of Chen and Norton's detection.

Weston and Norton's manoeuvre detection is to detect changes in state which can be represented as due to impulses in the input via full measurements with offline processing. Its advantages are simplicity and effectiveness. But it cannot be applied to tracking with online processing. Weston and Norton's manoeuvre detection has been modified with fixed-lag smoothing instead of the fixed-interval smoothing in this thesis. The modified Weston and Norton's manoeuvre detection is able to work in online processing, and also obtain the more accurate manoeuvre step and quicker detection.

8.3 Trackers

The performance of the VD filter is superior to that of the input estimation (IE) filter if the manoeuvre is low-level, and its computational requirements less. However, the VD algorithm presumes that the target starts to manoeuvre at the starting point of a sliding window running back from when the manoeuvre is detected, but the actual and assumed manoeuvres may differ in timing, leading to large tracking errors. The VD filter has to reconstruct the process noise covariance and the state estimates within the sliding window when changing to the manoeuvre model. Furthermore, the filter uses only measurements at the start of the sliding window to initialise the augmented filter. This also increases the tracking error.

The common multiple-model filter is to assume two or more levels of process noise. Potentially, the estimate error is increased if the individual filters' probabilities are poorly estimated.

The interacting multiple model (IMM) algorithm is an effective technique for tracking a manoeuvring target. However, the IMM filter incorporates a large number of filters to cover the possible manoeuvre motions, resulting in heavy computation.

Another adaptive Kalman filter suggests increasing the parameter $Q(k)$ (Thorp, 1973; Bekir, 1983). Increasing this parameter effectively increases the bandwidth of the Kalman filter thus making the filter more responsive to future target dynamics and to noise! Potentially, loss of track can result.

Singer's augmented filter shows good performance for the target with a low-level manoeuuvre, but its performance rapidly degrades in the case of a high-level manoeuvring target and is worse than that with a Kalman filter that assumes no manoeuvres, during constant-velocity, straight-line motion. Fortunately, the manoeuvre

level is estimated by input estimation technique, and then regards target accelerations as a perturbation around the estimated acceleration level to construct Singer's filter. The final Singer's filter is called modified Singer's filter in this thesis. The modified Singer's filter would track a target executing a manoeuvre of any level. This thesis uses the modified Singer's filter to provide the state estimation in the cases of uncertain manoeuvre.

The input estimation shows poor performance in the case of low-level manoeuvring target because it tends to over-compensate for the manoeuvre. However, the input estimation technique has good performance for the target in rectilinear acceleration motion with a high-level manoeuvre if the manoeuvre is detected quickly. Unfortunately, if the target is in curvilinear acceleration manoeuvre, the rectilinear accelerations would not be constant any more; consequently, the input estimation could not handle this situation. For the curvilinear acceleration motion, this work suggests that the tracking be provided by the modified Best and Norton model.

Therefore, this work developed a new multiple-model filter for tracking manoeuvring targets. The new multiple model approach employs three manoeuvre models. One is straight-line acceleration manoeuvre, one is curvilinear acceleration manoeuvre, and the other is circular motion. The switching decision from one model to another is made by the manoeuvre detection and the estimated accelerations. For any ambiguous manoeuvre, the tracking is provided by modified Singer's filter. The new tracking method gives better performance than the common multiple-model tracking, IMM filter, and any other filters above.

The simulations show that a filter with a model matched to the target motion will give a good performance, and that the unmatched filter will lose track. The simulations therefore show that quick and accurate manoeuvre detection is as important as the tracking needs matched model.

Further improvement for multiple-model tracking is provided by using the fixed-lag smoothing technique. The fixed-lag smoothing multiple-model filter provides much better performance (even with fixed lag d=1) and can be implemented in a real time at the costs of a small delay and slight increase in computational load, compared to the multiple-model filter. As the fixed-lag d increase, the state estimation accuracy improves almost monotonically. The computational load of the fixed-lag smoothing increases linearly with lag. The performance of the multiple-model filter with the fixed-lag smoothing is much better than that of IMM filter at most times but with significant reduction in computational load.

8.4 Further Work

This thesis has developed manoeuvre detection, manoeuvring target modelling, and generated an effective and economical multiple-model filter. However, many aspects of the problem of tracking, such as clutter, multiple targets and multiple sensors, have to be ignored because this thesis work is only for a three-year project. Thus, it is necessary to continue this research and broaden the applications of the new multiple-model filter.

This work only concerned the tracking of targets in clear environments. However, in the presence of clutter, the computational load usually is increased, tracking is much more difficult, thus it is necessary to develop the multiple-model filter and the implementation of economical filters, and reliable manoeuvre detection. This work only

considered that measurement is associated with the same target, but it is necessary to solve tracking problems for measurements whose origin is uncertain due to: clutter due to spurious reflectors or radiators near the target of interest, interfering targets, and decoys or other countermeasures. This work has not considered multiple targets, so it is necessary to check if it is suitable for the multiple-target and multiple-sensor tracking. Furthermore, the uncertainty in the tracking model has been only considered to be Gaussian distribution and measurement outliers have been ignored. In practice, these outliers (e.g. glint, clutter, countermeasures) must be accounted for.

REFERENCES

Aidala, V. I., and Hammel, S. E., 1983, Utilization of modified polar coordinates for bearing-only tracking, IEEE Transactions on Automatic Control, 28, pp283-297.

Alouani, A. T., and Blair, W. D., 1991, Use of a kinematic constraint in tracking constant speed, maneuvering targets, Proceedings of the 30th Conferences on Decision and Control, Brighton, England, pp2055-2058.

Alouani, A. T., and Xia, P., 1991, A two-stage Kalman estimator for state estimation in the presence of random bias and for tracking maneuvering targets, Proceedings of the 30th Conference on Decision and Control, Brighton, England, pp2059-2062.

Alouani, A. T., Rice, T. R., and Blair, W. D., 1992, A two-stage filter for state estimation in the presence of dynamical stochastic bias, Proceedings of 1992 American Control Conference, Chicago, IL, pp1784-1788.

Alspach, D. L., 1974, A parallel filtering algorithm for linear systems with unknown time varying noise statistics, IEEE Trans. Automat. Contr. 17, pp552-555.

Anderson, B. D., And Moore, J. B., 1979, Optimal filtering, Prentice-Hall: N. J.

Anderson, K. L., and Iltis, R. A., 1996, A distributed bearings-only tracking algorithm using reduced sufficient statistics, IEEE Transactions on Aerospace and Electronic Systems, 32, pp339-349.

Aoki, M., 1967, Optimization of Stochastic Systems — Topics in Discrete-time Systems, Academic Press, New York.

Arcasoy, C. C., 1971, Return-difference-matrix properties for optimal stationary discrete Kalman filter, Proc. IEE, 118, pp1831-1834.

Atherton, D. P., and Lin, H. J., 1994, Parallel implementation of IMM tracking algorithm using transputers, IEE Proceedings on Radar, Sonar Navig., 141, pp325-332.

Averbuch, A., Itzikowitz, S., and Kapon, T., 1991, Parallel implementation of multiple model tracking algorithms, IEEE Tranactions on Parallel and Distributed Systems, 2, pp242-252.

Averbuch, A., Itzikowitz, S., and Kapon, T., 1991, Radar target tracking-viterbi versus IMM, IEEE Transactions on Aerospace and Electronic Systems, 27, pp550-563.

Bar-Shalom, Y., and Tse, E., 1975, Tracking in a cluttered environment with probabilistic data association, Automatic, 11, pp451-460.

Bar-Shalom, Y., and Birmiwl, K., 1982, Variable dimension filter for maneuvering target tracking, IEEE Transactions on Aerospace and Electronic Systems, 18, pp621-629.

Bar-Shalom, Y., and Birmiwal, K., 1983, Consistency and robustness of PDAF for target tracking in cluttered environments, Automatica, 19, pp431-437.

Bar-Shalom, Y., and Fortmann, T. E., 1988, Tracking and Data Association, Academic Press Limited, London.

Bar-Shalom, Y., 1987, MULTIDAT: Multiple-model data association tracker, Interactive Software Package.

Bar-Shalom, Y., Chang, K. C., and Blom, H.A.P., 1989, Tracking a maneuvering target using input estimation versus the interacting multiple model algorithm, IEEE Transactions on Aerospace and Electronic Systems, 25, pp296-300.

Bar-Shalom, Y., 1989, Recursive tracking algorithms: From the Kalman filter to intelligent trackers for cluttered environment, In Proceedings of the 1989 IEEE International Conference on Control and Applications.

Bar-Shalom, Y., Chang, K. D., and Blom, H. A. P., 1990, Automatic track formation in clutter with a recursive algorithm, In Bra-Shalom(Ed.), Multitarget-Multisensor Tracking: Advanced Applications, Vol. II, Norwood, MA: Artech House.

Bar-Shalom, Y., 1990, Multitarget-Multisensor Tracking: Advanced Applications, Norwood, MA: Artech House.

Bar-Shalom, Y., 1992, Multitarget-Multisensor Tracking: Applications and Advances, Vol. II, Norwood, MA: Artech House.

Bar-Shalom, Y., and Li, X., 1993, Estimation and tracking: Principles, techniques and software, Artech House, Norwood, MA.

Bar-Shalom, Y., and Li, X., 1995, Multitarget-Multisensor Tracking: Principles and techniques, Stoors, CT: YBS Publishing.

Bar-Shalom, Y., Kumar, A., Blair, W. D., Groves, G. W., 1994, Tracking low elevation targets in the presence of multipath propagation, IEEE Transactions on Aerospace and Electronic Systems, 30, pp973-979.

Beadle, E. R., and Djuric, P. M., 1997, A fast weighted Bayesian bootstrap filter for nonlinear model state estimation, Edward R. Beadle, Peter M. Djuric, IEEE Transactions on Aerospace and Electronic Systems, 33, pp338-343.

Bekir, E., 1983, Adaptive Kalman Filtering for Tracking Maneuvering Targets, Journal of Guidance, Control and Dynamics, 6, pp 414-416.

Belforte, G., and Tay, T. T., 1990, Two new algorithms for sequential parameter estimation with unknown but bounded noise, Proceedings of the 29th Conference on Decision and Control, Honolulu, Hawall, pp3546-3551.

Becker, R. A., 1954, Introduction to theoretical mechanics, New York: McGraw-Hill, 1954, 259-260.

Berg, R. F., 1983, Estimation and prediction for maneuvering target trajectories, IEEE Transactions on Automatic and Electronic Systems, 6, pp473-483.

Bertsekas D. P., and Rhodes, I. B., 1971, Recursive state estimation for a set-membership description of uncertainty, IEEE Transactions on Automatic Control, 16, pp117-128.

Best, R. A., and Norton, J. P., 1997, A new model and efficient tracking for a target with curvilinear motion, IEEE Transactions on Aerospace and Electronic Systems, 33, pp1030-1037.

Beuzit, M., 1989, Analytical steady-state solution for a three-state Kalman filter, IEEE Transactions on Aerospace and Electronic Systems, 25, pp828,835.

Bierman, G. J., 1974, An efficient fixed-lag smoothing algorithm for discrete linear systems, Automatic, 10, pp559-563.

Birmiwal, K., and Bar-Shalom, Y., 1984, On tracking a maneuvering target in clutter, IEEE Transactions on Aerospace and Electronic Systems, 20, pp635-644.

Biswas, K., and Mahalanablis, A., 1972, Optimal fixed-lag smoothing for time delayed system with colored noise, IEEE Transactions on Automatic Control, 19, pp387-388.

Blair, W. D., 1992, Two-stage alpha-beta-gamma-lambda estimator for tracking maneuvering targets, Proceedings of 1992 American Control Conference, Chicago, pp842-846.

Blair, W. D., and Watson, G. A., 1992, Two-stage alpha-beta-gamma-lambda estimator for tracking constant speed, maneuvering targets, Proceedings of 1992 American Control Conference, Chicago, pp847-851.

Blair, W. D., and Watson, G. A., 1992, Interacting multiple bias model algorithm with application to tracking maneuvering targets, Proceedings of the 31st Conferences on Decision and Control, Tucson, Artzons, pp3790-3795.

Blair, W. D., Watson, G. A., and Hoffman, S. A., 1994, Benchmark problem for bean pointing control of phased array radar against maneuvering targets, Proceedings American Control Conference.

Blair, W. D., Watson, G. A., Kirunarajan, T., and Bar-Shalom, Y., 1998, Benchmark for radar allocation and tracking in ECM, IEEE Transactions on Aerospace and Electronic Systems, 34, pp1097-1114.

Blom, H. A. P., 1984, An efficient filter for abruptly changing systems, In Proceedings of the 23rd IEEE Conference on Decision and Control, Las Vegas, pp656-658.

Blom, H. A. P., 1985, An efficient decision-making-free filter for processes with abrupt changes, IFAC Conference on Identification and System Parameter Estimation, pp631-636.

Blom, H. A. P., and Bar-Shalom, Y., 1988, The interacting multiple model algorithm for systems with Markovian switching coefficients, IEEE Transactions on Automatic Control, 33, pp780-783.

Blom, H. A. P., and Hogendoom, R. A., and Van Dorn, B. A., 1992, Design of a multisensor tracking system for advanced air traffic control, In Bar-Shalom, Y.(Ed), Multitarget-Multisensor Tracking: Applications and Advances, Vol. II, Norwood, MA: Artech House.

Bogler, P. L., 1987, Tracking a maneuvering target using input estimation, IEEE Transactions on Aerospace and Electronic Systems, 23, pp298-310.

Breiman, L., 1969, Probability and stochastic processes: with a view toward applications, Houghton Mifflin Company, Boston.

Brown, R. G. and Hwang, Patrick Y. C., 1992, Introduction to Random Signals and Applied Kalman Filtering, John Wiley & Sons, Inc.

Carpenter, J., Clifford, P., and Fearnhead, P., 1999, Improved particle filter for nonlinear problems, IEE Proceedings on Radar, Sonar and Navigation, 146, pp2-7.

Chan, Y. T., Hu, A. G. C., and Plant, J. B., 1979, A Kalman filter based tracking scheme with input estimation, IEEE Transactions on Aerospace and Electronic Systems, 15, pp237-244.

Chan, Y. T., Plant, J. B., and Bottomley, J. B. T., 1982, A Kalman tracker with a simple input estimator, IEEE Transactions on Aerospace and Electronic Systems, 18, pp235-241.

Chan, Y. T., and Couture, F., 1993, Manoeuvre detection and track correction by input estimation, IEE Proceedings-F, 140, pp21-28.

Chang, C. B., Whiting, R. H., and Athans, M., 1977, On the state and parameter estimation for maneuvering reentry vehicles, IEEE Transactions on Automatic Control, 22, pp99-105.

Chang, K. C., and Bar-Shalom, Y., 1984, Joint probabilistic data association for multitarget tracking with possibly unresolved measurements and maneuvers, IEEE Transactions on Automatic Control, 29, pp585-594.

Chansarkar, M. M., and Desai, U. B., 1997, A robust recursive least square algorithm, IEEE Transactions on Signal Processing, 45, pp1726-1735.

Chen, B., and Tugnait, J. K., 2000, Interacting multiple model fixed-lag smoothing algorithm for Markovian switching systems, IEEE Transactions on Aerospace and Electronic Systems, 36, pp243-249.

Chen, G., Wang, J., and Shieh, L. S. 1997, Interval Kalman filtering, IEEE Transactions on Aerospace and Electronic Systems, 33, pp250-258.

Chen, J., and Patton, R. J., 1996, Optimal filter and robust fault diagnosis of stochastic systems with unknown disturbances, IEE Proceedings on control theory and applications, 143, pp31-36.

Chen, M. J., and Norton, J. P. 1987, Estimation technique for tracking rapid parameter changes, International Journal of Control, 45, pp1387-1398.

Chirarttananon, S., and and Anderson, B. D., 1971, The fixed-lag smoother as a stable finite-dimensional linear system, Automatic, pp657-669.

Cloutier, J. R., 1993, Enhanced variable dimension filter for maneuvering target tracking, IEEE Transactions on Aerospace and Electronic Systems, 29, pp786-796.

Cortina, E., and Otero, D., 1991, Maneuvering target tracking using extended Kalman filter, IEEE Transactions on Aerospace and Electronic Systems, 27, pp155-158.

Costa, O. L. V., 1994, Linear minimum mean square error estimation for discrete-time Markovian jump linear systems, IEEE Transactions on Automatic Control, 39, pp1685-1689.

Daeipour, E., Bar-Shalom, Y., and Li, X. R., 1994, Adaptive beam pointing control of a phased array radar using an IMM estimator, Proceedings American Control Conference

Daeipour, E., and Bar-Shalom, Y., 1995, An interacting multiple model approach for target tracking with glint noise, IEEE Transactions on Aerospace and Electronic Systems, 31, pp706-715.

Dey, S., and Moore, J. B., 1997, Risk-sensitive filtering and smoothing via reference probability methods, IEEE Transactions on Automatic Control, 42, pp1587-1591.

Downing, A. J., and Norton, J. P., 1996, Computation of parameter bounds from bounded histogram error specifications, IFAC, 13th Triennial World Congress, San Francisco, USA, pp55-60.

Dufour, F., and Mariton, M., 1992, Passive sensor data fusion and maneuvering target tracking. In Bar-Shalom, Y. (Ed.), Multitarget-Mutisensor Tracking: Applications and Advances, Vol. II, Norwood, MA: Artech House.

Durovic, Z. M., and Kovacevic, B. D., 1999, Robust estimation with unknown noise statistics, IEEE Transactions on Automatic Control, 44, pp1292-1296.

Efe, M., and Atherton, D. P., 1997, A tracking algorithm for both highly maneuvering and nonmaneuvering targets, Proceedings of the 36th Conference on Decision & Control, pp3150-3155.

Farooq, M., and Bruder, S., 1989, Comments on "tracking a maneuvering target using input estimation", IEEE Transactions on Aerospace and Electronic Systems, 25, pp300-302.

Friedland, B., 1969, Treatment of bias in recursive filtering, IEEE Transactions on Automatic Control, 14, pp359-367.

Friedmann, J., Messer, H., and Cardoso, J. F., 2000, Robust parameter estimation of a deterministic signal in impulsive noise, IEEE Transactions on Signal Processing, 48, pp935-942.

Gauvrit, M., 1984, Bayesian adaptive filter for tracking with measurements of uncertain origin, Automatica, 20, pp217-224.

Gelb, A., 1974, Applied Optimal Estimation, M. I. T. Press, Cambridge.

Gelfand, S. B., Fortmann, T. E., and Bar-Shalom, Y., 1996, Adaptive detection threshold optimization for tracking in clutter, , IEEE Transactions on Aerospace and Electronic Systems, 32, pp514-522.

Gershman, A. B. and Bohme, J. F., 1997, Adaptive beamforming algorithms with robustness against jammer motion, IEEE Transactions on Signal Processing, 45, pp1878-1885.

Gholson, N. H., and Moose, R.. L., 1977, Maneuver target tracking using adaptive state estimation, IEEE Transaction on Aerospace and Electronic System, 13, pp310-317.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M., 1993, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, IEE Proceedings-F, 140, pp107-113.

Gordon, N., 1997, A hybrid bootstrap filter for target tracking in clutter, IEEE Transactions on Aerospace and Electronic Systems, 33, pp353-358.

Grimble, M. J., 1980, A finite-time linear filter for discrete-time systems, International Journal of Control, 31, pp413-432.

Grimble, M. J., 1996, Robust filter design for uncertain systems defined by both hard and soft bounds, IEEE Transactions on Signal Processing, 44, pp1063-1071.

Gustafson, J., and Maybeck, P., 1992, Control of a large flexible space structure with moving-bank multiple model adaptive algorithms, Proceedings of the 31st Conference on Desision and Control, Tucson, Arizona, pp1273-1278.

Guojie, T., Wen, C., and Soh, Y. C., 1997, Identification for systems with bounded noise, IEEE Transactions on Automatic Control, 42, pp996-1001.

Gutman, P. O., and Velger, M., 1988, Tracking targets with unknown process noise variance using adaptive Kalman filtering, Proceedings of the 27th Conference on Decision and Control, Austin, Texas, pp869-874.

Gutman, P. O., and Velger, M., 1990, Tracking targets using adaptive Kalman filtering, IEEE Transactions on Aerospace and Electronic Systems, 26, pp691-699.

Guu, J. A., and Wei, C. H., 1991, Maneuvering target tracking using IMM method at high measurement frequency, IEEE Transaction on Aerospace and Electronic Systems, 27, pp524-519.

Hägglund, T., 1983, New estimation techniques for adaptive control, Ph.D. thesis, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Hakvoot, R. G., Van, P. M. J., and Hof, D., 1997, Identification of probabilistic system uncertainty regions by explicit evaluation of bias and variance errors, IEEE Transactions on Automatic Control, 42, pp1516-1528.

Hampton, R. L. T., and Cooke, J. R., 1973, Unsupervised tracking of maneuvering vehicles, IEEE Transactions on Aerospace and Electronic Systems, 9, pp197-207.

Helmick, R. E., Blair, W. D., and Hoffman, S. A., 1995, Fixed-interval smoothing for Markovian switching systems, IEEE Transactions on Information Theory, 41, pp1845-1855.

Helmick, R. E., Blair, W. D., and Hoffman, S. A., 1996, One-step fixed-lag smoothers for Markovian switching systems, IEEE Transactions on Automatic Control, 41, pp1051-1056.

Hirshberg, D., and Merhav, N., 1996, Robust methods for model order estimation, IEEE Transactions on Signal Processing, 44, pp620-628.

Ho, Y. C., and Lee, R. C. K., 1964, A Bayesian approach to problems in stochastic estimation and control, IEEE Transactions on Automatic Control, pp333-337.

Hong, L., 1999, Multirate interacting multiple model filtering for target tracking using multirate models, IEEE Transactions on Automatic Control, 44, pp1326-1340.

Howard, R. A., 1964, System analysis of semi-Markov processes, IEEE Trans. Mil. Electron., 8, pp114-124.

Ignagni, M. B., 1981, An alternate derivation and extension of Friedland's two-stage Kalman estimator, IEEE Transactions on Automatic Control, 26, pp746-750.

Ignagni, M. B., 1990, Separate-bias Kalman estimator with bias state noise, IEEE Transactions on Automatic Control, 35, pp338-341.

Isaksson, A. J., and Gustafsson, F., 1995, Comparison of some Kalman filter based methods for manoeuver tracking and detection, Proceedings of the 34th Conference on Decision & Control, pp1525-222.

Jakeman, A., and Young, P., 1979, Joint parameter/state estimation, Electronics Letters, 15, pp582-583.

Jazwinski, A. H., 1970, Stochastic Processes and Filtering Theory, Academic Press, New York.

Jilkov, V. P., and Angelova, D. S., 1996, Performance evaluation and comparison of variable structure multiplemodel algorithms for tracking maneuvering radar targets, In Proceedings of the 26th European Microwave Conference-EuMC'96, pp332-336.

Jilkov, V. P., Angelova, D. S., and Semerdjiev, T. A., 1999, Design and comparison of mode-set adaptive IMM algorithms for maneuvering target tracking, IEEE Transactions on Aerosoace and Electronic Systems, 35, pp343-350.

Juditsky, A., and Priouret, P., 1994, A Robust Algorithm for Random Parameter Tracking, IEEE Transactions on Automatic Control, 39, pp1211-1221.

Kailath, T., 1975, Supplement to " A survey of data smoothing", Automatica, 11, pp109-111.

Keesman, K., and Straten, G. V., 1989, Identification and prediction propagation of uncertainty in models with bounded noise, International Journal of Control, 49, pp2259-2269.

Kendrick, J. D., Maybeck, P., and Reid, J. G., 1981, Estimation of aircraft target motion using orientation measurements, IEEE Transactions on Aerospace and Electronic Systems, 17, pp254-260.

Kershaw, D. J., 1996, A contribution to performance prediction for probabilistic data association tracking filters, IEEE Transactions on Aerospace and Electronic Systems, 32, pp1143-1148.

Kim, J. H., and Oh, J. H., 2000, Robust state estimator of stochastic linear systems with unknown disturbances, IEE Proceedings: Control Theory and Applications, 147, pp224-228.

Kirubarjan, T., Bar-Shalom, Y., Blair, W. D., and Watson, G. A., 1998, IMMPDAF for radar management and tracking benchmark with ECM, IEEE Transactions on Aerospace and Electronic Systems, 34, pp1115-1134.

Kirlin, R. L., and Moghaddamjoo, A., 1986, Robust adaptive Kalman filtering for systems with unknown step inputs and non-Gaussian measurement errors, , IEEE Transactions on Acoustics, Speech, and Signal Processing, 34, pp252-263.

Kirubarajan, T., Bar-Shalom, Y., Pattipati, K. R., and Kadar, I., 2000, Ground target tracking with variable structure IMM estimator, IEEE Transactions on Aerospace and Electronic Systems, 36, pp26-46.

Koch, W., and Keuk, G. V., 1997, Multiple hypothesis track maintenance with possibly unresolved measurements, IEEE Transactions on Aerospace and Electronic Systems, 33, pp883-892.

Koch, W., and Fkie, F., 2000, Fixed-interval retrodiction approach to Bayesian IMM-MHT for maneuvering multiple targets, IEEE Transactions on Aerospace and Electronic Systems, 36, pp2-14.

Korn, J., Gully, S. w., and Willsky, A. S., 1982, Application of the generalized likelihood ratioo algorithm to maneuver detection and estimation, In Proceedings of the 1982 American Control Conference, pp792-798.

Kumar, K. A., Rao G. R., and Murukutla, N. L. M., 1989, An improved tracking algorithm for target detection, IEEE Conference on Intelligent System, pp235-238.

Kumar, A., Bar-Shalom, Y., and Oron, E., 1995, Precision tracking based on segmentation with optimal layering for imaging sensors, IEEE Trnsactions on Pattern analysis and Machine Intelligence, 17, pp182-188.

Kumar, A., Bar-Shalom, Y., and Oron, E., 1995, Precision tracking based on segmentation with optimal layering for imaging sensors, IEEE Trnsactions on Pattern analysis and Machine Intelligence, 17, pp182-188.

Kumar, A., Bar-Shalom, Y., and Oron, E., 1995, Precision tracking based on segmentation with optimal layering for imaging sensors, IEEE Trnsactions on Pattern analysis and Machine Intelligence, 17, pp182-188.

Lainiotis, D. G., and Sims, F. L., 1969, A recursive algorithm for the calculation of the adaptive Kalman filter weighting coefficients, IEEE Trans. Automat. Contr. 14.

Lanka, O., 1984, Circle manoeuvre classification for maneuvering radar targets tracking, Tesla Electronics, 17, pp10-17.

Layne, J. R., and Piyasena, U. C., 1997, Adaptive interacting multiple model tracking of manuevering targets, Digital Avionics Systems Conference, 16th DASC., AIAA/IEEE, 1, pp5.3-16-23.

Lehner, P. E., Laskey, K. B., and Dubois, D., 1996, An introduction to issues in higher order uncertainty, IEEE Transactions on Systems, Man, and Cybernetics, 26, pp289-292.

Ledet, C. J., and Trahan, R. E., 1990, Determining the discrete equivalent Kalman steady state gain for the LQR problem, IEEE Proceedings-Southeast conference, pp911-913.

Lerro, D., and Bar-Shalom, Y., 1993, Interacting multiple model tracking with target amplitude feature, IEEE Transactions on Aerospace and Electronic Systems, 29, pp494-508.

Li, X. R., and Bar-Shalom, Y., 1992, Mode-set adaptation in multiple model estimators for hybrid systems, In Proceedings of the 1992 American Control Conference, pp1794-1799.

Li, X. R., and Bar-Shalom, Y., 1993, Performance prediction of the interacting multiple model algorithm, IEEE Transactions on Aerospace and Electronic Systems, 29, pp755-771.

Li, X. R., and Bar-Shalom, Y., 1993, Design of interacting multiple model algorithm for tracking in air traffic control systems, Proceedinds of the 32nd Confrence on Decision and Control, pp906-911.

Li, X. R., and Bar-Shalom, Y., 1994, Detection threshold selection for tracking performance optimization, IEEE Transactions on Aerospace and Electronic Systems, 30, pp742-749.

Li, X. R., and Bar-Shalom, Y., 1994, A recursive multiple model approach to noise identification, IEEE Transactions on Aerospace and Electronic Systems, 30, pp671-684.

Li, X. R., 1996, Hybrid estimation techniques, In Control and Dynamic Systems: Advances in Theory and Applications, 76, pp2136-2287.

Li, X. R., and Bar-Shalom, Y., 1996, Multiple-model estimation with variable structure, IEEE Transactions on Automatic Control, 41, pp478-493.

Li, X. R., Zhang, Y., and Zhi, X., 1997, Multiple-model estimation with variable structure: Model-group switching algorithm, Proceedings of the 36th Conference on Decision & Control, pp3114-3119.

Lin, H. J., and Atherton, D. P., 1993, An investigation of the SFIMM algorithm for tracking manoeuvring targets, Proceedings of the 32nd Conference on Decision and Control, San Antonlo, Texas, pp930-935.

Lin, H. J., and Atherton, D. P., 1993, Investigation of IMM tracking algorithm for the manoeuvring target tracking, First IEEE Regional Conference on Aerospace Control Systems, pp113-117.

Magill, D. T., 1965, Optimal adaptive estimation of sampled stochastic processes, IEEE Transactions on Automatic Control, 10, pp434-439.

Maksarov, D. G., and Norton, J. P., 1996, State bounding with ellipsoidal set description of the uncertainty, International Journal of Control, 65, pp847-866.

Masreliez, C. J., and Martin, R. D., 1977, Robust Bayesian estimation for the linear model and robustifying the Kalman filter, IEEE Transactions on Automatic Control, 22, pp361-371.

Matasov, A. I., 1994, The Kalman-Bucy filter accuracy in the guaranteed parameter estimation problem with uncertain statistics, IEEE Transactions on Automatic Control, 39, pp635-639.

Maybeck, P. et al., 1979, A new tracker for air-to-air missile targets, IEEE Transactions on Automatic Control, 24, pp900-905.

Maybeck, P. S., Jensen, R. L., and Harnly, D. A., 1981, An adaptive extended Kalman filter for target image tracking, IEEE Transactions on Aerospace and Electronic Systems, 17, pp173-180.

Maybeck, P. S., 1982, Stochastic Models, Estimation, and Control, Volume 1 and 2, Academic Press, New York.

Maybeck, P., and Hentz, K. P., 1987, Investigation of moving-bank multiple model adaptive algorithms, AIAA Journal of Guidance, Control, and Dynamics, 10, pp90-96.

Mazor, E., Averbuch, A., Bar-Shalom, Y., and Dayan, J., 1998, Interacting multiple model methods in target tracking: A survey, IEEE Transactions on Aerospace and Electronic Systems, 34, pp103-122.

McAulay, R. J., and Denlinger, E., 1973, A decision-directed adaptive tracker, IEEE Transactions on Aerospace and Electronic Systems, 9, pp229-236.

McGarty, T. P., 1974, Stochastic Systems and State Estimation, John Wiley, New York.

McGinnity, S., and Irwin, G. W., 1998, Fuzzy logic approach to manoeuvering target tracking, IEE Proceeding - Radar Sonar and Navigation, 145, pp.337-341.

Meditch, J. S., 1967, Orthogonal projection and discrete linear smoothing, J. ASAM Control, 5, pp74-89.

Meditch, J. S., 1969, Stochastic optimal linear estimation and control, McGraw-Hill: New York.

Meditch, J. S., 1973, A survey of data smoothing for linear and non-linear dynanmic system, Automatic, 9, pp151-162.

Melsa, J. L., and Sage, A. P., 1973, An introduction to probability and stochastic process, Prentice-Hall, Inc., Englewood Cliffs, N. J.

Moore, J. B., 1973, Discrete-time fixed-lag smoothing algorithms, Automatica, 9, pp163-173.

Moose, R. L., 1975, An adaptive state estimation solution to the maneuvering target problem, IEEE Transactions on Automatic Control, 20, pp359-362.

Morrell, D. R., and Stirling, W. C., 1991, Set-valued filtering and smoothing, IEEE Transactions on Systems, Man, and Cybernetics, 21, pp184-193.

Morrell, D. R., 1993, Epistemic utility estimation, IEEE Transactions on Systems, Man, and Cybernetics, 23, pp129-140.

Morrell, D. R., and Stirling, W. C., 1993, Set-valued Kalman filtering, World Scientific, Singapore.

Munir, A., and Atherton, D. P., 1994, Maneuvering target tracking using adaptive interacting multiple model algorithm, Proceedings of the American Control Conference, Baltimore, Maryland.

Munir, A., and Atherton, D. P., 1995, Maneuvering target tracking using different turn rate models in the interacting multiple model algorithm, Proceedings of the 34th Conference on Decision & Control, pp2747-2751.

Munir, A., and Atherton, D. P., 1995, Adaptive interacting multiple model algorithm for tracking a manoeuvring target, IEE Proceedings on Radar, Sonar and Navigation, 142, pp11-17.

Myers, K. A., and Tapley, B. D., 1976, Adaptive sequential estimation with unknown noise statistics, IEEE Transactions on Automatic Control, pp520-523.

Moghaddamjoo, A., and Kirlin, R. L., 1989, Robust adaptive Kalman filtering with unknown inputs, IEEE Transactions on Acoustics, Speech, and Signal Processing, 37, pp1166-1175.

Moheimani, S. O. R., Savkin, A. V., and Petersen, I. R., 1998, Robust filtering, prediction, smoothing, and observability of uncertain systems, IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, 45, pp446-457.

Moose, R. L., 1975, An adaptive state estimation solution to the maneuvering target problem, IEEE Transactions on Automatic Control, 20, pp359-362.

Moose, R. L., Vanlandingham, H. F., and McCabe, D. H., 1979, Modeling and estimation for tracking maneuvering targets, IEEE Transactions on Aerospace and Electronic Systems, 15, pp448-456.

Neapolitan, R. E., 1996, Is higher-order uncertainty needed? IEEE Transactions on Systems, Man, and Cybernetics, 26, pp294-301.

Negron, C. D., 1977, Discrimination of motion dynamics for maneuvrering target by the application of statistical decision theory, In Proceedings of Conference on Advances in Target Tracking, Monterey, Calif.

Nishiyama, K., 1997, A nonlinear filter for estimating a sinusoidal signal and its parameters in white noise: on the case of a single sinusoid, IEEE Transactions on Signal Processing, 45, pp970-981.

Nishiyama, K., 1999, Robust estimation of a signal complex sinusoid in white noise-H filtering approach, IEEE Transactions on Signal Processing, 47, pp2853-2856.

Norton, J. P., 1976, Identification by optimal smoothing using integrated random walks, Proc. IEE, 123, pp451-452.

Norton, J. P., 1989, Identification of parameter bounds for ARMAX models from records with bounded noise, International Journal of Control, 45, pp375-390.

Park, Y. H., Seo, J. H. and Lee, J. G., 1995, Tracking using the variable-dimension filter with input estimation, IEEE Transactions on Aerospace and Electronic Systems, 31, pp399-408.

Politis, D. N., Leger, C. and Romano, J. P., 1992, Bootstrap technology and applications, Technometrics, 34, pp378-398.

Popoli, R., and Mendel, J., 1993, Estimation using subjective knowledge with tracking applications, IEEE Transactions on Aerospace and Electronic Systems, 29, pp610-623.

Potter, J. M., and Anderson, B. D. O., 1980, Partial prior information and decision making, IEEE Transactions on Systems, Man, and Cybernetics, 10, pp125-133.

Priel, B., and Shaked, U., 1983, Multivariable nyquist plot of discrete-time stationary optimal linear filters, IEE Proceedings-D, 130, pp128-130.

Priemer, R., and Vaeroux, A. G., 1971, On smoothing in linear discrete systems with time delays, Int. J. Control, 13, pp299-303.

Raghavan, V., Pattipati, K. R., and Bar-Shalom, Y., Bar-Shalom, 1992, Efficient square-root algorithms for PDA, IMM and IMMMPDA filters, Proceedings of the 31st Conferences on Decision and Control, Tucson, Arizons, pp3680-3685.

Rauch, H. E., 1963, Solutions to the linear smoothing problem, IEEE Transactions on Automatic Control, 8, pp371-372.

Rauch, H. E., Tung, F., and Streibel, C. T., 1965, Maximum likehood estimates of linear dynamic systems, AIAA J., 3, pp1445-1450.

Repperger, D. W., 1984, An implementation method for the discrete Kalman filter with applications to large-scale systems, Int. J. Control, 40, pp53-64.

Ricker, G. G., and Williams, J. R., 1978, Adaptive tracking filter for maneuvering targets, IEEE Transactions on Aerospace and Electronic Systems, 14, pp185-193.

Roecker, J. A., McGillem, C. D., 1989, Target tracking in maneuver-centered coordinates, IEEE Transactions on Aerospace and Electronic Systems, 25, pp836-843.

Rosin, P. L., 1999, Robust pose estimation, IEEE Transactions on Systems, Man, and Cybernetics, 29, pp297-303.

Rosin, P. L., 1999, Robust pose estimation, IEEE Transactions on Systems, Man, and Cybernetics, 29, pp297-303.

Rosin, P. L., 1999, Robust pose estimation, IEEE Transactions on Systems, Man, and Cybernetics, 29, pp297-303.

Sage, A. P., and Melsa, J. L., 1971, Estimation theory with application to communication theory, McGraw-Hill, New York.

Sanyal, P., and Shen, C. N., 1974, Bayes decision rule for rapid detection and adaptive estimation scheme with space applications, IEEE Trans. Automat. Contr. 19, pp228-231.

Schweppe, F. C., 1968, Recursive state estimation: Unknown but bounded errors and system inputs, IEEE Transactions on Automatic Control, 13, pp22-28.

Shaked, U., and Souza, C. E. D., 1995, Robust minimum variance filtering, IEEE Transactions on Signal Processing, 43, pp2474-2483.

Shi, P., 1998, Filtering on sampled-data systems with parametric uncertainty, IEEE Transactions on Automatic Control, 43, pp1022-1027.

Singer, R. A., 1970, Estimating optimal tracking filter performance for manned maneuvring targets, IEEE Transactions on Aerospace and Electronics Systems 6, pp473-483.

Snow, P., 1991, Improved posterior probability estimates from prior and conditional linear constraint systems, IEEE Transactions on Systems, Man, and Cybernetics, 21, pp464-469.

Snow, P., 1996, The posterior probabilities of linearly constrainted priors and interval-bounded conditionals, IEEE Transactions on Systems, Man, and Cybernetics, 26, pp655-659.

Song, T. L., Ahn, J. Y., and Park, C., 1988, Suboptimal filter design with pseudomeasurements for target tracking, IEEE Transactions on Aerospace and Electronic Systems, 24, pp28-39.

Song, T. L., and Lee, D. G., 2000, Effective filtering of target glint, IEEE Transactions on Aerospace and Electronic Systems, 36, pp234-241.

Souza, C. E. D., Shaked, U., and Fu, M., 1995, Robust filtering for continuous time varying uncertain systems with deterministic input signals, IEEE Transactions on Signal Processing, 43, pp709-719.

Spingarn, K., and Weidemann, H. L., 1972, Linear regression filtering and prediction for tracking maneuvering aircraft vechicles, IEEE Transactions on Aerospace and Electronics Systems, 8, pp800-810.

Stirling, W. C., and Morrell, D. R., 1991, Convex Bayes decision theory, IEEE Transactions on Systems, Man, and Cybernetics, 21, pp173-183.

Stoffer, D. S., and Wall, K. D., 1991, Bootstrapping state-space models: Gaussian maximum likelihood estimation and the Kalman filter, Journal of the American Statistical Association, 86, pp1024-1033.

Tenney, R. R., Ballard, T. B., and Miller, L. E., 1977, A dual passive tracking filter for maneuvering targets, In Proceedings of Conference on Advances in Target Tracking, Monterey, Calif.

Tenney, R. R., Herbert, R. S., and Sandell, N. R., 1977, A tracking filter for maneuvering sources, IEEE Transactions on Automatic Control, 22, pp246-251.

Thorp, J. S., 1973, Optimal tracking of maneuvering targets, IEEE Transactions on Aerospace and Electronic Systems, 9, pp512-519.

Trulsson, E., 1983, Adaptive control based on explicit criterion minimization. Dissertation No. 106, Division of Automatic Control, Department of Electrical Engineering, Linköping University, Linköping , Sweden.

Tugniat, J. K., 1982, Detection and estimation for abruptly changing systems, Automatica , 18, pp607-615.

Vacher, P., Barret, I., and Gauvrit, M., 1992, Design of a tracking algorithm for an advanced ATC system, In Bar-Shalom, Y., (Dd.), Multitarget-Multisensor Tracking: Applications and Advances, Vol. II, Nrwood, MA: Artech House.

Vicino, A., and Milanese, M., 1991, Optimal inner bounds of feasible parameter set in linear estimation with bounded noise, IEEE Transactions on Automatic Control, 36, pp759-763.

Wang, I. H., Lee, J. G., and Sung, T. K., 1994, A modified input estimation technique using pseudoresiduals, IEEE Transactions on Aerospace and Electronic Systems, 30, pp591-598.

Wang, W. J., and Mau, L. G., 1997, Stabilization and estimation for perturbed discrete time-delay large-scale systems, IEEE Transactions on Automatic Control, 42, pp1277-1282.

Wang, Z., Guo, Z., and Unbehauen, H., 1997, Robust state estimation for systems for discrete-time systems with error variance constraints, IEEE Transactions on Automatic Control, 42, pp1431-1435.

Wang, Z., and Unbehauen, H., 1999, Robust state estimation for systems with error variance constraints: The continuous-time case, IEEE Transactions on Automatic Control, 44, pp1061-1065.

Wang, Z., and Huang, B., 2000, Robust filtering for linear systems with error variance constraints, IEEE Transactions on Signal Processing, 48, pp2463-2467.

Watson, G. A., and W. D. Blair, 1992, IMM algorithm for tracking targets that maneuver through coordinated turns, In Proceedings of Signal and Data Processing of Small Targets, SPIE 1698, pp236-247.

Watson, G. A., and W. D. Blair, 1995, Interacting acceleration compensation algorithm for tracking maneuvering targets, IEEE Transactions on Aerospace and Electronic Systems, 31, pp1152-1159.

Weston, P. F., and Norton, J. P., 1997, Detection and estimation of abrupt changes in input or state, International Journal of Control, 67, pp699-711.

Whang, I. H., Lee, J. G., and Sung, T. K., 1994, Modified input estimation technique using pseudoresiduals, IEEE Transactions on Aerospace and Electronic Systems, 30, pp220-228.

White, C. C., 1986, A posteriori representations based on linear inequality descriptions of a priori and conditional probabilities, IEEE Transactions on Systems, Man, and Cybernetics, 16, pp570-573.

Witsenhausen, H. S., 1968, Sets of possible states of linear systems given perturbed observations, IEEE Transactions on Automatic Control, pp556-558.

Worsley, W. H., 1980, Comparison of three extended Kalman filters for air-to-air tracking, M. S. Thesis, U. S. Air Force institute of Technology, Wright Patterson AFB, Ohio.

Wu, W. R., and Chang, D. C., 1996, Maneuvering target tracking with colored noise, IEEE Transactions on Aerospace and Electronic Systems, 32, pp1311-1320.

Wu, W. R., 1996, Maximum likelihood identification of glint noise, IEEE Transactions on Aerospace and Electronic Systems, 32, pp41-51.

Xie, L., 1995, On robust filtering for linear systems with parameter uncertainty, Proceedings of the 34th Conference on Decision & Control New Orleans, pp2087-2092Norton, J. P., 1975, Optimal smoothing in the identification of linear time-varying systems, IEE Proceedings, 122, pp663-668.

Yardimci, Y., Cetin, A. E., and Cadzow, J. A., 1998, Robust direction-of-arrival estimation in non-Gaussian noise, IEEE Transactions on Signal Processing, 46, pp1443-1451.

Zhang, Y., and Li, X. R., 1998, Detection and diagnosis of sensor and actuator failures using IMM estimator, IEEE Transactions on Aerospace and Electronic Systems, 34, pp1293-1313.

Zoubir, A. M., and Boashash, B., 1998, The bootstrap and its application in signal processing, IEEE Signal Processing Magazine, 15, pp56-76.

APPENDIX I

DERIVATION OF *Q*(*K*)

A first-order Markov system is obtained as follows,

$$\dot{a}(t) = -\alpha\, a(t) + w(t) \tag{I.1}$$

where $a(t)$ is an acceleration and $w(t)$ is a zero-mean white noise with variance $E\{w^2(\tau)\} = q\delta(t-\tau)$. And process noise variance q is given by $q = 2\alpha\sigma_m^2$, where σ_m is a variance of target manoeuvre acceleration.

The equations of *x*-direction and *y*-direction are similar, so here we only give the model in the *x*-direction. Defining a state vector $X(t)$ of position, velocity, and acceleration in the *x*-direction, the target dynamics model in the *x*-direction can be written by

$$\dot{X}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\alpha \end{bmatrix} X(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w(t) \tag{I.2}$$

The discrete form of the above equation is sought for digital implementation. Many sensors have a constant data rate, sampling target position every T seconds. The appropriate (discrete-time) target equations of motion for this application are given by

$$X(k+1) = \begin{bmatrix} 1 & T & \frac{1}{\alpha^2}[-1+\alpha T + e^{-\alpha T}] \\ 0 & 1 & \frac{1}{\alpha}[1-e^{-\alpha T}] \\ 0 & 0 & e^{-\alpha T} \end{bmatrix} X(k) + U(k) \tag{I.3}$$

where $U(k)$ is noise term.

$$U(k)=\int_{kT}^{(k+1)T}\begin{bmatrix}1 & (k+1)T-\tau & \dfrac{-1+\alpha[(k+1)T-\tau]+e^{-\alpha[(k+1)T-\tau]}}{\alpha^2}\\ 0 & 1 & \dfrac{1-e^{-\alpha[(k+1)T-\tau]}}{\alpha}\\ 0 & 0 & e^{-\alpha[(k+1)T-\tau]}\end{bmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix}w(\tau)d\tau$$

$$=\int_{kT}^{(k+1)T}\begin{bmatrix}\dfrac{-1+\alpha[(k+1)T-\tau]+e^{-\alpha[(k+1)T)-\tau]}}{\alpha^2}\\ \dfrac{1-e^{-\alpha[(k+1)T-\tau]}}{\alpha}\\ e^{-\alpha[(k+1)T-\tau]}\end{bmatrix}w(\tau)d\tau$$

$$=\int_{kT}^{(k+1)T}\begin{bmatrix}n_1(\tau)\\ n_2(\tau)\\ n_3(\tau)\end{bmatrix}w(\tau)d\tau \qquad \text{(I.4)}$$

The manoeuvre excitation covariance matrix $Q(k)$ satisfies, using (I.4),

$$Q(k)=E[U(k)U^{\mathrm{T}}(k)]$$

$$=\int_{kT}^{(k+1)T}\int_{kT}^{(k+1)T}\begin{bmatrix}n_1(\tau)\\ n_2(\tau)\\ n_3(\tau)\end{bmatrix}\begin{bmatrix}n_1(\tau) & n_2(\tau) & n_3(\tau)\end{bmatrix}2\alpha\sigma_m^2\delta(\tau-s)d\tau ds$$

$$=2\alpha\sigma_m^2\int_{kT}^{(k+1)T}\begin{bmatrix}n_1^2(\tau) & n_1(\tau)n_2(\tau) & n_1(\tau)n_3(\tau)\\ n_2(\tau)n_1(\tau) & n_2^2(\tau) & n_2(\tau)n_3(\tau)\\ n_3(\tau)n_1(\tau) & n_3(\tau)n_2(\tau) & n_3^2(\tau)\end{bmatrix}d\tau$$

In this equation,

$$n_1^2(\tau)=\frac{1}{\alpha^4}\{e^{-2\alpha[(k+1)T-\tau]}+1-2e^{-\alpha[(k+1)T-\tau]}+\alpha^2[(k+1)T-\tau]^2-2\alpha[(k+1)T-\tau]$$
$$+2\alpha[(k+1)T-\tau]e^{-\alpha[(k+1)T-\tau]}\}$$

$$n_1(\tau)n_2(\tau)=\frac{1}{\alpha^3}\{-e^{-2\alpha[(k+1)T-\tau]}+2e^{-\alpha[(k+1)T-\tau]}-\alpha[(k+1)T-\tau]e^{-\alpha[(k+1)T-\tau]}-$$
$$1+\alpha[(k+1)T-\tau]\}$$

$$n_1(\tau)n_3(\tau)=\frac{1}{\alpha^2}\{-e^{-\alpha[(k+1)T-\tau]}+\alpha[(k+1)T-\tau]e^{-\alpha[(k+1)T-\tau]}+e^{-2\alpha[(k+1)T-\tau]}\}$$

$$n_2^2(\tau)=\frac{1}{\alpha^2}\{1-2e^{-\alpha[(k+1)T-\tau]}+e^{-2\alpha[(k+1)T-\tau]}\}$$

$$n_2(\tau)n_3(\tau)=\frac{1}{\alpha}\{e^{-\alpha[(k+1)T-\tau]}-e^{-2\alpha[(k+1)T-\tau]}\}$$

$$n_3^2(\tau)=e^{-2\alpha[(k+1)T-\tau]}$$

Letting $Z=(k+1)T-\tau,$ it follows that

$$Q(k)=2\alpha\sigma_m^2\int_0^T\begin{bmatrix}h_{11}(Z) & h_{12}(Z) & h_{13}(Z)\\ h_{12}(Z) & h_{22}(Z) & h_{23}(Z)\\ h_{13}(Z) & h_{23}(Z) & h_{33}(Z)\end{bmatrix}dZ$$

where $h_{11}(Z)=\frac{1}{\alpha^4}[e^{-2\alpha Z}+1-2e^{-\alpha Z}+\alpha^2Z^2-2\alpha Z+2\alpha Z e^{-\alpha Z}]$

$$h_{12}(Z)=\frac{1}{\alpha^3}[-e^{-2\alpha Z}+2e^{-\alpha Z}-\alpha Z e^{-\alpha Z}-1+\alpha Z]$$

$$h_{13}(Z)=\frac{1}{\alpha^2}[-e^{-\alpha Z}+\alpha Z e^{-\alpha Z}+e^{-2\alpha Z}]$$

$$h_{22}(Z)=\frac{1}{\alpha^2}[1-2e^{-\alpha Z}+e^{-2\alpha Z}]$$

$$h_{23}(Z)=\frac{1}{\alpha}[e^{-\alpha Z}-e^{-2\alpha Z}]$$

$$h_{33}(Z)=e^{-2\alpha Z}$$

Utilizing the identities

$$\int_0^T dZ=T,\ \int_0^T e^{-\alpha Z}dZ=\frac{1-e^{-\alpha T}}{\alpha},\ \int_0^T e^{-2\alpha Z}dZ=\frac{1-e^{-2\alpha T}}{2\alpha},\ \int_0^T \alpha Z dZ=\frac{\alpha T^2}{2},$$

$$\int_0^T \alpha^2Z^2dZ=\frac{\alpha^2T^3}{3},\ \int_0^T \alpha Z e^{-\alpha Z}dZ=\frac{1}{\alpha}[1-e^{-\alpha T}-\alpha T e^{-\alpha T}]$$

it follows that

$$Q(k) = 2\alpha\sigma_m^2 \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix}$$

where

$$q_{11} = \frac{1}{2\alpha^5}[1 - e^{-2\alpha T} + 2\alpha T + \frac{2\alpha^3 T^3}{3} - 2\alpha^2 T^2 - 4\alpha T e^{-\alpha T}]$$

$$q_{12} = \frac{1}{2\alpha^4}[e^{-2\alpha T} + 1 - 2e^{-\alpha T} + 2\alpha T e^{-\alpha T} - 2\alpha T + \alpha^2 T^2]$$

$$q_{13} = \frac{1}{2\alpha^3}[1 - e^{-2\alpha T} - 2\alpha T e^{-\alpha T}]$$

$$q_{22} = \frac{1}{2\alpha^3}[4e^{-\alpha T} - 3 - e^{-2\alpha T} + 2\alpha T]$$

$$q_{23} = \frac{1}{2\alpha^2}[e^{-2\alpha T} + 1 - 2e^{-\alpha T}]$$

$$q_{33} = \frac{1}{2\alpha}[1 - e^{-2\alpha T}]$$

APPENDIX II

PUBLISHED AND TO BE PUBLISHED PAPERS ON THIS WORK

1. Yang, Z., and Norton, J. P., Tracking of manoeuvring targets using fixed-lag smoothing, IEE Control 2000 Conference in Cambridge, UK.
2. Yang, Z., and Norton, J. P., A variable-dimension filtering for tracking manoeuvring targets via a change-detection technique with fixed-lag smoothing
3. Yang, Z., and Norton, J. P., A new multiple-model filter tracking for manoeuvring targets

TRACKING OF MANOEUVRING TARGETS USING FIXED-LAG SMOOTHING

Zhufang Yang and J. P. Norton
School of Electronic & Electrical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT,UK
Author for correspondence: email j.p.norton@bham.ac.uk, tel. +44 121 414 4301

Key words: Kalman filter, IMM, smoothing

Abstract
This paper discusses an improved method for tracking a manoeuvring target in two dimensions. The proposed tracking scheme consists of manoeuvre detection and tracking taking account of the manoeuvre. The manoeuvre is detected by testing a normalised squared-acceleration statistic, with the acceleration components estimated by fixed-lag smoothing (with a small lag) using an augmented-state filter. The tracking is performed by a conventional variable-dimension Kalman filter (KF) (with state augmented by accelerations during manoeuvres). The proposed manoeuvre-detection technique gives quicker detection than the classical normalised squared-innovations statistic, in spite of the smoothing lag. In tracking, the lag associated with smoothed estimates may be acceptable in track reconstruction (e.g. to help in identifying target type or to resolve ambiguities such as crossing targets or the effects of glint). The track quality yielded by optimal smoothing, with a model matched to the presence or absence of a manoeuvre, is shown to be superior to that produced by variable-dimension filtering alone, and comparable with or better than IMM filtering. The computational load is modest, comparable to that of IMM with only 2 or 3 filters.

I. INTRODUCTION

Many different approaches for tracking a manoeuvring target have been considered in the literature. Approaches based on Kalman filtering include the early work of Singer (1970), who augmented the target motion model with the target acceleration, represented as a first-order autoregressive process. The augmented filter tracks a manoeuvring target well but its performance degrades, as compared with a Kalman filter that assumes no manoeuvres, during constant-velocity, straight-line motion. A common method is to use a non-manoeuvring target model for tracking a target moving at a constant velocity and then switch to an appropriate manoeuvring model, when a target manoeuvre is detected.

Chan (1979, 1982) and Bogler (1987) proposed an adaptive Kalman filter for tracking manoeuvring targets, using input estimation (IE). Their method is to construct a estimate of the input (manoeuvre forcing) from the innovation of the non-manoeuvring filter using least squares, then to correct the state estimate with the estimated input.

The variable-dimension (VD) filter was suggested by Bar-Shalom and Birmiwal (1982). In it, the target-motion model is changed during manoeuvres by introducing extra state components, accelerations, estimated recursively along with other state variables consisting of position and velocity. The augmented state model reverts to the normal model on detection of the end of the manoeuvre. The performance of the VD filter is superior to that of the IE filter and its computational requirements less. However, the VD algorithm presumes that the target starts to manoeuvre at the starting point of a sliding window running back from when the manoeuvre is detected, but the actual and assumed manoeuvres may differ in timing, leading to large tracking errors. The VD filter has to reconstruct the process noise covariance and the state estimates within the sliding window when changing to the manoeuvre model. Furthermore, the filter uses only measurements at the start of the sliding window to initialise the augmented filter. This also may increase the tracking error.

In several recent publications (Bar-Shalom, 1989; Lin, 1993; Mazor, 1998), the interacting multiple model (IMM) algorithm has been suggested as useful for tracking a manoeuvring target. The IMM algorithm is a suboptimal hybrid filter that is recognised as one of the most cost-effective state-estimation schemes. Its main feature is its ability to estimate the state of a dynamic system which can switch between many behaviour modes. In particular, the IMM filter can act as a self-adjusting

variable-bandwidth filter for tracking manoeuvring targets. However, the IMM filter incorporates a large number of filters to cover the possible manoeuvre motions, resulting in heavy computation. To reduce the computation load, a variable-dimension IMM (Bar-Shalom, 1989) has been suggested. This requires a prior suitable choice of process noise covariance of the manoeuvring model according to the manoeuvre input level, and a prior choice of mode transition probability, which may be difficult to supply. Poor choices may degrade its performance compared with Kalman filters based on correct manoeuvre and non-manoeuvre models.

Another approach is to modify the KF by constant changes (Bar-Shalom and Fortmann, 1988), multiplying the state-error covariance by a scalar at every sampling instant to compensate for manoeuvre-model error. Thus past data are discounted and the target-dynamics model is made more responsive by attaching a higher state-error covariance.

Previous work has focused on linear motion models and optimal filtering, i.e., optimal estimation of the state at time t_j, based upon knowledge of all measurements taken up to time t_j. However, optimal smoothing (retrospective state estimation) also has a long history. It receives less attention than optimal filtering because its recursive implementation is more complicated than that of filtering. However, optimal smoothing is potentially useful to help identify target type, resolve manoeuvre ambiguities and reconstruct tracks. The smoothing algorithm can be divided into two filtering problems: one using initial conditions and the "past" history of measurements {**z**(1) , **z**(2) , …, **z**(j)} and the other incorporating only "future" measurements {**z**(j+1) , **z**(j+2) , …, **z**(k-1) , **z**(k)}, with the smoothed estimate being the optimal combination of the outputs from these two filters. The additional information contained in the measurements taken after t_j improves estimation accuracy. This paper will consider fixed-lag smoothing, estimating **x**(k-L) from {**z**(1) , **z**(2) , …, **z**(k)} with lag L constant.

Classical manoeuvre detection computes a χ^2 statistic from the innovations, but typically suffers much delay. This paper suggests combining the variable-dimension filter and fixed-lag smoothing, using a χ^2 statistic based on estimated accelerations to detect the beginning and end of manoeuvres.

The proposed manoeuvre-detection method is to test the acceleration statistic with the acceleration components estimated by fixed-lag smoothing (with a small lag) using the acceleration-augmented variable-dimension filter. For tracking, as distinct from manoeuvre detection, the unaugmented standard Kalman filter is employed for non-manoeuvring and the variable-dimension filter for manoeuvring motion. The state-error covariance is multiplied by a scalar temporarily at the end and beginning of manoeuvres to make the filter more responsive to the future target manoeuvres, and fixed-lag smoothing is used to improve state estimation in both filters, augmented and unaugmented.

Section II describes the variable-dimension filter. Section III reviews fixed-lag smoothing. The proposed method for manoeuvre detection and tracking is described in Section IV. In Section V, the results of Monte Carlo simulation runs of several tracking techniques on the same system with the same disturbances are presented.

II. VARIABLE-DIMENSION FILTER

The variable-dimension filter adds extra state variables once a manoeuvre is detected. The extent of the manoeuvre as detected is then used to estimate the extra state components, and corrections are made to the other state variables. Tracking employs the augmented state model until it reverts to the normal model on detection of the end of the manoeuvre.

In the absence of manoeuvre, the target motion is modelled as constant-velocity motion in a plane, subject to variations in velocity induced by piecewise constant zero-mean accelerations with white sample-instant values. The corresponding state equation is

$$\mathbf{x}(k+1)=\mathbf{F}\mathbf{x}(k)+\mathbf{G}\mathbf{w}(k) \qquad (2.1)$$

where, using Cartesian coordinates, the state is

$$\mathbf{x}=[x \;\; y \;\; \dot{x} \;\; \dot{y}\,]^T$$

$$\mathbf{F}=\begin{bmatrix}1 & 0 & T & 0\\ 0 & 1 & 0 & T\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix},\ \mathbf{w}(k)=[w_1(k)\quad w_2(k)]^T$$

$$\mathbf{G}=\begin{bmatrix}T^2/2 & 0\\ 0 & T^2/2\\ T & 0\\ 0 & T\end{bmatrix},\ \mathbf{Ew}(k)=\mathbf{0},\ \mathbf{E}[\mathbf{w}(i)\mathbf{w}(j)^T]=\mathbf{Q}\delta_{ij}$$

and T is the sampling interval. The initial state estimate is $\hat{\mathbf{x}}(0|0)$ with covariance $\mathbf{P}(0|0)$. Position measurements (transformed to Cartesian coordinates) are made, so

$$\mathbf{z}(k)=\mathbf{H}\mathbf{x}(k)+\mathbf{v}(k) \qquad (2.2)$$

where

$$\mathbf{H}=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\end{bmatrix},\ \mathbf{Ev}(k)=\mathbf{0},\ \mathbf{E}[\mathbf{v}(i)\mathbf{v}(j)^T]=\mathbf{R}\delta_{ij}$$

In the manoeuvring model, the augmented state is

$$\mathbf{x}^m=[x\quad y\quad \dot{x}\quad \dot{y}\quad \ddot{x}\quad \ddot{y}\ ]^T$$

and the state equation is

$$\mathbf{x}^m(k+1)=\mathbf{F}^m\mathbf{x}^m(k)+\mathbf{G}^m\mathbf{w}^m(k) \qquad (2.3)$$

where

$$\mathbf{F}^m=\begin{bmatrix}1 & 0 & T & 0 & T^2/2 & 0\\ 0 & 1 & 0 & T & 0 & T^2/2\\ 0 & 0 & 1 & 0 & T & 0\\ 0 & 0 & 0 & 1 & 0 & T\\ 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix},\quad \mathbf{G}^m=\begin{bmatrix}T^2/2 & 0\\ 0 & T^2/2\\ T & 0\\ 0 & T\\ 1 & 0\\ 0 & 1\end{bmatrix}$$

$$\mathbf{w}^m=[w_{m1}\quad w_{m2}]^T,\ \mathbf{Ew}^m(k)=\mathbf{0},\ \mathbf{E}[\mathbf{w}^m(i)\,\mathbf{w}^m(j)^T]=\mathbf{Q}^m\delta_{ij}$$

The process noise covariance $\mathbf{Q}^m$ is selected according to the estimated acceleration. The process noise standard deviation is usually taken as 5% of the estimated acceleration (Bar-Shalom and Fortmann, 1988).

The corresponding observation equation is

$$\mathbf{z}(k)=\mathbf{H}^m\mathbf{x}(k)+\mathbf{v}(k) \qquad (2.4)$$

where

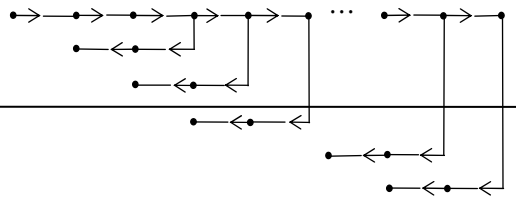
$$\mathbf{H}^m=\begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0\end{bmatrix}$$

III. FIXED-LAG SMOOTHING

Figure 1 illustrates the progress of fixed-lag smoothing with a fixed lag of 2 samples.

The optimal fixed-lag smoothed estimate of the state at time j=k-L is based on measurement data up to time k, where k>j; it is $\hat{\mathbf{x}}(j|k)=\mathbf{E}\{\mathbf{x}(j)\,|\,\mathbf{Z}(k)\}$, where $\mathbf{Z}(k)\equiv\{\mathbf{z}(i), i=1, 2, \ldots, k\}$. Meditch (1969) gives an algorithm for fixed-lag smoothing that is complicated. A simpler approach is that the fixed-lag smoothed estimate is provided using the fixed-interval smoother described by equations (3.1)-(3.3) (Rauch, 1963; Rauch et al., 1965). After each step of the forward filtering sweep, optimal smoothing moves backward by the fixed lag:

Forward filtering sweep yields $\mathbf{P}(k|k)$, $\hat{\mathbf{x}}(k|k)$, $\mathbf{P}(k+1|k)$ and $\hat{\mathbf{x}}(k+1|k)$.



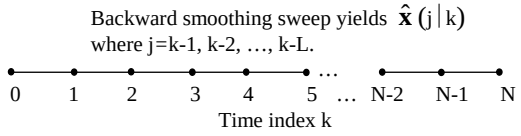


Figure 1. Procedure for fixed-lag smoothing.

$$\hat{\mathbf{x}}(j|k)=\hat{\mathbf{x}}(j|j)+\mathbf{A}(j)[\hat{\mathbf{x}}(j+1|k)-\hat{\mathbf{x}}(j+1|j)] \quad (3.1)$$

where the smoothing gain $\mathbf{A}(j)$ is given by

$$\mathbf{A}(j)=\mathbf{P}(j|j)\mathbf{F}^{T}\mathbf{P}^{-1}(j+1|j) \quad (3.2)$$

and j=k-1, k-2, …, k-L, where L is the fixed lag.
The error covariance matrix for the smoothed estimates is given by

$$\mathbf{P}(j|k)=\mathbf{P}(j|j)+\mathbf{A}(j)[\mathbf{P}(j+1|k)-\mathbf{P}(j+1|j)]\mathbf{A}^{T}(j) \quad (3.3)$$

Details of the fixed-lag smoothing can be found in (Brown, 1992).

IV. PROPOSED METHOD FOR MANOEUVRE DETECTION AND TRACKING

Manoeuvre detection

One of the simplest detection methods uses the normalised innovations squared

$$\varepsilon_{\nu}(k)=\nu_k^{T}\mathbf{S}_k^{-1}\nu_k \quad (4.1)$$

which is a chi-square variate with n_z (the dimension of measurement) degrees of freedom. A threshold is set up based on the target model (for the non-manoeuvring situation):

$$P\{\varepsilon_{\nu}(k)\le \varepsilon_{max}\}=1-\alpha \quad (4.2)$$

and if the threshold is exceeded, a manoeuvre is assumed to have occurred.
A less noise-sensitive method uses a windowed sample mean of the normalised innovations squared:

$$\varepsilon_{\nu}(k)=\frac{1}{N}\sum_{i=k-N+1}^{k}\nu_i^{T}\mathbf{S}_i^{-1}\nu_i \quad (4.3)$$

$N\varepsilon_{\nu}(k)$ is chi-square with Nn_z degrees of freedom.

However, manoeuvre detection by testing this statistic incurs considerable delay.

Instead, we use the augmented model throughout the process, with process noise covariance the same as the normal model, and fixed-lag smoothing is used to provide better estimates of the accelerations and their error covariance. The statistic and test based on smoothed estimated accelerations are:

$$\delta_a(j)=\hat{\mathbf{x}}_a^{T}(j|k)\ \mathbf{P}_a^{-1}(j|k)\ \hat{\mathbf{x}}_a(j|k) \quad (4.4)$$

$$P\{\delta_a(j)\le\varepsilon_{max}\}=1-\alpha \quad (4.5)$$

where $\hat{\mathbf{x}}_a(j|k)$ is the smoothed estimate of the acceleration components, $\mathbf{P}_a(j|k)$ is the corresponding block from the smoothed covariance matrix, and $\delta_a(j)$ has a chi-square distribution with N_a (the dimension of acceleration) degrees of freedom. When the sum

$$\mu_a(j)=\sum_{i=j-p+1}^{j}\delta_a(i) \quad (4.6)$$

exceeds a specified threshold, the hypothesis that a manoeuvre is taking place is accepted, at which point the estimator switches from the quiescent model to the manoeuvring model. If the accelerations drop rapidly to zero, the smoothed acceleration estimates will decrease in size monotonically. Thus, when the statistic falls twice in succession, the end of the manoeuvre is assumed and the estimator switches from the manoeuvre model to the quiescent model. A more conservative method would be to apply the test to p successive values of the statistic with p>2, at the price of greater delay.

The simulation in Section V will indicate that the test based on normalised smoothed acceleration squared is reliable and quick for detecting the beginning and end of manoeuvres.

Manoeuvre tracking

Manoeuvre tracking has the following 3 steps:

1. In the absence of manoeuvre, tracking is by a Kalman filter, using the unaugmented model, improved by the fixed-lag smoothing.
2. When the manoeuvre is detected, the state is augmented and the initial state estimate of manoeuvre model is provided by the smoothed estimate in manoeuvre detection, to avoid having to reconstruct the state estimate. The smoothed state-error covariance is multiplied by a scalar $\phi>1$ at the instant when the manoeuvre is detected, and regarded as the initial error covariance of the manoeuvre model. The process-noise covariance during manoeuvres is taken to be the same as that in the unaugmented filter.
3. When the end of the manoeuvre is detected, the target-motion model is returned to normal. The initial state estimate is provided by the smoothed estimate in Step 2 and the smoothed error covariance is multiplied by a scalar $\phi>1$ and regarded as the initial error covariance for the non-manoeuvre model, with optimal smoothing again improving the estimate.

V. SIMULATION RESULTS

Consider a target whose position in the *x*-*y* plane is sampled every T=10s. The target is on a constant course at constant speed until t=400s, when it starts to manoeuvre, and it completes the manoeuvre after 20 sampling periods. The process noise covariance $\mathbf{Q}= 10^{-5}\mathbf{I}$ and the measurement noise covariance has $R_{11}=R_{22}=2500\text{m}^2$ and $R_{12}=R_{21}=0\text{m}^2$. The initial conditions of the target are $x(0)=2000$m, $\dot{x}(0)=0$m/s, $y(0)=10000$m, and $\dot{y}(0)=-15$m/s. The manoeuvre is the result of the acceleration input

$$u_x=u_y=0.075\text{m/s}^2, \quad 400\text{s} \le t \le 600\text{s}.$$

The IMM filter consists of one 4-state model with acceleration zero and process noise $\mathbf{Q}=10^{-5}\mathbf{I}$, an other 4-state model with constant acceleration 0.075m/s^2 and process noise $\mathbf{Q}=10^{-5}\mathbf{I}$, and a 6-state model with process noise $\mathbf{Q}=10^{-3}\mathbf{I}$. The process noise is the random part of the acceleration increment over a sampling period in all cases. Note that the process noise $\mathbf{Q}$ for the 6-state model is taken to be about $(u/2.5)^2$ in order that the IMM filter has a rapid jump to a non-zero acceleration from zero and then a jump back to zero acceleration at termination of the manoeuvre. The Markov transition probability matrix governing the transition between these 3 models is chosen as $\begin{bmatrix} 0.95 & 0.05 & 0 \\ 0.05 & 0.95 & 0 \\ 0.33 & 0.34 & 0.33 \end{bmatrix}$

The VD filter is constructed by two models, one the standard 4-state model in the absence of manoeuvre, the other a 6-state model during manoeuvres. The standard deviation of process noise in the manoeuvre model was taken to be 5 percent of the estimated acceleration. The manoeuvre was detected by the statistic of normalised innovations squared, the threshold was taken to be 18.3 in equation (4.3) according to 0.95 confidence region for a 10-degrees-of-freedom chi-square distribution and the end of manoeuvre was also detected by the statistic of normalised innovations squared.

In the fixed-lag smoothing, the lag was 2 sample intervals. The threshold for $\mu_a(k)$, (given in equation (4.6) with p=2 and $\alpha=0.05$), was 9.49 according to 0.95 confidence region for a 4-degrees-of-freedom chi-square distribution. The scalar ϕ of multiplication of the state covariance was 100 at the points where the manoeuvre end and beginning were detected.

The simulation results are shown in Figures 2-9. Figures 2 and 3 present one-run results of manoeuvre detection by testing the statistic of normalised innovations squared and the statistic of normalised smoothed-accelerations squared respectively. Figures 4 to 9 show the average estimation errors found, averaged

over 20 Monte Carlo runs for each filter. RMS errors are shown for both the position and velocity errors. The average errors for two x and y directions are similar and thus only the errors in the x direction are displayed.

Figure 3 shows that detection using the statistic of normalised smoothed-accelerations squared is faster than that using normalised innovations squared (Figure 2).

Figures 8 and 9 show that the performance of the proposed method is much better than that of VD filtering alone, comparing with Figures 6 and 7 respectively. Figures 8 and 9 also show that the track quality yielded by optimal smoothing is much better

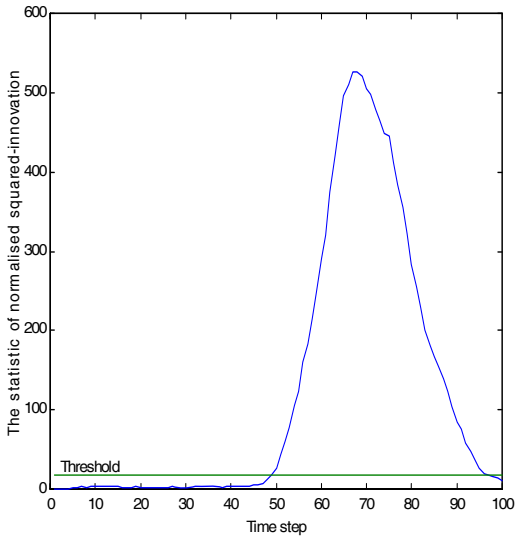


Figure 2. Statistic of normalised innovations squared.

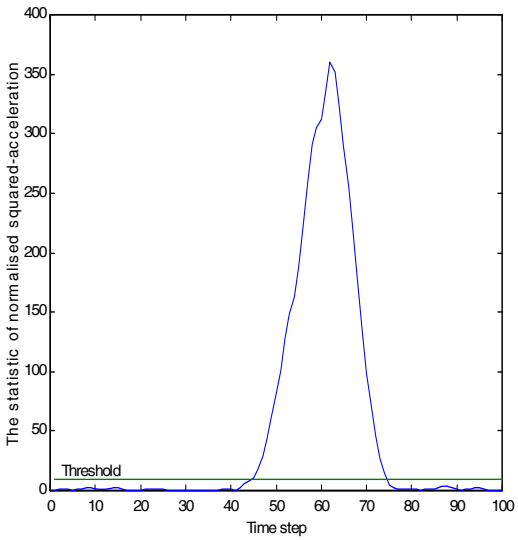


Figure 3. Statistic of normalised smoothed-accelerations squared.

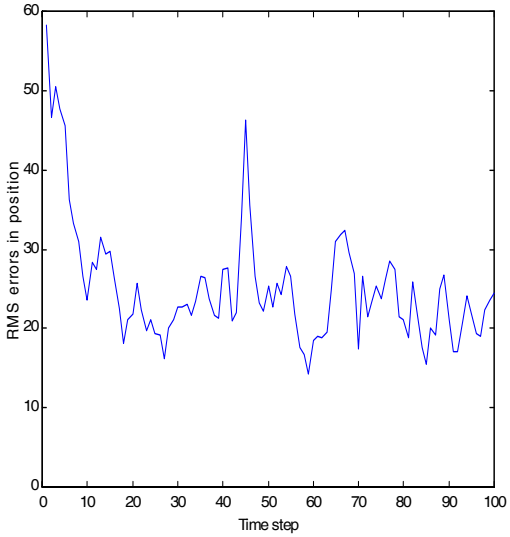


Figure 4. RMS errors in position using IMM.

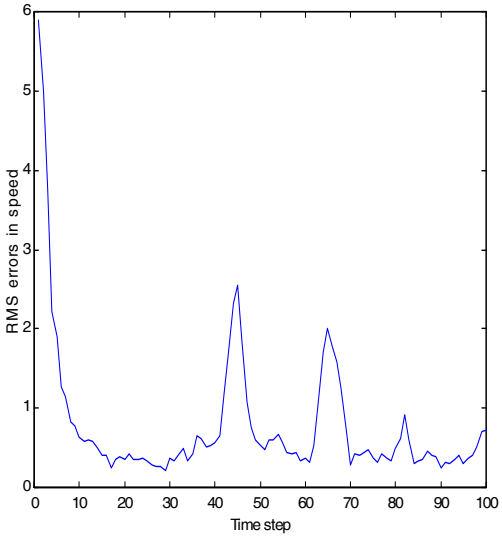


Figure 5. RMS errors in speed using IMM.

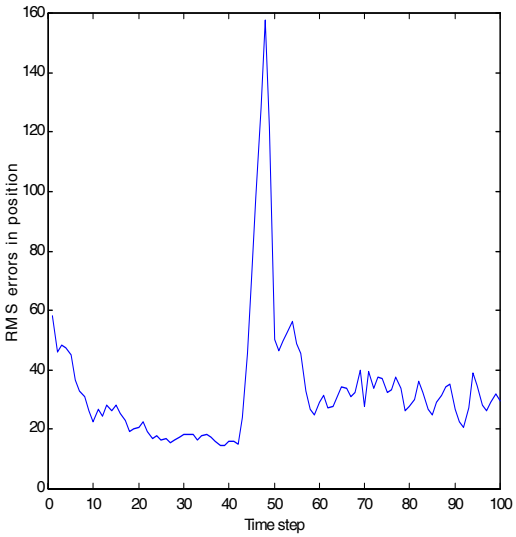


Figure 6. RMS errors in position using VD.

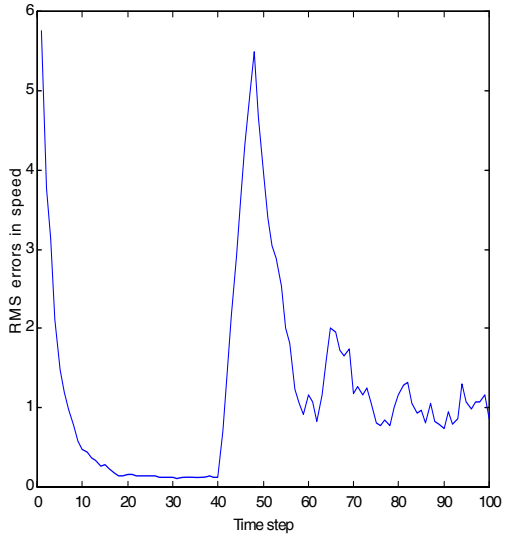


Figure 7. RMS errors in speed using VD.

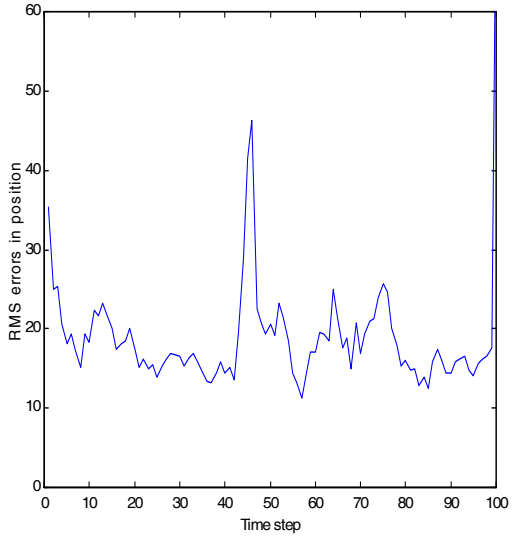


Figure 8. RMS errors in position using smoothing.

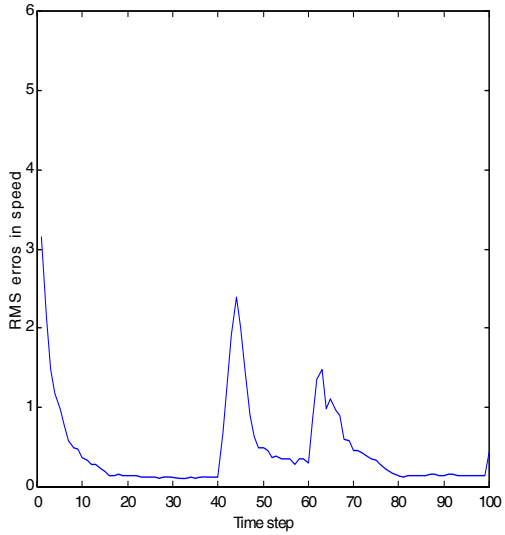


Figure 9. RMS errors in speed using smoothing.

than IMM filtering in the absence of manoeuvre and comparable with or better than with IMM filtering in the presence of manoeuvre, comparing with Figures 4 and 5 respectively.

VI. CONCLUSIONS

It has been shown that the proposed manoeuvre-detection technique, testing a normalised smoothed-acceleration statistic, gives quicker detection than the classical normalised squared-innovations statistic. The proposed tracking with fixed-lag smoothing performs much better than VD filtering alone, and also much better than IMM filtering in the absence of manoeuvre and better than or comparable with IMM filtering during manoeuvres. The proposed scheme is simple in concept and the computational load is modest, comparable to that IMM with only 2 or 3 filters. It also has the advantage of being implementable without prior knowledge of the manoeuvre characteristics of the target, with minimal design parameters to select.

REFERENCES

Bar-Shalom, Y., and Birmiwal, K., 1982, Variable dimension filter for maneuvering target tracking, IEEE Transactions on Aerospace and Electronic Systems, 18, pp621-629.

Bar-Shalom, Y., and Fortmann, T. E., 1988, Tracking and data association, Academic Press, New York.

Bar-Shalom, Y., 1989, Tracking a maneuvering target using input estimation versus the Interacting Multiple Model algorithm. IEEE Transactions on Aerospace and Electronics Systems, 25, pp296-300.

Bogler, P. L., 1987, Tracking a maneuvering target using input estimation, IEEE Transactions on Aerospace and Electronic Systems, 23, pp298-310.

Brown, R. G., and Hwang, Patrick Y. C., Introduction to random signals and applied Kalman filtering, 2nd ed., John Wiley & Sons, Inc.

Chan, Y. T., Hu, A. G. C., and Plant, J. B., 1979, A Kalman filter based tracking scheme with input estimation, IEEE Transactions on Aerospace and Electronic Systems, 15, pp237-244.

Chan, Y. T., Plant, J. B., and Bottomley, J. R. T., 1982, A Kalman tracker with a simple input estimator, IEEE Transactions on Aerospace and Electronic Systems, 18, pp235-241.

Lin, H. J., and Atherton, D. P., 1993, Investigation of IMM tracking algorithm for the manoeuvring target tracking. First IEEE Regional Conference on Aerospace Control Systems, pp113-117.

Mazor, E., Averbuch, A., Bar-Shalom, Y., and Dayan, J., 1998, Interacting multiple model methods in target tracking: a survey, IEEE Transactions on Aerospace and Electronic Systems, 34, pp103-123.

Meditch, S., 1969, Stochastic optimal linear estimation and control, McGraw-Hill, New York.

Rauch, H. E., 1963, Solutions to the linear smoothing problem, IEEE Transactions on Automatic Control, 8, pp371-372.

Ranch, H. E., Tung, F., and Streibel, C. T., 1965, Maximum likelihood estimates of linear dynamic systems, AIAA J., 3, pp1445-1450.

Singer, R. A., 1970, Estimating optimal tracking filter performance for manned maneuvering targets, IEEE Transactions on Aerospace and Electronic Systems, 14, pp473-483.