



University of Pennsylvania  
**ScholarlyCommons**

---

Technical Reports (CIS)

Department of Computer & Information Science

---

February 1990

## Analysis of the Effects of Model Mismatch and Flat MMF for Estimating Particle Motion

Siu-Leong lu  
*University of Pennsylvania*

Follow this and additional works at: [https://repository.upenn.edu/cis\\_reports](https://repository.upenn.edu/cis_reports)

---

### Recommended Citation

Siu-Leong lu, "Analysis of the Effects of Model Mismatch and Flat MMF for Estimating Particle Motion", . February 1990.

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-90-10.

This paper is posted at ScholarlyCommons. [https://repository.upenn.edu/cis\\_reports/543](https://repository.upenn.edu/cis_reports/543)  
For more information, please contact [repository@pobox.upenn.edu](mailto:repository@pobox.upenn.edu).

---

# Analysis of the Effects of Model Mismatch and Flat MMF for Estimating Particle Motion

## Abstract

In this report, we analyze the performance degradation due to three classes of model mismatch: parameter jumping, undermodeling and overmodeling, in estimating the particle motion by using the orthogonal polynomials to model the trajectory. We find that these model mismatches make the 'optimal estimator' to have large bias and mean squared error. For the case of undermodeling, the estimation error increases, in general, without a bound as the observation interval increases. We then propose the *Finite Lifetime Alternately Triggered Multiple Model Filter* (FLAT MMF), as a solution. FLAT MMF is a filter composed of a set of  $K$  identical conventional state estimation filters, each triggered alternately. After the last filter is triggered, the oldest one is triggered again and so on. The structure of Multiple Model Filter is used to combine these estimates optimally, in the sense of minimum mean squared error.

We find that the ratio of weightings in FLAT MMF are related to some independent non-central  $\chi^2$  random variables. Consequently, we show that the FLAT MMF can provide an estimate that follows abrupt changes in the trajectory and has the small bias for undermodeling. For the case of overmodeling or the case that the trajectory model matches to the actual motion, the estimate does not degrade significantly.

A number of simulations are conducted to illustrate the estimation performance degradation due to the model mismatches for the conventional Kalman filter and the performance improvement as the proposed FLAT MMF is used.

## Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-90-10.

**Analysis Of The Effects Of  
Model Mismatch And Flat MMF  
For Estimating Particle Motion**

**MS-CIS-90-10  
GRASP LAB 204**

**Siu-Leong Iu**

**Department of Computer and Information Science  
School of Engineering and Applied Science  
University of Pennsylvania  
Philadelphia, PA 19104-6389**

**February 1990**

**Acknowledgements:**

**This research was support in part by DARPA grant  
N00014-88-K-0632, NSF equipments grant  
CCR87-16987 and NSF grant IRI89-06770.**

# ANALYSES OF THE EFFECTS OF MODEL MISMATCH AND FLAT MMF FOR ESTIMATING PARTICLE MOTION

## *Abstract*

In this report, we analyze the performance degradation due to three classes of model mismatch: parameter jumping, undermodeling and overmodeling, in estimating the particle motion by using the orthogonal polynomials to model the trajectory. We find that these model mismatches make the 'optimal estimator' to have large bias and mean squared error. For the case of undermodeling, the estimation error increases, in general, without a bound as the observation interval increases. We then propose the *Finite Lifetime Alternately Triggered Multiple Model Filter* (FLAT MMF), as a solution. FLAT MMF is a filter composed of a set of  $K$  identical conventional state estimation filters, each triggered alternately. After the last filter is triggered, the oldest one is triggered again and so on. The structure of Multiple Model Filter is used to combine these estimates optimally, in the sense of minimum mean squared error.

We find that the ratio of weightings in FLAT MMF are related to some independent non-central  $\chi^2$  random variables. Consequently, we show that the FLAT MMF can provide an estimate that follows abrupt changes in the trajectory and has the small bias for undermodeling. For the case of overmodeling or the case that the trajectory model matches to the actual motion, the estimate does not degrade significantly.

A number of simulations are conducted to illustrate the estimation performance degradation due to the model mismatches for the conventional Kalman filter and the performance improvement as the proposed FLAT MMF is used.

## 1. INTRODUCTION

The estimation of the 3-D motion and structure of moving objects in space from video images is one of the major problems in computer vision. These estimates may be utilized in building intelligent robots, tracking moving objects and implementing autonomous navigation systems. Some of previous approaches have relied on a large number of images [Weng87, Broi86a,b, Boll85, Iu89a,b] seeking for better performance on noise filtering than two-view or three view motion analysis does [Weng87]. These approaches are all based upon the basic assumption that the motion of object is matched to the motion model which has been fixed prior to the analysis. An example would be a model which requires the object to move with a constant acceleration over the time interval of interest. Since an object can move almost arbitrarily in front of the camera, such model becomes invalid as the length of the observation time interval increases. The mismatch may cause a very large estimation error or even make the estimation process diverge. This observation motivates us to study the problem of analyzing the object trajectory under unconstrained motion in front of TV camera. We shall first analyze the performance degradation due to the model mismatch and then propose the *Finite Lifetime Alternately Triggered Multiple Model Filter* (FLAT MMF), as a new solution.

The motion analysis of a particle or an isolated target is usually formulated as a state estimation problem [Chan84]. There are three major issues involved in motion analysis: the computation speed, the nonlinearity and the representation of particle trajectory. Since the moving object must be tracked in real time, most batch approaches, such as Newton-type search algorithms, become unattractive. Recursive filters, such as the (extended) Kalman filter, are normally used in solving the estimation problem. In many cases, the plant equation for the particle motion and/or the measurement model are nonlinear. Thus, many linear estimation algorithms, such as the Kalman filter, can not be applied directly to the problem. The nonlinearity makes the estimate hard to obtain and difficult to analyze. One common solution is to linearize the nonlinear equation and then apply the linear algorithm to estimate the motion. In order to alleviate the effect of the nonlinearity which may cause the large estimation error or even make the estimation diverge, one may solve for the estimates iteratively or may incorporate the non-linear filtering technique [Mayb82].

Since we do not know how an object moves in space, we should model its trajectory before we can analyze its motion. If we assume that the object moves smoothly, its trajectory may be modeled as a power series with the specified order. Consequently, three kinds of model mismatch may be occurred. They are the model mismatches of parameter jumping, undermodeling and overmodeling, respectively. The problem of parameter jumping arises when the object changes its motion abruptly

because of the maneuver. An example is the motion of a bouncing ball. Since the order of the power series in modeling the trajectory may be less than or greater than that of the actual trajectory, the model may undermodel or overmodel the actual motion. The problem of undermodeling is more serious than that of overmodeling because in most situations we are forced to use the low-order model. In this report, we will concern on the problem of motion analysis under the above three classes of model mismatch.

Since the estimation problem of the particle motion from perspective measurements is nonlinear, there is no closed-form solution in general. In order to understand the detail of the performance degradation due to the model mismatch, in this report, we assume that the particle moves on a plane parallel to the image plane. For this kind of '2-D motion', we will find the analytic closed-form solution for the estimation and the corresponding mean squared error in case the trajectory model matches to the actual trajectory. Then we will analyze the estimation behavior for the model mismatch. For the general particle motion, a number of simulations were conducted in [Iu89b] and it was found that the estimation have the similar behavior as that of the 2-D motion. In this report, we also assume that the positional measurements are available.

The motion analysis of a target such as an airplane or a missile from radar measurements and that of a particle from video images share the same goal: estimating the object motion from the noisy measurements. An extensive literature has been available in the first area for more than two decades [Chan84] while the second problem is more recent [Iu89a]. However, we should notice some major differences between these two problems. First of all, although in both cases we do not know the exact form of the object trajectory, the trajectory of an airplane or a missile can be modeled adequately as a straight line between two maneuvers. On the other hand, the object can move almost arbitrarily in front of the camera, and can have very sharp changes, ( $180^\circ$  turns) as in the motion of a bouncing ball. Second, the relations between the measurement and the motion parameters for the radar data and for the video image are different. Third, the noise in the measurement from the optical image data is much lower than that from the radar data. Sometimes, the position of object in an optical image can be located with the subpixel accuracy. Fourth, from the radar data, we can only locate the object instead of the detailed 'look' at the object, while the high quality image of the object can be obtained even from the commercial video camera. Because of the above differences between two problems, there is no guarantee that some approaches that have been used successfully in one domain can be applied to the other.

In the rest of this section, we will review some existing approaches to the problem of model mismatch in the area of motion analysis from the radar or similar data. We will first discuss some general approaches for model mismatches and then discuss some particular approaches for the parameter jumping. Finally, we will outline the reason to seek a new algorithm in analyzing the object motion from the video images.

Three major approaches are found in the literature to compensate the effect of model mismatches in estimating the object motion. They are the approach of changing the covariance of plant noise adaptively, the finite memory filter and the fading memory filter. Since the plant noise in the Kalman filter formulation [Mayb82] is a random process representing the system model error, and since the residual process indicates how well the filter operates up to the current measurement, one can monitor the fluctuation of residual, and change the covariance of the plant noise adaptively. A number of different approaches in forming this adaptation, such as the maximum likelihood estimator, the Bayesian estimator and the covariance matching technique, have been proposed. A survey of these schemes is given in [Chin79]. Although this kind of approaches show that we can adapt the noise covariance systematically, the extremely high computational burden makes them to be less attractive for solving the model mismatches in real time [Chan84]. Furthermore, it has been observed that this approach was not fast enough to compensate the model error due to the maneuver, i.e., the estimation process takes a long time to converge or it may even diverge if another maneuvers occur before it converges [Mayb81].

The finite memory filter [Jazw70] and the fading memory filter [Gelb74] are motivated by the observation that many estimators are of 'growing memory' type, i.e., the estimate is based on the entire past history. The evolution of a system may be adequately described by some model over a short time interval, but the model may be increasingly inaccurate for the longer interval. For example, in our motion problem, the constant acceleration model may describe the trajectory quite well locally, but it is obvious that this model is invalid for general trajectory over a large interval. Thus, one may seek for an estimator in which the old measurement becomes less important as time increases.

The basic structure of the finite memory filter is to use two Kalman filters in which their estimates are based upon all measurements up to current time and up to  $N$  samples before the current time, respectively. These two estimates are then weighted by the inverse of their covariance matrices, and their difference is multiplied by some covariance matrix to yield the final estimate. This estimate can be interpreted as the optimal estimate using only the past  $N$  measurements. The drawback of the finite memory filter is that the computation burden is much higher than that of the conventional Kalman filter, and that  $N$  measurements must be stored.

The fading memory filter weighs the measurement in an exponential manner in time, the most recent sample having the largest weight. The filter turns out quite simple, with its covariance being the covariance of the normal Kalman filter multiplied by a scalar quantity which amounts to the argument of the exponential weighting. The drawback of this filter is that the relative importance of measurement is fixed by the exponential-weight model and the weighting argument. Also, the estimation performance will degrade if the model is actually matched to the true system [Simm86].

For the motion with maneuver, one common approach to modeling the actual trajectory is to augment the original state vector with the dynamics of maneuver. If the maneuver dynamics is totally unknown, one may model the acceleration of the maneuver as a Wiener process or may adopt Singer's model [Sing70]. Since we do not know when the maneuver occurs, we must keep the augmented state vector throughout the entire track. Obviously, the filter performance degrades when the motion is non-maneuvering. The above discussion motivates the approach of maneuver detection.

The basic idea of maneuver detection is to pose two hypotheses which indicate whether a maneuver has occurred or not. The detection is performed by using the generalized maximum likelihood ratio test. Before the maneuver is detected, one may track the object by using the state vector without maneuver. Then the filter may be restarted with the augmented state vector at the time the maneuver is detected. There are two alternatives in restarting the filter. One is to reprocess a large batch of past measurements and the other is just to initialize the filter. The former needs extra storage and processing time while the latter may have a large estimate error due to the improper filter initialization. Although this kind of detection-directed approach may avoid the performance degradation of the augmented state vector, there are some drawbacks of this approach. There is the detection delay, and large transient errors may occur during the state vector is switched [Chan84]. These drawbacks may be avoided if we use the multiple model filter (MMF).

The original idea of MMF is due to Magill [Magi65]. The basic idea of MMF is to construct a number of filters each of which is based upon a different system model. The final state estimate is given as a linear combination of the estimates from these filters. The weights in this combination are changed adaptively according to the residual process of filters. There are some alternatives to applying the MMF to the motion analysis. One approach is to construct two filters which use the model with and without maneuver, respectively. Since there are two hypotheses at each sampling instant, the total number of hypotheses grows exponentially with time. Assumption that the event of the occurrence of a maneuver is a Markov process may be used to simplify the computation. Another approach is to use several hypotheses to model different maneuvers. A large number of filters are required for fine



discretized maneuver models.

As we mentioned earlier the object under consideration can move almost arbitrarily in front of the camera. Consequently, motion estimators should have the ability to track trajectories with very sharp change. Due to the real-time constraint, the degree of the power series in the trajectory model must be kept low. This is especially true for the case of tracking an object in three-dimensional space from the projective positional measurement, because the size of the required state vector equals to  $3(d_m + 1)$  at least, where  $d_m$  is the degree of the power series in the model. An estimator is 'good' if it provides an estimate that follows abrupt changes in trajectory and suffers only a reasonably small bias in the event of undermodeling. Moreover, the performance of the estimator should not degrade significantly for the case of overmodeling or the case that the actual trajectory exactly matches the model. Furthermore, due to the limited resources, the use of a large number of sub-filters as in the MMF discussed before is precluded, even though these sub-filters can be operated simultaneously. (Note that, although the MMF in [Mayb81] only consists of four sub-filters, all these sub-filters have been tuned to some specific values in order to match a certain type of particular maneuver). Thus, based upon the requirements discussed above, none of the approaches reviewed so far are satisfactory. This motivates us to seek for a new filter in order to meet these requirements.

The rest of this report is organized as follows. Section 2 gives the problem statement of motion analysis. Section 3 develops the trajectory model in terms of power series. Since the mathematical analysis of the filter performance is formidable when the degree of the power series is greater than three, an alternative trajectory model which uses a basis of orthogonal polynomial, is used. We will show that the optimal estimates and their corresponding covariances of these two models are identical. We note that the optimal estimates with the orthogonal polynomial and that from Kalman filter are the same. Section 4 discusses three classes of model mismatch: parameter jumping, undermodeling and overmodeling. The biases and mean squared errors of the 'optimal estimate' for these three mismatches are derived. Section 5 introduces the FLAT MMF in order to solve the problems of model mismatch and an analysis of the FLAT MMF for the model mismatches are discussed in detail. Some simulations for sharp maneuvers and circular trajectories are presented to demonstrate the performance of the FLAT MMF. Section 6 concludes the paper with discussions and some final comments.

## 2. PROBLEM STATEMENT

For the 2-D motion discussed in the introduction, since the depth does not change and the motion parameters related to the X and Y coordinates are decoupled, without loss of generality, the motion analysis can be reduced to the following 1-D motion problem.

Let  $m(t)$  be the positional measurement at time  $t$  of the trajectory of a moving particle in one dimension. Assume that the actual trajectory  $X_s(t)$  is piecewise differentiable up to  $d_s$ -th order. The subscript 's' denotes the actual signal (trajectory). Let  $X_s^{[m]}(t)$  be the  $m$ -th derivative of  $X_s(t)$ . Let  $\Delta$  be the time separation between measurement samples. Suppose that  $(J - J_0)$  total samples of  $m(t)$  at time  $t_j = j \Delta$ ,  $j = J_0, \dots, J - 1$ , are available, and we model them as

$$m(t_j) = X_s(t_j) + n(t_j) , \quad (2.1)$$

where  $n(t_j)$  is a discrete-time random process describing the noise in the measurements. Throughout this report, the  $\{ n(t_j) \}_{j=0}^{J-1}$  is assumed to be white zero mean Gaussian random sequence with common variance  $\sigma^2$ . Let  $\hat{X}_{m, J_0, J-1}(t)$  be the estimates of  $X_s^{[m]}(t)$ ,  $m = 0, \dots, d_s$  and define the error correlations as

$$R_{mm, J_0, J-1}(t) = E \{ [ \hat{X}_{m, J_0, J-1}(t) - X_s^{[m]}(t) ] [ \hat{X}_{n, J_0, J-1}(t) - X_s^{[n]}(t) ] \} . \quad (2.2)$$

The subscript 'm,  $J_0, J - 1$ ' of  $\hat{X}_{m, J_0, J-1}(t)$  denotes that we use the measurements  $\{ m(t_j) \}_{j=J_0}^{J-1}$  to find the estimates of  $X_s^{[m]}(t)$ . The auto-correlations  $R_{mm, J_0, J-1}(t)$  are called mean squared error of the estimates  $\hat{X}_{m, J_0, J-1}(t)$ . Then the goal is to find the estimates  $\hat{X}_{m, J_0, J-1}(t)$ ,  $m = 0, \dots, d_s$ , optimally, in the sense of the minimum mean squared error (MMSE), by using all the available measurements.

In the numerical analysis context, the above problem is called data (curve) fitting, and the objective is to find the function itself and its derivatives from noisy measurements. According to the filtering terminology, the problem is named as filtering, smoothing, forward prediction and backward prediction in case of  $t = t_{j-1}$ ,  $t_{j_0} \leq t < t_{j-1}$ ,  $t > t_{j-1}$ , and  $t < t_{j_0}$ , respectively. In this report, we will focus our discussion on filtering and on one-step forward prediction, i.e., the estimation at  $t = t_{j-1}$  and  $t = t_j$ , respectively.

## 3. ORTHOGONAL POLYNOMIAL FORMULATION AND KALMAN FILTER SOLUTION

In this section, we model the actual trajectory as power series and orthogonal polynomials. Based upon these two models, their optimal estimates of  $X_s^{[m]}(t)$  and the corresponding covariances will be found. We will show that these optimal estimates are identical and will discuss the advantages of

using the orthogonal polynomial over the power series. Then we will formulate the motion estimation as the state estimation problem and will summarize the Kalman filter solution. We observe that the optimal estimates of  $X_s^{[ml]}(t)$  in terms of orthogonal polynomial and the estimates from Kalman filter are identical.

In general, we do not know the exact form of actual trajectory  $X_s(t)$ . However, if we assume that the particle moves smoothly, we may model the  $X_s(t)$  as a finite order power series;

$$X(t) = \sum_{p=0}^{d_m} \theta'_{p, J_0} \frac{(t - t_{J_0})^p}{p!}, \quad (3.1)$$

where  $d_m$  stands for the degree (order) of trajectory model  $X(t)$ , and the parameters  $\theta'_{p, J_0}$ ,  $p = 0, \dots, d_m$ , denote the  $p$ -th derivatives of  $X(t)$  at time  $t_{J_0}$ , respectively. The symbol ' is used to distinguish from the parameters of using orthogonal polynomials which will be introduced shortly. Let  $\underline{\theta}'$  be the vector of these unknown motion parameters defined as

$$\underline{\theta}' = [\theta'_{0, J_0} \quad \theta'_{1, J_0} \cdots \theta'_{d_m, J_0}]^T, \quad (3.2)$$

where the symbol 'T' denotes the transpose. Define the composite vector which comprises the measurement  $m(t)$  from  $t_i$  to  $t_j$ , and denote it as  $\underline{M}_{i, j}$ , such that

$$\underline{M}_{i, j} = [m(t_i) \quad m(t_{i+1}) \cdots m(t_j)]^T. \quad (3.3)$$

If the trajectory model  $X(t)$  in (3.1) matches to the actual trajectory  $X_s(t)$ , i.e.,  $X_s(t) = X(t)$ , then the joint probability density function  $p(\underline{M}_{J_0, J-1}; \underline{\theta}')$  of the measurement vector  $\underline{M}_{J_0, J-1}$  can be represented as

$$p(\underline{M}_{J_0, J-1}; \underline{\theta}') = \frac{1}{(2\pi)^{(J-J_0)/2} \sigma^{J-J_0}} \exp\left\{-\frac{1}{2\sigma^2} \sum_{j=J_0}^{J-1} [m(t_j) - X(t_j)]^2\right\}. \quad (3.4)$$

Also, from (3.1),

$$X_s^{[ml]}(t) = \sum_{p=m}^{d_m} \theta'_{p, J_0} \frac{(t - t_{J_0})^{p-m}}{(p-m)!}, \quad m = 0, 1, \dots, d_m. \quad (3.5)$$

Thus,  $X_s^{[ml]}(t_{J_0}) = \theta'_{p, J_0}$ . We may be tempted to find the unknown parameters  $\theta'_{p, J_0}$  from the measurements and then use (3.5) to obtain the estimates of  $X_s^{[ml]}(t)$ . However, we will see that the estimates of  $X_s^{[ml]}(t)$  can be found directly from the measurements. The unknown parameters  $\theta'_{p, J_0}$  are useful for describing the trajectories  $X_s(t)$  and  $X(t)$  as well as for deriving the estimators.

The minimum variance unbiased (MVUB) estimates  $\hat{X}'_{m, J_0, J-1}(t)$  of  $X_s^{[m]}(t)$ ,  $m = 0, 1, \dots, d_m$ , are given by [Peeb70] as

$$\hat{X}'_{m, J_0, J-1}(t) = X_s^{[m]}(t) + \sum_{p=0}^{d_m} \sum_{q=0}^{d_m} \frac{\partial X_s^{[m]}(t)}{\partial \theta'_{p, J_0}} I'_{pq} \frac{\partial \ln p ( \underline{M}_{J_0, J-1} ; \underline{\theta}' )}{\partial \theta'_{q, J_0}}, \quad (3.6)$$

where 'ln' stands for natural logarithm and  $I'_{pq}$  denotes the  $pq$ -th element of the inverse of the Fisher information matrix  $[I']$  having elements

$$I'_{pq} = E \left[ \frac{\partial \ln p ( \underline{M}_{J_0, J-1} ; \underline{\theta}' )}{\partial \theta'_{p, J_0}} \frac{\partial \ln p ( \underline{M}_{J_0, J-1} ; \underline{\theta}' )}{\partial \theta'_{q, J_0}} \right]. \quad (3.7)$$

Note that the first term in the right hand side of (3.6),  $X_s^{[m]}(t)$ , will be canceled out by a part of the second term. Also, the estimates in (3.6) are equal to the MMSE estimates [Mayb82], because of the earlier assumption of Gaussian noise in the measurement. The covariances of the estimates  $\hat{X}'_{m, J_0, J-1}(t)$  and  $\hat{X}'_{n, J_0, J-1}(t)$ , for  $m, n = 0, 1, \dots, d_m$ , are given by

$$\begin{aligned} V'_{mn, J_0, J-1}(t) &= E \{ [ \hat{X}'_{m, J_0, J-1}(t) - E ( \hat{X}'_{m, J_0, J-1}(t) ) ] [ \hat{X}'_{n, J_0, J-1}(t) - E ( \hat{X}'_{n, J_0, J-1}(t) ) ] \} \\ &= \sum_{p=0}^{d_m} \sum_{q=0}^{d_m} \frac{\partial X_s^{[m]}(t)}{\partial \theta'_{p, J_0}} \frac{\partial X_s^{[n]}(t)}{\partial \theta'_{q, J_0}} I'_{pq}. \end{aligned} \quad (3.8)$$

Using (3.4), (3.1) and (3.5), we obtain

$$\hat{X}'_{m, J_0, J-1}(t) = \sum_{j=J_0}^{J-1} w'_{mj, J_0, J-1}(t) m(t_j), \quad (3.9)$$

where

$$w'_{mj, J_0, J-1}(t) = \frac{1}{\sigma^2} \sum_{p=m}^{d_m} \sum_{q=0}^{d_m} \frac{(t - t_{J_0})^{p-m}}{(p-m)!} I'_{pq} \frac{(j - J_0)^q \Delta^q}{q!}, \quad (3.10)$$

$$V'_{mn, J_0, J-1}(t) = \sum_{p=m}^{d_m} \sum_{q=n}^{d_m} \frac{(t - t_{J_0})^{p-m}}{(p-m)!} \frac{(t - t_{J_0})^{q-n}}{(q-n)!} I'_{pq}. \quad (3.11)$$

Although we could find the element of the  $(d_m + 1) \times (d_m + 1)$  matrix  $[I']$  from

$$I'_{pq} = \frac{\Delta^{p+q}}{\sigma^2 p! q!} \sum_{j=J_0}^{J-1} (j - J_0)^{p+q}, \quad (3.12)$$

it is not feasible to derive the closed-form analytical solution for the estimators and the corresponding covariances in (3.9) and (3.11) when  $d_m \geq 3$  because we need invert the matrix  $[I']$  to obtain  $I'_{pq}$ .

However, this problem can be solved if we represent the trajectory model  $X(t)$  in terms of orthogonal polynomials.

Let  $\{ \xi_{p, N}(u) \}_{p=0}^{N-1}$  be a set of orthogonal polynomials with respect to the discrete points  $u = 1, 2, \dots, N$ . The function  $\xi_{p, N}(u)$  is a  $p$ -degree polynomial and its factorial representation is given by [Rals65] as

$$\xi_{p, N}(u) = d_{0, p, N} + \sum_{k=1}^p d_{k, p, N} (u-1) (u-2) \cdots (u-k), \quad (3.13)$$

where

$$d_{k, p, N} = \frac{(-1)^{p+k} (p+k)! (N-k-1)! (p!)^2}{(2p)! (p-k)! (N-p-1)! (k!)^2}. \quad (3.14)$$

Note that the leading coefficient of  $\xi_{p, N}(u)$  is equal to one, i.e.,  $d_{p, p, N} = 1$ . The first four of these polynomials can be written as follows.

$$\begin{aligned} \xi_{0, N}(u) &= 1, \\ \xi_{1, N}(u) &= (u - \bar{u}), \quad \bar{u} = (N+1) / 2, \\ \xi_{2, N}(u) &= (u - \bar{u})^2 - (N^2 - 1) / 12, \\ \xi_{3, N}(u) &= (u - \bar{u})^3 - (u - \bar{u}) (3N^2 - 7) / 20. \end{aligned} \quad (3.15)$$

The orthogonal polynomials satisfy the recursion relationship

$$\xi_{p+1, N}(u) = \xi_{1, N}(u) \xi_{p, N}(u) - \frac{p^2 (N^2 - p^2)}{4 (4p^2 - 1)} \xi_{p-1, N}(u), \quad (3.16)$$

and the orthogonality property

$$\sum_{u=1}^N \xi_{p, N}(u) \xi_{q, N}(u) = S(p, N) \delta_{p, q}, \quad (3.17)$$

where

$$S(p, N) = \frac{(p!)^4 \prod_{r=-p}^p (N-r)}{(2p)! (2p+1)!}, \quad (3.18)$$

and  $\delta_{p, q}$  is the Kronecker delta. The following three theorems will play the integral role for the further discussion. Their proofs are given in Appendix.

**Theorem 1**

$$(i) \quad \xi_{p, N(N+1)} = \begin{cases} 1 & p = 0 \\ \frac{(p!)^2}{(2p)!} \prod_{r=1}^p (N+r) & 0 < p \leq N \end{cases} \quad (3.19)$$

$$(ii) \quad \xi_{p, N(N)} = \begin{cases} 1 & p = 0 \\ \frac{(p!)^2}{(2p)!} \prod_{r=1}^p (N-r) & 0 < p \leq N \end{cases} \quad (3.20)$$

**Theorem 2**

The  $p$ -th derivative of  $\xi_{p, N}(u)$  is given as

$$\xi_{p, N}^{[p]}(u) = p! \quad p \leq N. \quad (3.21)$$

For large  $u$  and  $m < p \leq N$ , the  $m$ -th derivatives of  $\xi_{p, N}(u)$  can be approximated as

$$\xi_{p, N}^{[m]}(u) \approx \frac{p!}{(p-m)!} u^{p-m}. \quad (3.22)$$

**Theorem 3**

Let

$$V(p, N, u, m) = \frac{[\xi_{p, N}^{[m]}(u)]^2}{S(p, N)}, \quad (3.23)$$

then the followings are true.

$$(i) \quad V(p, N, N+1, 0) = \begin{cases} 1/N & p = 0 \\ \frac{2p+1}{N} \prod_{r=1}^p \frac{N+r}{N-r} & 0 < p \leq N \end{cases} \quad (3.24)$$

$$(ii) \quad V(p, N, N, 0) = \begin{cases} 1/N & p = 0 \\ \frac{2p+1}{N} \prod_{r=1}^p \frac{N-r}{N+r} & 0 < p \leq N \end{cases} \quad (3.25)$$

(iii)  $V(p, N, N+1, m) = 0$ , if  $p < m$ . Furthermore, for  $m \leq p \leq N$  and large  $N$ ,

$$V(p, N, N+1, m) \approx \frac{(2p)! (2p+1)!}{(p!)^2 ((p-m)!)^2} (N+1)^{-(2m+1)}. \quad (3.26)$$

(iv)  $V(p, N, N, m) = 0$ , if  $p < m$ . Furthermore, for  $m \leq p \leq N$  and large  $N$ ,

$$V(p, N, N, m) \approx \frac{(2p)! (2p+1)!}{(p!)^2 ((p-m)!)^2} N^{-(2m+1)}. \quad (3.27)$$

□

We can express the trajectory model  $X(t)$  alternatively in terms of orthogonal polynomial as follows.

$$X(t) = \sum_{p=0}^{d_m} \theta_{p, J_0, J-1} \xi_{p, J-J_0}(u_{J_0}(t)), \quad (3.28)$$

where the argument of  $\xi(\cdot)$  is the normalized and shifted function

$$u_{J_0}(t) = t / \Delta - J_0 + 1. \quad (3.29)$$

The  $m$ -th derivatives of  $X(t)$  are given by

$$X^{[m]}(t) = \frac{1}{\Delta^m} \sum_{p=m}^{d_m} \theta_{p, J_0, J-1} \xi_{p, J-J_0}^{[m]}(u_{J_0}(t)), \quad (3.30)$$

where  $\xi_{p, J-J_0}^{[m]}(u_{J_0}(t))$  stands for the  $m$ -th derivatives of  $\xi_{p, J-J_0}(u)$  with respect to  $u$  at  $u = u_{J_0}(t)$ . Similarly to (3.6) and (3.8), the MVUB (and the MMSE) estimator  $\hat{X}_{m, J_0, J-1}(t)$  of  $X_s^{[m]}(t)$  and the corresponding covariances are given by

$$\hat{X}_{m, J_0, J-1}(t) = X_s^{[m]}(t) + \sum_{p=0}^{d_m} \sum_{q=0}^{d_m} \frac{\partial X_s^{[m]}(t)}{\partial \theta_{p, J_0, J-1}} I^{pq} \frac{\partial \ln p(\underline{M}_{J_0, J-1}; \underline{\theta})}{\partial \theta_{q, J_0, J-1}}, \quad (3.31)$$

$$V_{mn, J_0, J-1}(t) = \sum_{p=0}^{d_m} \sum_{q=0}^{d_m} \frac{\partial X_s^{[m]}(t)}{\partial \theta_{p, J_0, J-1}} \frac{\partial X_s^{[n]}(t)}{\partial \theta_{q, J_0, J-1}} I^{pq}, \quad (3.32)$$

where  $p(\underline{M}_{J_0, J-1}; \underline{\theta})$  is the joint probability density function of the measurement vector  $\underline{M}_{J_0, J-1}$  in terms of the unknown parameter vector

$$\underline{\theta} = [\theta_{0, J_0, J-1} \quad \theta_{1, J_0, J-1} \quad \cdots \quad \theta_{d_m, J_0, J-1}]^T. \quad (3.33)$$

Note that there is difference between  $V_{mn, J_0, J-1}(t)$  and  $R_{mn, J_0, J-1}(t)$ . The former is the covariance of the estimate  $\hat{X}_{m, J_0, J-1}(t)$  and the latter is the correlation between the errors  $[\hat{X}_{m, J_0, J-1}(t) - X_s^{[m]}(t)]$  and  $[\hat{X}_{n, J_0, J-1}(t) - X_s^{[n]}(t)]$ . The  $I^{pq}$  is the  $pq$ -th element in the inverse of the information matrix  $[I]$  having elements

$$I^{pq} = E \left[ \frac{\partial \ln p(\underline{M}_{J_0, J-1}; \underline{\theta})}{\partial \theta_{p, J_0, J-1}} \frac{\partial \ln p(\underline{M}_{J_0, J-1}; \underline{\theta})}{\partial \theta_{q, J_0, J-1}} \right]. \quad (3.34)$$

By using (3.30) and the orthogonality property in (3.17), it can be shown that [Peeb70]

$$\hat{X}_{m, J_0, J-1}(t) = \sum_{j=J_0}^{J-1} w_{mj, J_0, J-1}(t) m(t_j), \quad (3.35)$$

where

$$w_{mj, J_0, J-1}(t) = \frac{1}{\Delta^m} \sum_{p=m}^{d_m} \frac{\xi_{p, J-J_0}^{[m]}(u_{J_0}(t)) \xi_{p, J-J_0}(j - J_0 + 1)}{S(p, J - J_0)}, \quad (3.36)$$

$$V_{mn, J_0, J-1}(t) = \frac{\sigma^2}{\Delta^{m+n}} \sum_{p=m}^{d_m} \frac{\xi_{p, J-J_0}^{[m]}(u_{J_0}(t)) \xi_{p, J-J_0}^{[n]}(u_{J_0}(t))}{S(p, J - J_0)}. \quad (3.37)$$

Note that if  $X(t) = X_s(t)$ , then  $X^{[m]}(t) = X_s^{[m]}(t)$ , and we can verify that the estimates  $\hat{X}_{m, J_0, J-1}(t)$  of  $X_s^{[m]}$  are unbiased by using (2.1), (3.28) and (3.17);

$$\begin{aligned} E [ \hat{X}_{m, J_0, J-1}(t) ] &= E [ \sum_{j=J_0}^{J-1} w_{mj, J_0, J-1}(t) m(t_j) ] \\ &= \sum_{j=J_0}^{J-1} [ \frac{1}{\Delta^m} \sum_{p=m}^{d_m} \frac{\xi_{p, J-J_0}^{[m]}(u_{J_0}(t)) \xi_{p, J-J_0}(j - J_0 + 1)}{S(p, J - J_0)} ] [ \sum_{q=0}^{d_m} \theta_{sq, J_0, J-1} \xi_{q, J-J_0}(j - J_0 + 1) ] \\ &= \frac{1}{\Delta^m} \sum_{p=m}^{d_m} \sum_{q=0}^{d_m} \frac{\xi_{p, J-J_0}^{[m]}(u_{J_0}(t)) \theta_{sp, J_0, J-1}}{S(p, J - J_0)} S(q, J - J_0) \delta_{p, q} \\ &= \frac{1}{\Delta^m} \sum_{p=m}^{d_m} \theta_{sp, J_0, J-1} \xi_{p, J-J_0}^{[m]}(u_{J_0}(t)) \\ &= X_s^{[m]}(t). \end{aligned} \quad (3.38)$$

The relation between the estimates obtained from the above two models (one in terms of power series and the other in terms of orthogonal polynomials) is given in the following theorem.

**Theorem 4**

Two estimators,  $\hat{X}'_{m, J_0, J-1}(t)$  in (3.6) and  $\hat{X}_{m, J_0, J-1}(t)$  in (3.31), are identical. Their corresponding covariances in (3.8) and (3.32) are also the same.

□

The proof of this theorem is given in Appendix. Because the estimators  $\hat{X}_{m, J_0, J-1}(t)$  and their covariances in terms of the orthogonal polynomials have the explicit expressions (3.35) and (3.37) even for



$d_m > 3$ , and because, as we shall see, the orthogonality property in (3.17) is very useful for analyzing the estimators in case of model mismatch, we shall use the orthogonal polynomial instead of the conventional power series in describing the actual and model trajectories.

The second half of this section is to formulate the problem of motion estimation as the state estimation problem. Let the state vector be

$$\underline{s}(t) = [ X_s^{[0]}(t) \ X_s^{[1]}(t) \ \cdots \ X_s^{[d_m]}(t) ]^T . \quad (3.39)$$

The evolution of state and the discrete measurement satisfy the following plant equation and measurement equation,

$$\frac{d}{dt} \underline{s}(t) = A \underline{s}(t) , \quad (3.40)$$

$$m(t_j) = H \underline{s}(t_j) + n(t_j), \quad j = J_0, 1, \dots, J-1 , \quad (3.41)$$

where  $A$  is a  $(d_m + 1) \times (d_m + 1)$  matrix with elements

$$A_{i,j} = \begin{cases} 1 & \text{if } j = i + 1 \\ 0 & \text{else} \end{cases} , \quad (3.42)$$

and the  $1 \times (d_m + 1)$  matrix  $H$  is given by

$$H = [ 1 \ 0 \ 0 \ \cdots \ 0 ] . \quad (3.43)$$

Then the state estimation problem is to estimate the state  $\underline{s}(t)$  at time  $t_{j-1}$  based upon all measurements  $\{ m(t_j) \}_{j=J_0}^{J-1}$ .

It is well known that the optimal solution of the above state estimation problem, in the sense of MMSE, is given by the Kalman filter. Let  $\hat{\underline{s}}(t_i^- ; J_0)$  and  $\hat{\underline{s}}(t_i^+ ; J_0)$  be the optimal estimate of state  $\underline{s}(t)$  at time  $t_i$ , given  $\{ m(t_j) \}_{j=J_0}^{i-1}$  and  $\{ m(t_j) \}_{j=J_0}^i$ , respectively. Let the  $(d_m + 1) \times (d_m + 1)$  matrices  $P(t_i^- ; J_0)$  and  $P(t_i^+ ; J_0)$  be the covariances of  $\hat{\underline{s}}(t_i^- ; J_0)$  and  $\hat{\underline{s}}(t_i^+ ; J_0)$ , respectively. Then the optimal state estimates  $\hat{\underline{s}}(t_i^+ ; J_0)$  and the covariances  $P(t_i^+ ; J_0)$ ,  $i = J_0, 1, \dots, J-1$ , can be obtained recursively as follows [Mayb82]. The state estimate  $\hat{\underline{s}}(t_i^- ; J_0)$  and the covariance  $P(t_i^- ; J_0)$  are propagated from measurement time  $t_{j-1}$  to  $t_i$  by the relations

$$\hat{\underline{s}}(t_i^- ; J_0) = \Phi(t_i, t_{i-1}) \hat{\underline{s}}(t_{i-1}^+ ; J_0) , \quad (3.44)$$

$$P(t_i^- ; J_0) = \Phi(t_i, t_{i-1}) P(t_{i-1}^+ ; J_0) \Phi^T(t_i, t_{i-1}) , \quad (3.45)$$

where  $\Phi(t_i, t_{i-1})$  is the state transition matrix of (3.40). Since the matrix  $A$  in (3.40) is time invariant with elements  $A_{ij}$  given in (3.42), it can be shown that

$$\Phi(t_i, t_{i-1}) = e^{A \Delta} = \sum_{n=0}^{d_m} \frac{A^n \Delta^n}{n!}, \quad (3.46)$$

and the pq-th elements of  $\Phi(t_i, t_{i-1})$  are given by

$$\left[ \Phi(t_i, t_{i-1}) \right]_{pq} = \begin{cases} \frac{\Delta^n}{n!} & p = q - n \\ 0 & p > q \end{cases}, \quad (3.47)$$

for  $n = 0, \dots, q$  and  $p, q = 0, \dots, d_m$ . The estimate and the covariance are then updated by the following relations, using the newly measurement  $m(t_i)$ .

$$K(t_i) = P(t_{i-1}^-; J_0) H^T [ H P(t_{i-1}^-; J_0) H^T + \sigma^2 ]^{-1}, \quad (3.48)$$

$$\hat{\underline{s}}(t_i^+; J_0) = \hat{\underline{s}}(t_i^-; J_0) + K(t_i) [ m(t_i) - H \hat{\underline{s}}(t_i^-; J_0) ], \quad (3.49)$$

$$P(t_i^+; J_0) = P(t_i^-; J_0) - K(t_i) H P(t_i^-; J_0), \quad (3.50)$$

where  $K(t)$  is called the Kalman gain at time  $t$ . The initial condition for the recursion is given by

$$\hat{\underline{s}}(t_{J_0-1}^+; J_0) = E [ \underline{s}(t_{J_0-1}) ] = \hat{\underline{s}}_{J_0-1}, \quad (3.51)$$

$$P(t_{J_0-1}^+; J_0) = E \{ [ \underline{s}(t_{J_0-1}) - \hat{\underline{s}}_{J_0-1} ] [ \underline{s}(t_{J_0-1}) - \hat{\underline{s}}_{J_0-1} ]^T \} = P_{J_0-1}. \quad (3.52)$$

$\hat{\underline{s}}_{J_0-1}$  and  $P_{J_0-1}$  contain the prior information of the state before the recursion starts. One may set them to be a zero vector and a diagonal matrix with very large diagonal value, respectively, if apriori information is not available. Or, one may initialize the state vector and covariance matrix from the measurements directly and then start the recursion. The second approach will be discussed in a more detail later.

Since the state estimation problem discussed above is exactly equivalent to the estimation problem in terms of power series which we had discussed in the beginning of this section, the optimal solution of these two problems are the same. Using theorem 4, we have

$$\hat{\underline{s}}(t_i^-; J_0) = [ \hat{X}_{0, J_0, i-1}(t_i) \quad \hat{X}_{1, J_0, i-1}(t_i) \quad \cdots \quad \hat{X}_{d_m, J_0, i-1}(t_i) ]^T \quad (3.53)$$

$$\hat{\underline{s}}(t_i^+; J_0) = [ \hat{X}_{0, J_0, i}(t_i) \quad \hat{X}_{1, J_0, i}(t_i) \quad \cdots \quad \hat{X}_{d_m, J_0, i}(t_i) ]^T \quad (3.54)$$

$$P_{mn}(t_i^-; J_0) = V_{mn, J_0, i-1}(t_i) \quad (3.55)$$

$$P_{mn}(t_i^+; J_0) = V_{mn, J_0, i}(t_i), \quad (3.56)$$

where  $P_{mn}(t_i^-; J_0)$  and  $P_{mn}(t_i^+; J_0)$  are the  $mn$ -th elements of  $P(t_i^-; J_0)$  and  $P(t_i^+; J_0)$ , respectively, for  $m, n = 0, 1, \dots, d_m$ . The above observation is important in three ways. First, the Kalman filter solution in (3.44)-(3.50) gives us a procedure for finding the estimates recursively, i.e., we do not need to store and process the entire history of the measurement every time when a new measurement comes in. Second, we can obtain the initial state estimate and its covariance by using the measurements  $\{ m(t_j) \}_{j=J_0}^{d_m+J_0}$  from (3.54) and (3.56), if we model the trajectory as a  $d_m$ -degree polynomial. Subsequent estimates can be obtained by starting the Kalman filter at time  $t_{d_m+J_0+1}$ . Third, the optimal solutions in terms of orthogonal polynomials in (3.35) and (3.37) give us explicit expressions for the estimates  $\{ \hat{X}_{m, J_0, J-1}(t) \}_{m=0}^{d_m}$  and their covariances. These expressions are very useful in analyzing the performance degradation for the model mismatch which we will discuss in next section.

#### 4. PERFORMANCE ANALYSIS UNDER MODEL MISMATCH

In this section, the performance degradation of the estimation due to three classes of model mismatch discussed in the introduction are discussed. Note that if the trajectory model  $X(t)$  in (3.28) is true, i.e.  $X_s(t) = X(t)$ , then the estimates  $\hat{X}_{m, J_0, J-1}(t)$  of  $X_s^{[m]}(t)$  in (3.35) are unbiased; the corresponding variances  $V_{mn, J_0, J-1}(t)$  in (3.37) achieve the MMSE and the error correlations  $R_{mn, J_0, J-1}(t)$  become equal to the covariances  $V_{mn, J_0, J-1}(t)$ . If the trajectory model does not match the actual trajectory, then the above properties may not be valid, i.e., the estimators  $\hat{X}_{m, J_0, J-1}(t)$  may have biases

$$B_{m, J_0, J-1}(t) = E [ \hat{X}_{m, J_0, J-1}(t) ] - X_s^{[m]}(t) \quad (4.1)$$

and the error correlations become

$$R_{mn, J_0, J-1}(t) = V_{mn, J_0, J-1}(t) + B_{m, J_0, J-1}(t) B_{n, J_0, J-1}(t) . \quad (4.2)$$

Note that the biases  $B_{m, J_0, J-1}(t)$  and the mean squared error  $R_{mn, J_0, J-1}(t)$  are two useful quantities in measuring the performance of the estimators. The former one indicates how far the estimate  $\hat{X}_{m, J_0, J-1}(t)$  in average to the true value  $X_s^{[m]}(t)$  is and the latter measures how large the error of the estimate makes in average.

##### 4.1 MODEL MISMATCH DUE TO PARAMETER JUMPING

Let us start our discussion on the parameter jumping by a simple example.

**Example 1**

Suppose we use a constant value model for  $X(t)$  to estimate a piecewise constant trajectory  $X_s(t)$  at time  $t_{J-1}$  from the noisy measurements  $\{ m(t_j) \}_{j=J_0}^{J-1}$ , i.e.,

$$X(t) = \theta_{0, J_0, J-1}, \quad (4.3)$$

$$X_s(t) = \begin{cases} \theta_{s1, J_0, J-1} & t_{J_0} \leq t < t_{sw} \\ \theta_{s2, J_0, J-1} & t_{sw} \leq t \leq t_{J-1} \end{cases}, \quad (4.4)$$

where  $t_{sw} = sw \Delta$  is the time at which the trajectory switches its value. We assume that it is at one of the discrete measurement times. Then, from (3.35), (3.37) and (3.15), the optimal estimate  $\hat{X}_{0, J_0, J-1}(t)$  of  $X^{[0]}(t) = X_s(t)$  and the corresponding variance are given by

$$\hat{X}_{0, J_0, J-1}(t) = \frac{1}{J - J_0} \sum_{j=J_0}^{J-1} m(t_j), \quad (4.5)$$

$$V_{00, J_0, J-1}(t) = \frac{\sigma^2}{J - J_0}. \quad (4.6)$$

Substituting (2.1) and (4.4) into (4.5) and taking the expectation, we have

$$\begin{aligned} E [ \hat{X}_{0, J_0, J-1}(t) ] &= \frac{1}{J - J_0} \sum_{j=J_0}^{J-1} X_s(t_j) \\ &= \frac{1}{J - J_0} \left[ (sw - J_0)\theta_{s1, J_0, J-1} + (J - sw)\theta_{s2, J_0, J-1} \right] \\ &= \theta_{s2, J_0, J-1} + \frac{sw - J_0}{J - J_0} (\theta_{s1, J_0, J-1} - \theta_{s2, J_0, J-1}). \end{aligned} \quad (4.7)$$

Thus the bias and the mean squared error of the estimate  $\hat{X}_{0, J_0, J-1}(t)$  at time  $t_{J-1}$  are given by

$$B_{0, J_0, J-1}(t_{J-1}) = \frac{sw - J_0}{J - J_0} (\theta_{s1, J_0, J-1} - \theta_{s2, J_0, J-1}), \quad (4.8)$$

$$R_{00, J_0, J-1}(t_{J-1}) = \frac{\sigma^2}{J - J_0} + \frac{(sw - J_0)^2}{(J - J_0)^2} (\theta_{s1, J_0, J-1} - \theta_{s2, J_0, J-1})^2. \quad (4.9)$$

Figure 1 shows the mean of the estimate and the original trajectory for  $\theta_{s1, J_0, J-1} < \theta_{s2, J_0, J-1}$ . The bias in (4.8) is proportional to the difference of  $\theta_{s1, J_0, J-1}$  and  $\theta_{s2, J_0, J-1}$ , and the value of  $\frac{sw - J_0}{J - J_0}$ . Although this bias will decrease as  $J$  increases, a large number of  $J$  is required to suppress it if  $sw$  is large. For

example, if we want the bias  $B_{0, J_0, J-1}(t_{J-1}) = (\theta_{s1, J_0, J-1} - \theta_{s2, J_0, J-1}) / 2$ , then  $(J - J_0) = 2(sw - J_0)$ . It means that, after the trajectory switches its value from  $\theta_{s1, J_0, J-1}$  to  $\theta_{s2, J_0, J-1}$  at time  $(sw - J_0)\Delta$ , we need another total  $(sw - J_0)$  samples to get an estimate whose value, in average, equals to the average of  $\theta_{s1, J_0, J-1}$  and  $\theta_{s2, J_0, J-1}$ . The physical interpretation of this bias is that the filter based upon the proposed model has memorized all the past 'invalid' measurements. Therefore a large number of new measurements are required in order to nullify these invalid values. This is the reason that the estimate is so bad after the trajectory jumps to a new value and that the estimate takes a long time to converge to a reasonable value.  $\square$

For the general case, we may describe the actual trajectory for which the particle changes its motion abruptly at time  $t_{sw}$  as

$$X_s(t) = \begin{cases} X_{s1}(t) = \sum_{p=0}^{d_s} \theta_{s1p, J_0, J-1} \xi_{p, J-J_0}(u_{J_0}(t)) & t_{J_0} \leq t < t_{sw} \\ X_{s2}(t) = \sum_{p=0}^{d_s} \theta_{s2p, J_0, J-1} \xi_{p, J-J_0}(u_{J_0}(t)) & t_{sw} \leq t < t_{J-1} \end{cases} \quad (4.10)$$

If we use the  $d_m$ -degree polynomial  $X(t)$  in (3.28) to model  $X_s(t)$  and assume  $d_s = d_m$ , then the mean of the estimates  $\hat{X}_{m, J_0, J-1}(t)$  of  $X_s^{[m]}(t)$  are given by

$$\begin{aligned} E [ \hat{X}_{m, J_0, J-1}(t) ] &= E [ \sum_{j=J_0}^{J-1} w_{mj, J_0, J-1}(t) m(t_j) ] \\ &= \sum_{j=J_0}^{[sw]-1} w_{mj, J_0, J-1}(t) X_{s1}(t_j) + \sum_{j=[sw]}^{J-1} w_{mj, J_0, J-1}(t) X_{s2}(t_j) \\ &= \sum_{j=J_0}^{J-1} w_{mj, J_0, J-1}(t) X_{s2}(t_j) + \sum_{j=J_0}^{[sw]-1} w_{mj, J_0, J-1}(t) [ X_{s1}(t_j) - X_{s2}(t_j) ], \end{aligned} \quad (4.11)$$

where  $[sw]$  is the smallest integer which is greater than or equal to  $sw$ . Similarly to the derivation of (3.38), one may show that the first term in the right hand side of (4.11) is equal to  $X_s^{[m]}(t)$  if  $t \geq t_{sw}$ . So the biases of the estimates at time  $t_{J-1}$  are:

$$B_{m, J_0, J-1}(t_{J-1}) = \sum_{j=J_0}^{[sw]-1} w_{mj, J_0, J-1}(t_{J-1}) [ X_{s1}(t_j) - X_{s2}(t_j) ], \quad (4.12)$$

and the error correlations of the estimates  $R_{mn}(t_{J-1})$  are given in (4.2) with the above  $B_{m, J_0, J-1}(t_{J-1})$ .

More generally, if the trajectory switches  $L$  times within the time interval  $[ t_{J_0}, t_{J-1} ]$  and is described by

$$X_s(t) = \begin{cases} X_{sk}(t) & \text{if } t \in [t_{sw_{(k-1)}}, t_{sw_k}) \\ X_{s(L+1)}(t) & \text{if } t \in [t_{sw_L}, t_{sw_{(L+1)}}] \end{cases} \quad \text{for } k = 1, 2, \dots, L \quad (4.13)$$

where  $t_{sw_k}$  indicates the  $k$ -th switch time,  $t_{sw_0} = t_{J_0}$  and  $t_{sw_{(L+1)}} = t_{J-1}$ , then it can be shown that the biases of the estimates  $\hat{X}_{m, J_0, J-1}(t)$  at time  $t_{J-1}$  are given by

$$B_{m, J_0, J-1}(t_{J-1}) = \sum_{k=1}^L \sum_{j=\lceil sw_{(k-1)} \rceil}^{\lceil sw_k \rceil - 1} w_{mj, J_0, J-1}(t_{J-1}) [X_{sk}(t_j) - X_{s(L+1)}(t_j)] \quad (4.14)$$

and the error correlations are given in (4.2) with these biases.

## 4.2 MODEL MISMATCH DUE TO UNDERMODELING

Suppose that the actual trajectory  $X_s(t)$  is described by

$$X_s(t) = \sum_{p=0}^{d_s} \theta_{sp, J_0, J-1} \xi_{p, J-J_0}(u_{J_0}(t)) , \quad (4.15)$$

then  $X_s^{[m]}(t)$  which we would like to estimate is given as

$$X_s^{[m]}(t) = \frac{1}{\Delta^m} \sum_{p=m}^{d_s} \theta_{sp, J_0, J-1} \xi_{p, J-J_0}^{[m]}(u_{J_0}(t)) . \quad (4.16)$$

Based upon the trajectory model in (3.28), the estimates  $\hat{X}_{m, J_0, J-1}(t)$  of  $X_s^{[m]}(t)$  and the covariances  $V_{mn, J_0, J-1}(t)$  are given in (3.35) and (3.40), respectively. If the model is correct, i.e.,  $d_m = d_s$ , then these estimates are optimal. The estimation problems are called undermodeling and overmodeling if  $d_m < d_s$  and  $d_m > d_s$ , respectively. We will analyze the performance of the estimators of (3.35) for these two cases in this sub-section and next sub-section. We motivate the discussion on the issue of undermodeling by a simple example.

### Example 2

Suppose we use a constant value model for  $X(t)$  to estimate a constant velocity trajectory  $X_s(t)$  at time  $t_{J-1}$  from the noisy measurements  $\{ m(t_j) \}_{j=J_0}^{J-1}$ , i.e.,

$$X(t) = \theta_{0, J_0, J-1} \quad (4.17)$$

$$X_s(t) = \theta_{s0, J_0, J-1} + \theta_{s1, J_0, J-1} \xi_{1, J-J_0}(u_{J_0}(t)) \quad (4.18)$$

Then the optimal estimate  $\hat{X}_{0, J_0, J-1}(t)$  of  $X^{[0]}(t) = X_s(t)$  and the corresponding variance are again given by

$$\hat{X}_{0, J_0, J-1}(t) = \frac{1}{J - J_0} \sum_{j=J_0}^{J-1} m(t_j), \quad (4.19)$$

$$V_{00, J_0, J-1}(t) = \frac{\sigma^2}{J - J_0}. \quad (4.20)$$

Taking the expectation of (4.19) and using (2.1) and (4.18), we have

$$\begin{aligned} E [ \hat{X}_{0, J_0, J-1}(t) ] &= \frac{1}{J - J_0} \sum_{j=J_0}^{J-1} X_s(t_j) \\ &= \frac{1}{J - J_0} \sum_{j=J_0}^{J-1} \left[ \theta_{s0, J_0, J-1} + \theta_{s1, J_0, J-1} \left( j - J_0 + 1 - \frac{J - J_0 + 1}{2} \right) \right] \\ &= \theta_{s0, J_0, J-1}. \end{aligned} \quad (4.21)$$

Thus the bias and the mean squared error of the estimate at time  $t_{J-1}$  are given by

$$B_{0, J_0, J-1}(t_{J-1}) = - \theta_{s1, J_0, J-1} \xi_{1, J-J_0}(J - J_0) = - \frac{\theta_{s1, J_0, J-1} (J - J_0 - 1)}{2}, \quad (4.22)$$

$$R_{00, J_0, J-1}(t_{J-1}) = \frac{\sigma^2}{J - J_0} + \left[ \frac{\theta_{s1, J_0, J-1} (J - J_0 - 1)}{2} \right]^2. \quad (4.23)$$

The value of the actual trajectory and the mean of the estimate are depicted in figure 2. It shows that the bias of the estimate is small around the initial time  $t_{J_0}$  and then increases as the time increases. It means that the estimate will not converge to the true value in average, even if we use more and more measurements! Hence, it is not a consistent estimate. The reason is that, although the trajectory may be well approximated by the constant value model in the small time interval around time  $t_{J_0}$ , the model error will becomes larger as the length of the interval increases.  $\square$

For the general trajectory in (4.15) and model in (3.28), we can find the mean of estimates  $\hat{X}_{m, J_0, J-1}(t)$ , similarly to the derivation in (3.38), as follows.

$$E [ \hat{X}_{m, J_0, J-1}(t) ] = \frac{1}{\Delta^m} \sum_{p=m}^{d_m} \theta_{sp, J_0, J-1} \xi_{p, J-J_0}^{[m]}(u_{J_0}(t)). \quad (4.24)$$

Thus, from (4.16), (4.1), (4.2) and (3.37), the biases and the error correlations at  $t_{J-1}$  are given by

$$B_{m, J_0, J-1}(t_{J-1}) = - \frac{1}{\Delta^m} \sum_{p=d_m+1}^{d_s} \theta_{sp, J_0, J-1} \xi_{p, J-J_0}^{[m]}(J - J_0), \quad (4.25)$$

$$\mathbf{R}_{m_n, J_0, J-1}(t_{J-1}) = \frac{\sigma^2}{\Delta^{m+n}} \sum_{p=m}^{d_m} \frac{\xi_{p, J-J_0}^{[m]}(J-J_0) \xi_{p, J-J_0}^{[n]}(J-J_0)}{S(p, J-J_0)} + \mathbf{B}_{m, J_0, J-1}(t_{J-1}) \mathbf{B}_{n, J_0, J-1}(t_{J-1}) . \quad (4.26)$$

The mean squared error of the estimates  $\hat{X}_{m, J_0, J-1}(t)$  at time  $t_{J-1}$  become

$$\mathbf{R}_{m_m, J_0, J-1}(t_{J-1}) = \frac{\sigma^2}{\Delta^{2m}} \sum_{p=m}^{d_m} \frac{[\xi_{p, J-J_0}^{[m]}(J-J_0)]^2}{S(p, J-J_0)} + \mathbf{B}_{m, J_0, J-1}^2(t_{J-1}) . \quad (4.27)$$

The first term in the right hand side of (4.27) is proportional to the common variance of the noise. From theorem 3, each term in the summation will converge to zero at the rate of  $(J - J_0)^{-(2m+1)}$  as  $J$  increases. The second term in (4.27) reveals the extra error due to the undermodeling. From theorem 2,  $\xi_{p, J-J_0}^{[m]}(J - J_0) \approx \frac{p!}{(p-m)!} (J - J_0)^{p-m}$ . Consequently, as  $J$  increases, the extra error term as well as the absolute value of the biases in (4.25) will increase, in general, without a bound!

### 4.3 MODEL MISMATCH DUE TO OVERMODELING

In the following, we will show that in the case of overmodeling, the estimates  $\hat{X}_{m, J_0, J-1}(t)$  of  $X_s^{[m]}(t)$  are still unbiased but the mean squared errors increase as the degree of the model  $d_m$  increases. The following is a simple example to illustrate the effect of overmodeling.

#### Example 3

Suppose we use a constant velocity model  $X(t)$  to estimate a constant value trajectory  $X_s(t)$  at time  $t_{J-1}$  from the noisy measurements  $\{m(t_j)\}_{j=J_0}^{J-1}$ , i.e.,

$$X(t) = \theta_{0, J_0, J-1} + \theta_{1, J_0, J-1} \xi_{1, J-J_0}(u_{J_0}(t)) \quad (4.28)$$

$$X_s(t) = \theta_{s0, J_0, J-1} \quad (4.29)$$

From (3.35), (3.18) and (3.15), the optimal estimate  $\hat{X}_{n, J_0, J-1}(t)$  of  $X_s^{[m]}(t) = X_s(t)$ , for  $m = 0, 1$ , have means as follows.

$$\begin{aligned} E[\hat{X}_{0, J_0, J-1}(t)] &= \sum_{j=J_0}^{J-1} w_{0j, J_0, J-1} X_s(t_j) \\ &= \theta_{s0, J_0, J-1} \sum_{j=J_0}^{J-1} \left[ \frac{1}{J-J_0} + \frac{\xi_{1, J-J_0}(u_{J_0}(t)) \xi_{1, J-J_0}(j-J_0+1)}{S(1, J-J_0)} \right] \\ &= \theta_{s0, J_0, J-1} + \frac{12 \theta_{s0, J_0, J-1}}{(J-J_0)(J-J_0-1)(J-J_0+1)} (u_{J_0}(t) - \frac{J-J_0+1}{2}) \end{aligned}$$



$$\begin{aligned} & \times \sum_{j=J_0}^{J-1} \left( j - J_0 + 1 - \frac{J - J_0 + 1}{2} \right) \\ & = \theta_{s0, J_0, J-1} , \end{aligned} \tag{4.30}$$

$$\begin{aligned} E [ \hat{X}_{1, J_0, J-1}(t) ] &= \sum_{j=J_0}^{J-1} w_{1j, J_0, J-1}(t) X_s(t_j) \\ &= \frac{\theta_{s0, J_0, J-1}}{\Delta} \sum_{j=J_0}^{J-1} \frac{\xi_{1, J-J_0}(u_{J_0}(t)) \xi_{1, J-J_0}(j - J_0 + 1)}{S(1, J - J_0)} \\ &= \frac{12 \theta_{s0, J_0, J-1}}{\Delta (J - J_0)(J - J_0 - 1)(J - J_0 + 1)} \sum_{j=J_0}^{J-1} \left( j - J_0 + 1 - \frac{J - J_0 + 1}{2} \right) \\ &= 0 . \end{aligned} \tag{4.31}$$

Note that  $X_s^{[0]}(t) = X_s(t) = \theta_{s0, J_0, J-1}$  and  $X_s^{[1]}(t) = \frac{d}{dt} X_s(t) = 0$ . Hence, the above estimates are unbiased. Then, from (3.37) and (4.2), the mean squared errors of the estimates  $\hat{X}_{m, J_0, J-1}(t)$ , for  $m = 0, 1$ , at time  $t_{J-1}$ , are given by

$$\begin{aligned} R_{00, J_0, J-1}(t_{J-1}) &= \sigma^2 \left[ \frac{1}{J - J_0} + \frac{[ \xi_{1, J-J_0}(u_{J_0}(t_{J-1})) ]^2}{S(1, J - J_0)} \right] \\ &= \frac{\sigma^2}{J - J_0} + \frac{3 \sigma^2 (J - J_0 - 1)}{(J - J_0)(J - J_0 + 1)} , \end{aligned} \tag{4.32}$$

$$\begin{aligned} R_{11, J_0, J-1}(t_{J-1}) &= \frac{\sigma^2 [ \xi_{1, J-J_0}^{[1]}(u_{J_0}(t_{J-1})) ]^2}{\Delta^2 S(1, J - J_0)} \\ &= \frac{12 \sigma^2}{\Delta^2 (J - J_0)(J - J_0 - 1)(J - J_0 + 1)} . \end{aligned} \tag{4.33}$$

Although the estimates do not have biases, the above mean squared errors show the performance degradation. In the right hand side of (4.32), the first term  $\frac{\sigma^2}{J - J_0}$  is equal to the minimum mean squared error if we use the correct model, i.e., the constant value model. The second term indicates the extra error due to the overmodeling. It equals about three times of  $\frac{\sigma^2}{J - J_0}$ . The error variance in (4.33)

shows the similar degradation. Note that  $R_{11, J_0, J-1}(t_{j-1})$  is equal to zero if we use the constant value model. Fortunately, these extra errors converge to zero at the rate of  $(J - J_0)^{-1}$  and  $(J - J_0)^{-3}$ , respectively, as  $J$  increases.  $\square$

Similarly to the discussion on the undermodeling in previous sub-section, we can show that the mean of estimates  $\hat{X}_{m, J_0, J-1}(t)$  for the general trajectory model (4.15) are given by

$$E [ \hat{X}_{m, J_0, J-1}(t) ] = \frac{1}{\Delta^m} \sum_{p=m}^{d_m} \sum_{q=0}^{d_s} \frac{\xi_{p, J-J_0}^{[m]}(u_{J_0}(t)) \theta_{sq, J_0, J-1}}{S(p, J - J_0)} S(q, J - J_0) \delta_{p, q} . \quad (4.34)$$

From (4.16) and using  $d_m > d_s$ ,

$$\begin{aligned} E [ \hat{X}_{m, J_0, J-1}(t) ] &= \frac{1}{\Delta^m} \sum_{p=m}^{d_s} \theta_{sp, J_0, J-1} \xi_{p, J-J_0}^{[m]}(u_{J_0}(t)) \\ &= X_s^{[m]}(t) \end{aligned} \quad (4.35)$$

Thus, the estimates are unbiased and the error correlations  $R_{mn, J_0, J-1}(t)$  are equal to the covariances of estimates  $V_{mn, J_0, J-1}(t)$ , i.e.,

$$R_{mn, J_0, J-1}(t) = \frac{\sigma^2}{\Delta^{m+n}} \sum_{p=m}^{d_m} \frac{\xi_{p, J-J_0}^{[m]}(u_{J_0}(t)) \xi_{p, J-J_0}^{[n]}(u_{J_0}(t))}{S(p, J - J_0)} \quad (4.36)$$

It implies that the mean squared errors  $R_{mm, J_0, J-1}(t)$  of  $\hat{X}_{m, J_0, J-1}(t)$  at time  $t_{j-1}$  are given by

$$\begin{aligned} R_{mm, J_0, J-1}(t_{j-1}) &= \frac{\sigma^2}{\Delta^{2m}} \sum_{p=m}^{d_m} \frac{[ \xi_{p, J-J_0}^{[m]}(J - J_0) ]^2}{S(p, J - J_0)} \\ &= \frac{\sigma^2}{\Delta^{2m}} \sum_{p=m}^{d_s} \frac{[ \xi_{p, J-J_0}^{[m]}(J - J_0) ]^2}{S(p, J - J_0)} + \frac{\sigma^2}{\Delta^{2m}} \sum_{p=d_s+1}^{d_m} \frac{[ \xi_{p, J-J_0}^{[m]}(J - J_0) ]^2}{S(p, J - J_0)} \end{aligned} \quad (4.37)$$

The first term in the right hand side of (4.37) is equal to the minimum mean squared errors if the degree of the trajectory model  $d_m = d_s$ . By using theorem 3, each term in the summation converge to zero at the rate of  $(J - J_0)^{-(2m+1)}$  as  $J$  increases. The second term is the extra error due to the overmodeling. Since  $S(p, J - J_0)$  is greater than zero for  $(J - J_0) > d_m > d_s$ , this extra error is positive and will increase as  $d_m$  increases. Fortunately, this term will also converge to zero at the rate of  $(J - J_0)^{-(2m+1)}$  as  $J$  increases.

## 5. FLAT MMF

In this section, we propose a new filter called FLAT MMF in an attempt to solve the problem of the model mismatches we have discussed in previous sections. Section 5.1 discusses the motivation for this filter. Section 5.2 reviews the multiple model filter and describes the basic structure of FLAT MMF. Section 5.3 analyzes the behavior of FLAT MMF on the model mismatches. Section 5.4 summarizes some simulation results of applying FLAT MMF on the model mismatches.

### 5.1 MOTIVATION FOR FLAT MMF

Let us consider the problem of parameter jumping first. As we discussed in section 4.1, the reason that the estimates of the parameter jumping have a very large biases and mean squared errors after the particle switches its value is that the filter has memorized many invalid measurements. For example 1 in section 4.1, one may expect that if we start another filter some time after the first one, then the estimate from this filter will have smaller error since it has memorized less invalid measurements. In the extreme case, if the filter is started after the switch time, then the estimates will be unbiased because the Kalman filter provides the unbiased estimates with minimum mean squared error if the trajectory model matches to the actual motion. Thus, we would like to design a filter in which part or all of these invalid measurements are suppressed. To illustrate this discussion further, let us conduct a simple experiment.

#### Experiment 1

Consider a particle that moves from (5, 5, 20) units to (-4, 2, 20) units with velocity (-4.16, -1.38, 0) units/second then moves to (0, -5, 20) units with velocity (2.176, -3.81, 0) units/second. Thus, there is no depth change in the motion and the velocity is piecewise constant. Assume we use a constant velocity model to estimate this motion and we start the estimator at time 0, 20, 40, 60 and 80 samples. Figure 3a depicts the estimates of velocity  $X^{[1]}(t) / Z(0)$ . (Detail of the experiment setup and the procedure for finding the estimate will be discussed in section 5.4). As we expected, the model error due to the velocity jumping makes the estimates possess a very large error and the filters started later provide better estimate after the velocity jumping. Note that the older filter has the better noisy suppression than the younger one if there is no velocity jumping since they have started.  $\square$

The above observation may suggest that we restart the filter at every few samples in order to keep the number of invalid measurements small. But this does not work because the filter needs enough samples in finding the estimates and suppressing the noise. However, if we use two or more

filters which are started (triggered) at different times and combine the estimates from these filters properly, then we may be able to obtain the estimates which have good noise suppression and minimize the effect of parameter jumping. It is because that the older filter provides the estimates from more measurements while the younger one memorize less invalid measurements. Moreover, we may restart the oldest filter again after a reasonable period, say 100 samples, because the extra old measurements memorized in this filter become out of date. The above discussion motivates us to propose the FLAT MMF. We will discuss it in next section in more detail.

The proposed FLAT MMF may also solve the problem of undermodeling. As we discussed in section 4.2, if we use a low-order model to estimate a high-order trajectory, the estimates will have biases and these biases increase without a bound in general. However, we observe that these biases are small around the time the filter is started because the trajectory can be approximated well there. The following experiment is conducted to illustrate this point.

## Experiment 2

Suppose a particle moves with constant acceleration without depth change. We use a constant velocity model to estimate its motion. The initial position of the particle is  $(-6.8, -6.8, 20)$  units. It moves with velocity  $[3.4 \ 10 \ 0]^T + [0 \ -5 \ 0]^T \cdot t$  units/second. Figure 4a shows the estimates of  $Y^{(1)}(t) / Z(0)$  as the filter is started at different times. It confirms that the model error due to undermodeling makes the estimates diverge. However, the estimates from the filter started later are better than those from the filter started earlier, and the estimation error around the time the filter is started is small.  $\square$

Thus, if a new filter is started after the first one, it will provide better estimates from the time it is started than the estimates from the older one. Similarly to the discussion on the filter restarting and noise suppression for the case of parameter jumping, we may conclude that if we use a number of filters triggered at different times and combine the estimates from them properly, then the effect of undermodeling is reduced.

For the problem of overmodeling, the situation reverses. The estimates for overmodeling are unbiased and the extra mean squared error due to the 'over-freedom' contaminates the estimates further. If the trajectory model is fixed, then the only way to achieve the better estimates, i.e., less mean squared errors, is to use more measurements. Consequently, the estimates from the filter started later will have larger mean squared errors because it has used less measurement. However, if one can conceive a mechanism that combines the estimates from multiple filters so that the final estimates are dominated by that from the oldest filter, then the estimates obtained from these filters do not degrade

significantly, compared to that from a single filter. Note that, due to the real-time constraints, the order of the trajectory model can not to be high. This is especially true for estimating 3-D object motion from perspective measurements because the size of the state vector equals at least three times of the order of the trajectory model we use. Thus, the problem of overmodeling is less serious than the problem of undermodeling and parameter jumping.

Up to this point, we observe that we may solve the problem of parameter jumping, undermodeling and overmodeling by using a number of differently triggered filters if we combine the estimates from these filters properly. Thus, one may raise two questions: How do we combine these estimates properly and what do we mean ‘properly’? The multiple model filter discussed in the next section provides an answer to these questions.

## 5.2 BASIC STRUCTURE OF FLAT MMF

In this section, we will first review briefly the existing multiple model filter (MMF) and then describe the basic structure of the proposed FLAT MMF. The idea of multiple model filter is first proposed by Magill for estimating the state of system with uncertainty [Magi65]. Since then, a number of applications have been reported and several results on the behavior of MMF have been published. The detail of MMF can be found from the references in [Mayb82]. We give a brief description of MMF.

Suppose we want to estimate the state  $\underline{s}(t)$  at time  $t_i$  of a system of interest from the measurements  $\{\underline{m}(t_j)\}_{j=0}^i$ . Assume that this state estimation problem can be modeled properly by the one in which the plant equation and measurement equation are linear, except there are some uncertainties in the modeling, such as the covariance matrices of the model noise and measurement noise, and some parameters defining the state transition matrix. Let  $\underline{a}$  denote the vector of these uncertain parameters and assume that  $\underline{a}$  belongs to the set of values  $\{\underline{a}_k\}_{k=1}^K$ . Then, for each  $\underline{a}_k$ , we may construct a Kalman filter, based upon the model associated with  $\underline{a}_k$ , to estimate the state. Note that these K filters can be processed simultaneously. The final estimate of the state is obtained by combining the estimate of these K filters. The state estimation based upon the above structure is called multiple model filtering. It can be shown that for the above MMF structure, the optimal state estimate, in the sense of MMSE, is given by

$$\hat{\underline{s}}(t_i^+) = \sum_{k=1}^K \hat{\underline{s}}_k(t_i^+) p_k(t_i) , \quad (5.1)$$

where  $\hat{\underline{s}}_k(t_i^+)$  is the state estimate produced by k-th Kalman filter based on the assumption that the parameter vector equals  $\underline{a}_k$  [Mayb82].  $p_k(t_i)$  is the hypothesis conditional probability and

$$\begin{aligned}
 p_k(t_i) &= \text{Prob}\{ \underline{a} = \underline{a}_k \mid \underline{M}_{0,i} \} \\
 &= \frac{f(\underline{m}(t_i) \mid \underline{a}_k, \underline{M}_{0,i-1}) p_k(t_{i-1})}{\sum_{j=1}^K f(\underline{m}(t_i) \mid \underline{a}_j, \underline{M}_{0,i-1}) p_j(t_{i-1})}, \quad (5.2)
 \end{aligned}$$

The covariance of  $\hat{\underline{s}}(t_i^+)$  is

$$\mathbf{V}(t_i^+) = \sum_{k=1}^K p_k(t_i) \{ \mathbf{V}_k(t_i^+) + [ \hat{\underline{s}}_k(t_i^+) - \hat{\underline{s}}(t_i^+) ] [ \hat{\underline{s}}_k(t_i^+) - \hat{\underline{s}}(t_i^+) ]^T \}, \quad (5.3)$$

where  $\mathbf{V}_k(t_i^+)$  is the covariance of  $\hat{\underline{s}}_k(t_i^+)$  computed by the k-th Kalman filter. The conditional probability  $f(\underline{m}(t_i) \mid \underline{a}_k, \underline{M}_{0,i-1})$  in (5.2) can be evaluated as

$$f(\underline{m}(t_i) \mid \underline{a}_k, \underline{M}_{0,i-1}) = \frac{1}{(2\pi)^{n_m/2} | \mathbf{A}_k(t_i) |^{1/2}} \exp \left\{ - \frac{1}{2} \underline{r}_k^T(t_i) \mathbf{A}_k^{-1}(t_i) \underline{r}_k(t_i) \right\}, \quad (5.4)$$

where

$$\mathbf{A}_k(t_i) = \mathbf{H} \mathbf{V}_k(t_i^-) \mathbf{H}^T + \mathbf{R}, \quad (5.5)$$

$$\underline{r}_k(t_i) = \underline{m}(t_i) - \mathbf{H} \hat{\underline{s}}(t_i^-), \quad (5.6)$$

where  $\hat{\underline{s}}_k(t_i^-)$ ,  $\mathbf{V}_k(t_i^-)$  and  $\underline{r}_k(t_i)$  are the state estimate, the covariance and the residual at  $t_i$  of the k-th Kalman filter, based upon the measurements  $\{ \underline{m}(t_j) \}_{j=1}^{i-1}$ , respectively.  $\mathbf{R}$ ,  $\mathbf{H}$  and  $n_m$  are the covariance matrix of measurement noise, the measurement matrix and the total number of measurements at each time, respectively. Note that  $\mathbf{A}_k(t_i)$  and  $\underline{r}_k(t_i)$  are available as the intermediate result of the k-th Kalman filter. Thus, the conditional probability  $f(\underline{m}(t_i) \mid \underline{a}_k, \underline{M}_{0,i-1})$  as well as the weighting factors  $p_k(t_i)$  can be obtained with a small amount of increase of computation.

In summary, the MMF is composed of a bank of  $K$  separate Kalman filters, each of which is based on a particular parameter vector  $\underline{a}_k$ . The overall state estimate is the linear combination of the state estimates generated by these Kalman filters. The weighting factors  $p_k(t_i)$  is updated recursively according to (5.2), using the current  $\mathbf{A}_k(t_i)$  and  $\underline{r}_k(t_i)$ . The block diagram of the MMF is depicted in figure 5. All the filters are run simultaneously and the extra computation in updating the weighting factors  $p_k(t_i)$  compared to the normal Kalman filter is negligible.

The FLAT MMF is composed of a set of  $K$  identical Kalman filters, each triggered at different time. The overall state estimate is the probabilistically weighted average of the state estimates generated by these Kalman filters, as we discussed for the MMF. Without loss of generality, we assume that the k-th filter is triggered at time  $(k-1)J_t \Delta$ , where  $J_t$  is an integer. Each filter will die out every

( $K J_t \Delta$ ) seconds and then the filter will be triggered again, i.e., each filter only has a lifetime of ( $K J_t \Delta$ ). Figure 6 shows the timing of the FLAT MMF for  $K = 4$ . At any time, in general, there are  $K$  filters being processed simultaneously. Thus, a FLAT MMF is a MMF in which all the filters are identical but have the different starting time, i.e., the uncertain parameter vector  $\underline{a}$  discussed before is the time that the filter is started. For the nonlinear state estimation problem such as the problem of motion estimation from perspective measurements, FLAT MMF can still be used to estimate the state if we replace the Kalman filter by the extended Kalman filter or some nonlinear filter. Furthermore, we may include other uncertain parameters into the estimation process by replacing each filter with a MMF representing those uncertain parameters.

The key feature of the FLAT MMF is that the differently triggered filters operate on different sets of past measurements. Hence, as we discussed in section 5.1, the estimates from these filters contain the one that has good noise suppression for the case of the trajectory model matches to the actual trajectory or the case of overmodeling, the one that contains small number or none of invalid measurements for the case of parameter jumping, and the one that has a small model error for the case of undermodeling. The structure of MMF provides a way to combine these estimates properly, in the sense of MMSE, so that the 'best' estimate will 'show up' at the final estimate. The estimation behavior of FLAT MMF for model mismatch is discussed further in the next sub-section. Another feature of FLAT MMF is that all the filters can be processed simultaneously and the computational effort for combining the estimates from the filters is relatively small. Thus, the FLAT MMF can be implemented efficiently for real-time applications.

### 5.3 FLAT MMF AND MODEL MISMATCH

In this section, we will analyze three classes of mismatch model discussed in section 4 when the proposed FLAT MMF is used. Without loss of generality, we will focus on the discussion when  $K = 2$ , i.e., there are only two Kalman filters in the FLAT MMF and  $k = 1, 2$ . Let  $t_i$  be the time we want to estimate,  $t_{a_k}$  be the triggering time of  $k$ -th filter,  $d_s$  be the order of actual trajectory  $X_s(t)$  and  $d_m$  be the order of trajectory model  $X(t)$ . Throughout this section, we assume that  $i > d_m + a_k$ , to make sure that there are enough measurements. Note that the estimates of the filter triggered at  $t_{a_k}$  from the measurements  $\{m(t_j)\}_{j=0}^i$  are equal to that from the measurements  $\{m(t_j)\}_{j=a_k}^i$ . It means that all the formula related to the state estimation in the previous sections can be applied to the filters triggered at different time by simple replacing  $J_0$  with the corresponding  $a_k$ . That is why we included the starting time of available measurements  $J_0$  in the derivations at the previous sections.

For the 1-D motion problem discussed in section 3, the recursive expression of the weighting factors  $p_k(t_i)$  can be obtained explicitly as follows. From (3.43), (3.53), (3.55) and (3.35),  $A_k(t_i)$  in (5.5) and the residual  $r_k(t_i)$  in (5.6) are given by

$$A_k(t_i) = V_{00, a_k, i-1}(t_i) + \sigma^2, \quad (5.7)$$

$$r_k(t_i) = m(t_i) - \hat{X}_{0, a_k, i-1}(t_i) = m(t_i) - \sum_{j=a_k}^{i-1} w_{0j, a_k, i-1}(t_i) m(t_j). \quad (5.8)$$

Note that the total number of measurements at each time is  $n_m = 1$  and  $A_k(t_i) > 0$  for  $\sigma^2 \neq 0$ . The conditional probability is given by

$$f(m(t_i) | \underline{a}_k, \underline{M}_{0, i-1}) = \frac{1}{(2\pi)^{1/2} A_k^{1/2}(t_i)} \exp \left\{ - \frac{r_k^2(t_i)}{2 A_k(t_i)} \right\}. \quad (5.9)$$

The weighting factors  $p_k(t_i)$  in (5.2) are the normalized product of these  $f(m(t_i) | \underline{a}_k, \underline{M}_{0, i-1})$  by the last weighting factors  $p_k(t_{i-1})$ . Define the weighting ratio of  $p_k(t_i)$  and  $p_k(t_{i-1})$  as  $p_{kl}(t_i)$ , i.e.,  $p_{kl}(t_i) = p_k(t_i) / p_k(t_{i-1})$ . Let  $\underline{X}_s(i)$  be the composite vectors which comprise the trajectory  $X_s(t)$ ,  $\underline{n}(i)$  be the noise  $n(t)$  and  $\underline{m}(i)$  be the measurement  $m(t)$  from time  $t_0$  to  $t_i$ , i.e.,

$$\underline{X}_s(i) = [ X_s(t_0) \ X_s(t_1) \ \cdots \ X_s(t_i) ]^T, \quad (5.10)$$

$$\underline{n}(i) = [ n(t_0) \ n(t_1) \ \cdots \ n(t_i) ]^T, \quad (5.11)$$

$$\underline{m}(i) = [ m(t_0) \ m(t_1) \ \cdots \ m(t_i) ]^T. \quad (5.12)$$

Let the  $(i+1) \times 1$  vector  $\underline{b}_k(i)$  has elements  $b_{kj}(i)$ , for  $j=0, 1, \dots, i$ , where

$$\underline{b}_k(i) = \begin{cases} 0 & 0 \leq j < a_k \\ \frac{-w_{0j, a_k, i-1}(t_i) \sigma}{A_k^{1/2}(t_i)} & a_k \leq j < i \\ \frac{\sigma}{A_k^{1/2}(t_i)} & j = i \end{cases}. \quad (5.13)$$

From (5.2) and (5.9), we have

$$p_{kl}(t_i) = \left[ \frac{A_1(t_i)}{A_k(t_i)} \right]^{1/2} \exp \left\{ \frac{1}{2\sigma^2} [ (\underline{b}_k^T(i) \underline{m}(i))^2 - (\underline{b}_k^T(i) \underline{m}(i))^2 ] \right\} p_{kl}(t_{i-1}). \quad (5.14)$$

Let

$$\alpha_{kl}(i) = \ln \frac{p_{kl}(t_i)}{p_{kl}(t_{i-1})}. \quad (5.15)$$



It means that

$$p_{k|l}(t_i) = p_{k|l}(t_{i-1}) e^{\alpha_{k|l}(i)} . \quad (5.16)$$

From (5.14), we have

$$\alpha_{k|l}(i) = \frac{1}{2} \ln \frac{A_1(t_i)}{A_k(t_i)} + \frac{1}{2 \sigma^2} \underline{m}^T(i) [ \underline{b}_1(i) \underline{b}_1^T(i) - \underline{b}_k(i) \underline{b}_k^T(i) ] \underline{m}(i) . \quad (5.17)$$

Note that the norm of  $\underline{b}_k(i)$ , the inner product  $\underline{b}_k^T(i) \underline{b}_1(i)$  and the inner product  $\underline{b}_k^T(i) \underline{X}_s(i)$  satisfy the following lemma.

**Lemma 1**

- (i) The norm of  $\underline{b}_k(i)$  equals to one.
- (ii) The inner product of  $\underline{b}_k(i)$  and  $\underline{b}_1(i)$  equals approximately to one for large  $i$ .
- (iii) The inner product of  $\underline{b}_k(i)$  and  $\underline{X}_s(i)$  is given by

$$\underline{b}_k^T(i) \underline{X}_s(i) = \frac{-B_{0, a_k, i-1}(t_i) \sigma}{A_k^{1/2}(t_i)} , \quad (5.18)$$

where  $B_{0, a_k, i-1}(t_i)$  is the bias of the estimate  $\hat{X}_{0, a_k, i-1}(t_i)$ , using the measurement  $\{m(t_j)\}_{j=a_k}^{i-1}$ .  $\square$

The proof of this lemma is given in Appendix. The mean and the variance of  $\alpha_{k|l}(i)$  satisfy the following theorem.

**Theorem 5**

$$E [ \alpha_{k|l}(i) ] = \frac{1}{2} \ln \frac{A_1(t_i)}{A_k(t_i)} + \frac{1}{2 \sigma^2} [ ( \underline{b}_1^T(i) \underline{X}_s(i) )^2 - ( \underline{b}_k^T(i) \underline{X}_s(i) )^2 ] \quad (5.19)$$

$$= \frac{1}{2} \ln \frac{A_1(t_i)}{A_k(t_i)} + \frac{1}{2} \left[ \frac{B_{0, a_1, i-1}^2(t_i)}{A_1(t_i)} - \frac{B_{0, a_k, i-1}^2(t_i)}{A_k(t_i)} \right] , \quad (5.20)$$

$$\begin{aligned} \text{Var} [ \alpha_{k|l}(i) ] = & 1 - ( \underline{b}_k^T(i) \underline{b}_1(i) )^2 + \frac{1}{\sigma^2} [ ( \underline{b}_k^T(i) \underline{X}_s(i) )^2 + ( \underline{b}_1^T(i) \underline{X}_s(i) )^2 \\ & - 2 ( \underline{b}_k^T(i) \underline{b}_1(i) ) ( \underline{b}_k^T(i) \underline{X}_s(i) ) ( \underline{b}_1^T(i) \underline{X}_s(i) ) ] \quad (5.21) \end{aligned}$$

$$\approx \left[ \frac{B_{0, a_1, i-1}(t_i)}{A_1^{1/2}(t_i)} - \frac{B_{0, a_k, i-1}(t_i)}{A_k^{1/2}(t_i)} \right]^2 . \quad (5.22)$$

The proof of this theorem is given in Appendix.

Note that, from (3.37) and the definition of  $V(p, N, u, m)$  in (3.23), the variances of the estimates  $\hat{X}_{m, a_k, i-1}(t_i)$  and  $\hat{X}_{m, a_k, i}(t_i)$  can be represented as

$$V_{mm, a_k, i-1}(t_i) = \frac{\sigma^2}{\Delta^{2m}} \sum_{p=m}^{d_m} V(p, i - a_k, i - a_k + 1, m), \quad (5.23)$$

$$V_{mm, a_k, i}(t_i) = \frac{\sigma^2}{\Delta^{2m}} \sum_{p=m}^{d_m} V(p, i - a_k + 1, i - a_k + 1, m). \quad (5.24)$$

Thus, from theorem 3, we have the following theorem.

### Theorem 6

If  $i > a_k + d_m$  and  $i > a_1 + d_m$ , then the following is true.

- (i) if  $a_1 > a_k$  then  $V_{00, a_1, i-1}(t_i) > V_{00, a_k, i-1}(t_i)$ ,  $V_{00, a_1, i}(t_i) > V_{00, a_k, i}(t_i)$  and  $A_1(t_i) > A_k(t_i)$ .
- (ii) if  $a_1 > a_k$  then  $V_{mm, a_1, i-1}(t_i) > V_{mm, a_k, i-1}(t_i)$  and  $V_{mm, a_1, i}(t_i) > V_{mm, a_k, i}(t_i)$ , for  $m > 0$  and large  $i$ .

□

The proof of theorem 6 is given in Appendix.

### 5.3.1 PARAMETER JUMPING

Let  $t_{sw}$  be the time at which the actual trajectory switches its parameter values. Without loss of generality, we assume that  $3J_t \Delta < t_{sw} \leq 4J_t \Delta$ , where  $J_t \Delta$  is the time separation between two adjunct triggerings. We will analyze the behavior of FLAT MMF in the time intervals  $[3J_t \Delta, t_{sw})$ ,  $[t_{sw}, 4J_t \Delta)$  and  $[4J_t \Delta, 5J_t \Delta)$ . Figure 7 depicts the timing of the two parallelly operated filters and the switching time of the trajectory.

Still, we assume that the actual trajectory  $X_s(t)$  is represented as in (4.10) and the trajectory model  $X(t)$  is given by (3.28) with  $d_m = d_s$ . Then the optimal estimates  $\hat{X}_{m, a_k, i}(t_i)$  of  $X_s^{[m]}(t_i)$  and the corresponding covariances are given by (3.35) and (3.37), respectively as we substitute  $J_0 = a_k$ ,  $J = i + 1$  and  $t = t_i$  to them.

(i)  $3J_t \Delta \leq t_i < t_{sw}$  : In this interval, the triggered times of the filter are  $a_1 = 2J_t$  and  $a_2 = 3J_t$ . The mean  $\hat{X}_{m, a_k, i}(t_i)$ , from (3.35) and (4.10), is given by

$$E [\hat{X}_{m, a_k, i}(t_i)] = \sum_{j=a_k}^i w_{mj, a_k, i}(t_i) X_s(t_j) = \sum_{j=a_k}^i w_{mj, a_k, i}(t_i) X_{s1}(t_j). \quad (5.25)$$

Similarly to the discussion in (3.38), we can prove that the above expression equals to  $X_s^{[m]}(t) = \frac{d^m X_{s1}(t)}{dt^m}$  at time  $t = t_i$ . Hence, the estimates  $\hat{X}_{m, a_k, i}(t_i)$  are unbiased and the mean squared error is given as  $R_{mm, a_k, i}(t_i) = V_{mm, a_k, i}(t_i)$ . Similarly, we can show that the estimates  $\hat{X}_{m, a_k, i-1}(t_i)$  are also unbiased. Since  $a_1 < a_2$ , from theorem 6,  $R_{mm, a_1, i}(t_i) < R_{mm, a_2, i}(t_i)$  for large  $i$ . Thus, the estimates from filter 1 are better than that from filter 2 within this interval. This result makes sense because filter 1 uses more (valid) measurements than filter 2 during the estimation process. In the following, we will show that the overall estimates of FLAT MMF are almost equal to that of filter 1. It means that the performance of FLAT MMF is similar to that of the normal Kalman filter if the trajectory model matches the actual trajectory.

Since the estimates  $\hat{X}_{0, a_k, i-1}(t_i)$  are unbiased, i.e.  $B_{0, a_k, i-1}(t_i) = 0$  for  $k = 1, 2$ , from theorem 5 the mean and the variance of  $\alpha_{1|2}(i)$  are given by

$$E [ \alpha_{1|2}(i) ] = \frac{1}{2} \ln \frac{A_2(t_i)}{A_1(t_i)} , \quad (5.26)$$

$$\text{Var} [ \alpha_{1|2}(i) ] \approx 0 .$$

This implies that

$$p_{1|2}(t_i) = p_{1|2}(t_{i-1}) \frac{A_2^{1/2}(t_i)}{A_1^{1/2}(t_i)} . \quad (5.27)$$

Since  $a_1 < a_2$ , from theorem 6 we can derive that  $p_{1|2}(t_i) > p_{1|2}(t_{i-1})$ . Thus,

$$p_1(t_i) > p_1(t_{i-1}) > \dots > p_1(3J_t \Delta) = 1 - p_{st} , \quad (5.28)$$

$$p_2(t_i) < p_2(t_{i-1}) < \dots < p_2(3J_t \Delta) = p_{st} , \quad (5.29)$$

where  $p_{st}$  is a small number, say 0.01, assigned to the weighting factor of the newly starting filter. Consequently, the overall estimates of the FLAT MMF in (5.1) are dominated by the estimates from filter 1, as we expected.

(ii)  $t_{sw} \leq t_i < 4J_t \Delta$ : In this interval, the triggered times  $a_1$  and  $a_2$  are still equal to  $2J_t$  and  $3J_t$ , respectively. From (4.12), (4.2) and (3.37), the bias and the mean squared error of the estimates  $\hat{X}_{m, a_k, i}(t_i)$  are given by

$$B_{m, a_k, i}(t_i) = \sum_{j=a_k}^{[sw]-1} w_{mj, a_k, i}(t_i) [ X_{s1}(t_j) - X_{s2}(t_j) ] , \quad (5.30)$$

$$R_{mm, a_k, i}(t_i) = \frac{\sigma^2}{\Delta^{2m}} \sum_{p=m}^{d_m} \frac{[ \xi_{sp, i-a_k+1}^{[m]} (i - a_k + 1) ]^2}{S(p, i - a_k + 1)} + [B_{m, a_k, i}(t_i)]^2 . \quad (5.31)$$

These biases and mean squared errors depend on the values of  $w_{mj, a_k, i}(t_i)$ ,  $i$ ,  $X_{s1}(t_j)$  and  $X_{s2}(t_j)$ , for  $j = a_k, \dots, [sw] - 1$ . Hence, for an arbitrary trajectory, we can not say which filter always produce better estimates-less biases and smaller mean squared errors-within the entire interval  $[t_{sw}, 4J_t \Delta)$ . In figure 8, we present some examples and counter-examples. In figures 8a and 8b the mean of the estimated trajectories from filter 2 is always closer to the actual trajectory than that from filter 1. In figures 8c and 8d, the estimated trajectory from filter 2 is sometime worse than that from filter 1. However, since the filter 2 uses less invalid measurements than the filter 1, it is expected that, most of the time, the bias of the estimates from filter 2 are smaller than that from filter 1. If this is true, then  $B_{0, a_1, i-1}^2(t_i) > B_{0, a_2, i-1}^2(t_i)$ . Since  $a_1 < a_2$ , from theorem 6  $A_1(t_i) < A_2(t_i)$ . Thus,  $\frac{B_{0, a_1, i-1}^2(t_i)}{A_1(t_i)} > \frac{B_{0, a_2, i-1}^2(t_i)}{A_2(t_i)}$ .

From theorem 5, we have

$$E [\alpha_{21}(i)] \approx \frac{1}{2} \left[ \frac{B_{0, a_1, i-1}^2(t_i)}{A_1(t_i)} - \frac{B_{0, a_2, i-1}^2(t_i)}{A_2(t_i)} \right] > 0. \quad (5.32)$$

Consequently, we may conclude that the weighting factor  $p_{21}(t_i)$ , in (5.16), increases exponentially as  $i$  increases and the overall estimates of FLAT MMF will be dominated by the estimates from filter 2.

For the case of  $d_m = 0$ , the above expectation that filter 2 generates better estimates is always true. It is shown as follows. For  $d_m = 0$ , the bias and mean squared error of estimate  $\hat{X}_{0, a_k, i}(t_i)$  are given by

$$B_{0, a_k, i}(t_i) = \frac{[sw] - a_k}{i - a_k + 1} (\theta_{s10, a_k, i} - \theta_{s20, a_k, i}), \quad (5.33)$$

$$R_{00, a_k, i}(t_i) = \frac{\sigma^2}{i - a_k + 1} + B_{0, a_k, i}^2(t_i). \quad (5.34)$$

Note that  $\theta_{s10, a_1, i} = \theta_{s10, a_2, i}$  and  $\theta_{s20, a_1, i} = \theta_{s20, a_2, i}$ . Since  $a_1 < a_2$ , it can be shown that  $\frac{[sw] - a_1}{i - a_1 + 1} > \frac{[sw] - a_2}{i - a_2 + 1}$ . From (5.33),  $|B_{0, a_1, i}(t_i)| > |B_{0, a_2, i}(t_i)|$ , i.e., the estimate from filter 1 has the larger bias. For small noise, i.e.,  $\sigma^2 \approx 0$ , the first term in the right hand side of (5.34) can be neglected and then  $R_{00, a_1, i}(t_i) > R_{00, a_2, i}(t_i)$ .

(iii)  $4J_t \Delta \leq t_i < 5J_t \Delta$ : In this interval,  $a_1 = 4J_t$ , and  $a_2 = 3J_t$ . The estimates  $\hat{X}_{m, a_2, i}(t_i)$  from filter 2 still have biases and mean squared errors given by (5.30) and (5.31), respectively. On the other hand, the estimates  $\hat{X}_{m, a_1, i-1}(t_i)$  and  $\hat{X}_{m, a_1, i}(t_i)$  from filter 1 are both unbiased because all the measurements which filter 1 uses are valid. The mean squared error of estimates  $\hat{X}_{m, a_1, i}(t_i)$  is still given in (5.31) except  $B_{m, a_1, i}(t_i) = 0$ . Thus, the estimates from filter 1 are better than that from filter 2 within this interval.

Since  $B_{0, a_1, i-1}(t_i) = 0$ , from theorem 5 we have

$$E [ \alpha_{1l2}(i) ] \approx \frac{1}{2} \frac{B_{0, a_2, i-1}^2(t_i)}{A_2(t_i)} > 0 . \quad (5.35)$$

Hence, the weighting ratio  $p_{1l2}(t_i)$  in (5.16) increases at an exponential rate. Consequently, the overall estimates of FLAT MMF approach to that of filter 1 exponentially.

### 5.3.2 UNDERMODELING

Without loss of generality, we consider that filter 2 is the 'younger' one, i.e.,  $a_1 < a_2$ . For the general trajectory in (4.15) and trajectory model in (3.28), from (4.25) and (4.27), the bias and the mean squared error of the estimates  $\hat{X}_{m, a_k, i}(t_i)$  are given by

$$B_{m, a_k, i}(t_i) = - \frac{1}{\Delta^m} \sum_{p=d_m+1}^{d_s} \theta_{sp, a_k, i} \xi_{\mathcal{P}, i}^{[m]} - a_k + 1 (i - a_k + 1) , \quad (5.36)$$

$$R_{mm, a_k, i}(t_i) = \frac{\sigma^2}{\Delta^{2m}} \sum_{p=m}^{d_m} \frac{[ \xi_{\mathcal{P}, i}^{[m]} - a_k + 1 (i - a_k + 1) ]^2}{S(p, i - a_k + 1)} + B_{m, a_k, i}^2(t_i) . \quad (5.37)$$

The above biases and mean squared errors depend upon the values of  $d_s$ ,  $d_m$ ,  $i$  and  $\theta_{sp, a_k, i}$ , for  $k = 1, 2$ . Hence, for an arbitrary trajectory, we can not say that which filter always produces the better estimate. Figure 9 shows an example for the case of  $d_m = 0$ ,  $d_s = 2$ . The mean of the estimate  $\hat{X}_{0, a_k, i}(t_i)$ ,  $k = 1, 2$ , are plotted. It is seen that the bias of the estimate from filter 2 in the interval  $(t_A, t_B)$  is smaller than that from filter 1, while it is larger for other interval. However, for large  $i$  and using theorem 2, we may approximate the biases as follows

$$B_{m, a_k, i}(t_i) \approx - \frac{1}{\Delta^m} \theta_{sd_s, a_k, i} \frac{d_s!}{(d_s - m)!} (i - a_k + 1)^{d_s - m} . \quad (5.38)$$

Since  $\theta_{sd_s, a_1, i} = \theta_{sd_s, a_2, i}$  and  $a_1 < a_2$ ,  $|B_{m, a_1, i}(t_i)| > |B_{m, a_2, i}(t_i)|$ . For small noise and/or large  $i$ ,  $R_{mm, a_k, i}(t_i) \approx B_{m, a_k, i}^2(t_i)$ . This implies that  $R_{mm, a_1, i}(t_i) > R_{mm, a_2, i}(t_i)$ . Thus, the estimate from filter 2 is better than that from filter 1. Similarly, for large  $i$ ,  $|B_{m, a_1, i-1}(t_i)| > |B_{m, a_2, i-1}(t_i)|$ . From theorem 6

and using  $a_1 < a_2$ ,  $A_1(t_i) < A_2(t_i)$ . Hence,  $\frac{B_{0, a_1, i-1}^2(t_i)}{A_1(t_i)} > \frac{B_{0, a_2, i-1}^2(t_i)}{A_2(t_i)}$ . Using theorem 5, we have

$$E [ \alpha_{2l1}(i) ] \approx \frac{1}{2} \left[ \frac{B_{0, a_1, i-1}^2(t_i)}{A_1(t_i)} - \frac{B_{0, a_2, i-1}^2(t_i)}{A_2(t_i)} \right] > 0 . \quad (5.39)$$

Thus, we may conclude that the weighting factor  $p_{21}(t_i)$ , in (5.16), increases exponentially as  $i$  increases and the estimates from filter 2 dominate the overall estimates of FLAT MMF at an exponential rate.

Although there are still biases in the estimation, the biases are much smaller than that of one Kalman filter. The maximum of the absolute value of the biases of the estimates  $\hat{X}_{m, a_k, i}(t_i)$ , from (5.38), is approximately equal to  $(\frac{1}{\Delta^m} \theta_{sd, a_k, i} \frac{d_s!}{(d_s - m)!} (J_t + 1)^{d_s - m})$ . The maximum bias occurs at the time just before another filter is triggered. Then the bias decreases dramatically because the newly triggered filter provides much better estimates.

### 5.3.3 OVERMODELING

Still, we consider that filter 2 is the ‘younger’ one, i.e.,  $a_1 < a_2$ . For the actual trajectory in (4.15) and the trajectory model in (3.28) with  $d_s < d_m$ , from the discussion in section 4.3, the estimates  $\hat{X}_{m, a_k, i}(t_i)$  are unbiased and their mean squared error is given by  $R_{mm, a_k, i}(t_i) = V_{mm, a_k, i}(t_i)$ . Since  $a_1 < a_2$ , from theorem 6,  $R_{mm, a_1, i}(t_i) < R_{mm, a_2, i}(t_i)$  for large  $i$ . It means that the estimates from filter 1 are better than those from filter 2. Since the estimates  $\hat{X}_{m, a_k, i-1}(t_i)$  are also unbiased, similarly to the discussion in (i) of sub-section 5.3.1, we conclude that the overall estimates of FLAT MMF are dominated by the estimates from filter 1.

### 5.3.4 SUMMARY OF THE ANALYSIS OF FLAT MMF

We summarize the analysis of FLAT MMF from previous three sub-sections as follows.

- (i) For the problem of parameter jumping, the overall estimates before the particle switches its value, are dominated by the estimates from the oldest filter, i.e., the one that provides the largest noise suppression. After the switch occurs, the overall estimates approach exponentially to the estimates from the youngest filter which has been started before the switch. Then after another filter is triggered, the estimates from this newly started filter get control of the overall estimates exponentially and the estimation of FLAT MMF approaches to the one with MMSE estimates.
- (ii) For the undermodeling, the overall estimates approach exponentially to the estimates from the newly started filter whenever a new triggering occurs. Although the overall estimates have some biases, these biases are much smaller than those from a single filter. These biases depend upon the time interval between the triggerings ( $J_t \Delta$ ) and the difference between the trajectory model and the true trajectory.

- (iii) For the overmodeling, the overall estimates are dominated by the estimates of the oldest filter. Thus, the estimation does not degrade significantly if we use the FLAT MMF instead of the conventional filter.

## 5.4 SIMULATION RESULTS

In order to illustrate the performance degradation on model mismatches and the performance improvement when the FLAT MMF is adopted, a number of experiments on simulated data are conducted. The noisy measurements of the trajectory at different sampling time  $t_j$ ,  $j = 0, 1, \dots$ , are generated by adding white zero mean Gaussian noise to the actual trajectory. The standard deviations of the noise are set to 2.5 pixels. The focal length of the camera is set to one unit. The visible portion of the image plane is  $(-0.36, 0.36) \times (-0.36, 0.36)$  units. This portion corresponds to the viewing angle of  $\pm 20$  degrees. The size of observed image is considered as  $256 \times 256$  pixels. The time interval between frames is 0.04 second. Kalman filter is used to find the unknown states. The initial estimates of position are set to their measurements at  $t_0$  and those of the states corresponding to their derivatives are set to zero.

### Experiment 1 and 2 (continue)

For experiments 1 and 2 discussed in section 5.1, we use the FLAT MMF with two Kalman filters. Each filter is triggered at every 25 samples. Figures 3b and 4b depict the estimates of  $X^{[1]}(t) / Z(0)$  and  $Y^{[1]}(t) / Z(0)$  for these experiments, respectively. These results show that the FLAT MMF works quite well in handling the model mismatches. If we compare figures 3b and 4b to 3a and 3b respectively, one may find that the overall estimates of FLAT MMF are formed by just 'cutting out' the correct portions of the estimates from two filters.

### Experiment 3: parameter jumping

In this experiment, the particle moves on the plane  $Z = 20$  units from the initial position  $(-5, 0, 20)$  units. After six turns, the particle moves back to its starting position. The angles of these six turns are  $90^\circ$ ,  $120^\circ$ ,  $-60^\circ$ ,  $180^\circ$ ,  $-120^\circ$  and  $150^\circ$ , respectively. Figure 10a shows the trajectory and its noisy measurement. Figures 10b and 10c depict the estimates of velocity  $X^{[1]}(t) / Z(0)$  and the predictions of trajectory respectively, for the normal Kalman filter and the FLAT MMF. The motion is modeled with constant velocity. Two Kalman filters are used in the FLAT MMF and they are triggered at every 25 samples. From these results, we observe that the single filter almost losses track of the motion after

the first turn. On contrast, the FLAT MMF provides very good estimates even at the place having very sharp velocity change ( $180^\circ$  turn).

For the case that the particle moves three times faster, the estimation performance using the above FLAT MMF becomes worse, but the estimates are still fairly good. Figure 11a shows the estimate of  $X^{[1]}(t) / Z(0)$ . In order to improve the estimation, one may add more filters to the FLAT MMF. Figure 11b shows the estimate of  $X^{[1]}(t) / Z(0)$  from the FLAT MMF using five filters. Each filter is triggered at every ten samples.

#### **Experiment 4: undermodeling**

In this experiment, the particle moves on the plane  $Z = 20$  units, along an ellipse with angular velocity 0.15 radians/second. The lengths of the  $x$  and  $y$  axis of the ellipse are 12 and 8 units, respectively. The particle is at  $(6, 0, 20)$  units initially. Figure 12a shows the exact and noisy trajectories of the first 160 samples. The model with constant acceleration is used in modeling. This experiment is highly undermodeled because the circular motion requires very high order of power series to describe it accurately. Figures 12b and 12c show the estimate of velocity  $X^{[1]}(t) / Z(0)$  and the prediction of trajectory respectively, both for the normal Kalman filter and for the FLAT MMF with two filters triggered at every 25 samples. From these results, we observe that the estimation using one filter has a very large error while the estimation using FLAT MMF is quite good. Figure 13a and 13b depict the estimates of  $X^{[1]}(t) / Z(0)$  using the FLAT MMF with two and five filters respectively, for the case that the particle moves four times faster. These results show that even for such high-speed circular motion, FLAT MMF can still provide fairly good estimates and one may improve these estimates by adding more filters at the expense of more computation.

## **6. CONCLUSION**

In this report, we have shown that the estimation performance would degrade significantly if the motion model did not match to the actual trajectory. In order to solve such model mismatch problem, we proposed the FLAT MMF. Since the filters in FLAT MMF operate on different sets of past measurements, they can provide the estimates that contain the one that has good noisy suppression for the case of overmodeling or the case that the trajectory model matches to the actual trajectory, the one that contains small number or none of invalid measurements for the case of parameter jumping, and the one that has a small model error for the case of undermodeling. Also, the FLAT MMF uses the structure of Multiple Model Filter to combine these estimates optimally, in the sense of MMSE, so that the best



of them shows up at the final estimate. Thus, it is not surprise that the FLAT MMF is quite effective in suppressing the adversary effect due to the modeling error. Furthermore, all filters in the FLAT MMF can be run in parallel and the computational burden of combining the estimates from these filters to yield the final estimate is relatively small, i.e., the computational speed of obtaining the state estimate using FLAT MMF is similar to that using conventional filter.

## APPENDIX

### Theorem 1

$$(i) \quad \xi_{p, N(N+1)} = \begin{cases} 1 & p = 0 \\ \frac{(p!)^2}{(2p)!} \prod_{r=1}^p (N+r) & 0 < p \leq N \end{cases} \quad (3.19)$$

$$(ii) \quad \xi_{p, N(N)} = \begin{cases} 1 & p = 0 \\ \frac{(p!)^2}{(2p)!} \prod_{r=1}^p (N-r) & 0 < p \leq N \end{cases} \quad (3.20)$$

**Proof:** (i) We prove this claim by induction. From (3.15), it is easy to see that the claim in (3.19) is true for  $p = 0, 1, 2$ . Suppose this is true for  $p = m$ , where  $m \geq 2$ . Then

$$\xi_{m, N(N+1)} = \frac{(m!)^2}{(2m)!} \prod_{r=1}^m (N+r) \quad (a.1)$$

$$\xi_{m-1, N(N+1)} = \frac{((m-1)!)^2}{(2(m-1))!} \prod_{r=1}^{m-1} (N+r) \quad (a.2)$$

From the recursion relationship of the orthogonal polynomials in (3.16),

$$\begin{aligned} \xi_{m+1, N(N+1)} &= \xi_{1, N(N+1)} \xi_{m, N(N+1)} - \frac{m^2 (N^2 - m^2)}{4(4m^2 - 1)} \xi_{m-1, N(N+1)} \\ &= \frac{((m-1)!)^2}{(2m-2)!} \prod_{r=1}^{m-1} (N+r) \left[ \frac{N+1}{2} \frac{m^2(N+m)}{2m(2m-1)} - \frac{m^2(N+m)(N-m)}{4(2m+1)(2m-1)} \right] \\ &= \frac{((m-1)!)^2 (N+m)m}{4(2m-2)!(2m-1)(2m+1)} \prod_{r=1}^{m-1} (N+r) [(N+1)(2m+1) - m(N-m)] \\ &= \frac{(m!)^2}{2(2m+1)!} \prod_{r=1}^m (N+r) [(N+m+1)(m+1)] \\ &= \frac{((m+1)!)^2}{(2m+2)!} \prod_{r=1}^{m+1} (N+r) \end{aligned} \quad (a.3)$$

Hence, the claim in (3.19) is also true for  $p = m+1$ . Consequently, this claim is proved.

(ii) The proof is similar to that of (i). Q.E.D.

### Theorem 2

The  $p$ -th derivative of  $\xi_{p, N}(u)$  is given as

$$\xi_{p, N}^{[p]}(u) = p! \quad p \leq N. \quad (3.21)$$

For large  $u$  and  $m < p \leq N$ , the  $m$ -th derivatives of  $\xi_{p, N}(u)$  can be approximated as

$$\xi_{p, N}^{[m]}(u) \approx \frac{p!}{(p-m)!} u^{p-m}. \quad (3.22)$$

**Proof:** By using the factorial representation of orthogonal polynomial in (3.13), it is easy to see that  $\xi_{p, N}^{[p]}(u) = p!$  and, for large  $u$ , we may approximate  $\xi_{p, N}(u) \approx u^p$ . Note that the leading coefficient in (3.13)  $d_{p, p, N} = 1$ . Consequently, the  $m$ -th derivatives of  $\xi_{p, N}(u)$  is given by (3.22) and the theorem is proved. Q.E.D.

### Theorem 3

Let

$$V(p, N, u, m) = \frac{[\xi_{p, N}^{[m]}(u)]^2}{S(p, N)}, \quad (3.23)$$

then the followings are true.

$$(i) \quad V(p, N, N+1, 0) = \begin{cases} 1/N & p=0 \\ \frac{2p+1}{N} \prod_{r=1}^p \frac{N+r}{N-r} & 0 < p \leq N \end{cases}. \quad (3.24)$$

$$(ii) \quad V(p, N, N, 0) = \begin{cases} 1/N & p=0 \\ \frac{2p+1}{N} \prod_{r=1}^p \frac{N-r}{N+r} & 0 < p \leq N \end{cases}. \quad (3.25)$$

(iii)  $V(p, N, N+1, m) = 0$ , if  $p < m$ . Furthermore, for  $m \leq p \leq N$  and large  $N$ ,

$$V(p, N, N+1, m) \approx \frac{(2p)! (2p+1)!}{(p!)^2 ((p-m)!)^2} (N+1)^{-(2m+1)}. \quad (3.26)$$

(iv)  $V(p, N, N, m) = 0$ , if  $p < m$ . Furthermore, for  $m \leq p \leq N$  and large  $N$ ,

$$V(p, N, N, m) \approx \frac{(2p)! (2p+1)!}{(p!)^2 ((p-m)!)^2} N^{-(2m+1)}. \quad (3.27)$$

**Proof:** (i) From (3.18) and (3.15),  $\xi_{0, N}(u) = 1$  and  $S(0, N) = N$ , this implies that  $V(0, N, N+1, 0) = 1/N$ . Substituting (3.18) into (3.23), we have

$$V(p, N, u, m) = \frac{(2p)!(2p+1)!}{(p!)^4 \prod_{r=-p}^p (N-r)} [\xi_{p, N}^{[m]}(u)]^2. \quad (a.4)$$

From theorem 1 and for  $p > 0$ ,

$$V(p, N, N+1, 0) = \frac{(2p)!(2p+1)!}{(p!)^4 \prod_{r=-p}^p (N-r)} \frac{(p!)^4 \left[ \prod_{r=1}^p (N+r) \right]^2}{((2p)!)^2} \quad (a.5)$$

$$= \frac{(2p+1) \prod_{r=1}^p (N+r)}{\prod_{r=0}^p (N-r)}. \quad (a.6)$$

Consequently, the claim in (3.24) is followed.

(ii) The proof is similar to that of (i).

(iii) Since  $\xi_{p, N}(u)$  is a  $p$ -degree polynomial, the  $m$ -th derivatives of it, for  $m > p$ , is zero. Thus, from (3.23),  $V(p, N, N+1, m) = 0$ , for  $m > p$ . For large  $N$ ,  $\prod_{r=-p}^p (N-r) \approx (N+1)^{2p+1}$  and from theorem 2,  $\xi_{p, N}^{[m]}(N+1) = \frac{p!}{(p-m)!} (N+1)^{p-m}$ . Hence, from (a.4), we may approximate  $V(p, N, N+1, m)$  as in (3.26).

(iv) The proof is similar to that of (iii). Q.E.D.

#### Theorem 4

Two estimators,  $\hat{X}'_{m, J_0, J-1}(t)$  in (3.6) and  $\hat{X}_{m, J_0, J-1}(t)$  in (3.31), are identical. Their corresponding covariances in (3.8) and (3.32) are also the same.

**Proof:** For convenience, we rewrite the trajectory model in (3.1) and (3.28) as follows.

$$X(t) = \sum_{p=0}^{dm} \theta'_{p, J_0} \frac{(t - t_{J_0})^p}{p!} \quad (a.7)$$

$$= \sum_{p=0}^{dm} \theta_{p, J_0, J-1} \xi_{p, J-J_0}(u_{J_0}(t)). \quad (a.8)$$

From (3.13) and (3.29), (a.8) can be written as

$$f(t) = \sum_{p=0}^{dm} \theta_{p, J_0, J-1} \left[ d_{0, p, J-J_0} + \sum_{k=1}^p \frac{d_{k, p, J-J_0}}{\Delta^k} (t - J_0\Delta) (t - J_0\Delta - \Delta) \cdots (t - J_0\Delta - (k-1)\Delta) \right]. \quad (a.9)$$

Note that  $d_{p, p, J-J_0} = 1$ . Thus, by expanding and combining the terms, we can represent the parameter vector  $\underline{\theta}'$  in terms of  $\underline{\theta}$  as

$$\underline{\theta}' = \mathbf{B} \underline{\theta}, \quad (a.10)$$

where  $\underline{\theta}'$  and  $\underline{\theta}$  are defined in (3.2) and (3.33), respectively. The  $\mathbf{B}$  is a  $(dm+1) \times (dm+1)$  upper triangular matrix with diagonal elements  $B_{ii} = \frac{1}{\Delta^i}$ . Hence, the matrix  $\mathbf{B}$  is nonsingular, i.e., the inverse of  $\mathbf{B}$  exists. Thus, the parameters  $\{\theta'_{p, J_0}\}_{p=0}^{dm}$  and  $\{\theta_{p, J_0, J-1}\}_{p=0}^{dm}$  satisfy a unique and invertible linear transformation. Define

$$\underline{\mathbf{X}}(t) = [ X_s^{[0]}(t) \ X_s^{[1]}(t) \ \cdots \ X_s^{[dm]}(t) ]^T, \quad (a.11)$$

$$\hat{\underline{\mathbf{X}}}(t) = [ \hat{X}_{0, J_0, J-1}(t) \ \hat{X}_{1, J_0, J-1}(t) \ \cdots \ \hat{X}_{dm, J_0, J-1}(t) ]^T \quad (a.12)$$

Let  $\mathbf{V}(t)$  and  $\mathbf{G}(t)$  be the  $(dm+1) \times (dm+1)$  matrices with elements, respectively,  $V_{mn, J_0, J-1}(t)$  and

$$G_{mn}(t) = \frac{\partial X_s^{[m]}(t)}{\partial \theta_{n, J_0, J-1}}. \quad (a.13)$$

Let  $\underline{\beta}$  be a  $(dm+1) \times 1$  vector with elements

$$\beta_m = \frac{\partial \ln p ( \underline{M}_{J_0, J-1} ; \underline{\theta} )}{\partial \theta_{m, J_0, J-1}}. \quad (a.14)$$

$\hat{\underline{\mathbf{X}}}(t)$ ,  $\underline{\mathbf{V}}(t)$ ,  $\mathbf{G}'(t)$  and  $\underline{\beta}'$  can be defined similarly. Then, from (3.6), (3.31), (3.8) and (3.32), the estimators  $\hat{X}'_{m, J_0, J-1}(t)$  and  $\hat{X}_{m, J_0, J-1}(t)$  as well as their covariances can be put into the matrix form as follows.

$$\hat{\underline{\mathbf{X}}}(t) = \underline{\mathbf{X}}(t) + \mathbf{G}(t) \Gamma^{-1} \underline{\beta}, \quad (a.15)$$

$$\hat{\underline{\mathbf{X}}}'(t) = \underline{\mathbf{X}}(t) + \mathbf{G}'(t) \Gamma'^{-1} \underline{\beta}', \quad (a.16)$$

$$\mathbf{V}(t) = \mathbf{G}(t) \Gamma^{-1} \mathbf{G}^T(t), \quad (a.17)$$

$$\mathbf{V}'(t) = \mathbf{G}'(t) \Gamma'^{-1} \mathbf{G}'^T(t). \quad (a.18)$$

Using (a.10), it can be shown that  $\mathbf{G}'(t) = \mathbf{G}(t) \mathbf{B}$ ,  $\underline{\beta}'(t) = \mathbf{B}^T \underline{\beta}(t)$ , and  $\Gamma' = \mathbf{B}^T \Gamma \mathbf{B}$ . Then,  $\Gamma'^{-1} = \mathbf{B}^{-1} \Gamma^{-1} (\mathbf{B}^T)^{-1}$ . Thus,

$$\begin{aligned}
 \hat{\underline{X}}'(t) &= \underline{X}(t) + G(t) B [ B^{-1} \Gamma^{-1} (B^T)^{-1} ] B^T \underline{\beta} \\
 &= \underline{X}(t) + G(t) \Gamma^{-1} \underline{\beta} \\
 &= \hat{\underline{X}}(t) ,
 \end{aligned} \tag{a.19}$$

and

$$\begin{aligned}
 V'(t) &= G(t) B [ B^{-1} \Gamma^{-1} (B^T)^{-1} ] B^T G^T(t) \\
 &= G(t) \Gamma^{-1} G^T(t) \\
 &= V(t) .
 \end{aligned} \tag{a.20}$$

Consequently, the theorem is proved. Q.E.D

**Lemma 1**

- (i) The norm of  $\underline{b}_k(i)$  equals to one.
- (ii) The inner product of  $\underline{b}_k(i)$  and  $\underline{b}_l(i)$  equals approximately to one for large  $i$ .
- (iii) The inner product of  $\underline{b}_k(i)$  and  $\underline{X}_s(i)$  is given by

$$\underline{b}_k^T(i) \underline{X}_s(i) = \frac{- B_{0, a_k, i-1}(t_i) \sigma}{A_k^{1/2}(t_i)} , \tag{5.18}$$

where  $B_{0, a_k, i-1}(t_i)$  is the bias of the estimate  $\hat{X}_{0, a_k, i-1}(t_i)$ , using the measurement  $\{m(t_j)\}_{j=a_k}^{i-1}$ .

**Proof:** (i) From (5.13), the norm of  $\underline{b}_k(i)$  is given by

$$\begin{aligned}
 | \underline{b}_k^T(i) \underline{b}_k(i) |^2 &= \sum_{j=0}^i b_{kj}^2(i) \\
 &= \frac{\sigma^2}{A_k(t_i)} \left[ 1 + \sum_{j=a_k}^{i-1} [w_{0j, a_k, i-1}(t_i)]^2 \right] \\
 &= \frac{1}{[1 + V_{00, a_k, i-1}(t_i) / \sigma^2]} \left\{ 1 + \sum_{j=a_k}^{i-1} \left[ \sum_{p_1=0}^{dm} \frac{\xi_{p_1, i-a_k}(i-a_k+1) \xi_{p_1, i-a_k}(j-a_k+1)}{S(p_1, i-a_k)} \right] \right. \\
 &\quad \left. \times \left[ \sum_{p_2=0}^{dm} \frac{\xi_{p_2, i-a_k}(i-a_k+1) \xi_{p_2, i-a_k}(j-a_k+1)}{S(p_2, i-a_k)} \right] \right\} \\
 &= \frac{1}{[1 + V_{00, a_k, i-1}(t_i) / \sigma^2]} \left[ 1 + \sum_{p=0}^{dm} \frac{\xi_{p, i-a_k}^2(i-a_k+1)}{S(p, i-a_k)} \right] .
 \end{aligned} \tag{a.21}$$

The third equality of above expressions is obtained by using the orthogonal property of orthogonal polynomial. From (3.37), the last term is equal to one and the claim is proved.

(ii) For large  $i$ ,  $(i - a_k) \approx (i - a_l)$ . This implies that  $\underline{b}_k(i) \approx \underline{b}_l(i)$ . Thus,  $\underline{b}_k^T \underline{b}_l(i) \approx \underline{b}_k^T(i) \underline{b}_k(i) = 1$ .

(iii) Since  $\underline{b}_k^T(i)$  has elements  $b_{kj}(i)$  in (5.13), we have

$$\begin{aligned} \underline{b}_k^T(i) \underline{X}_s(i) &= \frac{\sigma}{A_k^{1/2}(t_i)} \left[ X_s(t_i) - \sum_{j=a_k}^{i-1} w_{0j, a_k, i-1}(t_i) X_s(t_j) \right] \\ &= \frac{\sigma}{A_k^{1/2}(t_i)} \left[ X_s^{[0]}(t_i) - E \left[ \hat{X}_{0, a_k, i-1}(t_i) \right] \right] \\ &= \frac{-B_{0, a_k, i-1}(t_i) \sigma}{A_k^{1/2}(t_i)}. \end{aligned} \tag{a.22}$$

Q.E.D.

### Theorem 5

$$E \left[ \alpha_{k,l}(i) \right] = \frac{1}{2} \ln \frac{A_l(t_i)}{A_k(t_i)} + \frac{1}{2 \sigma^2} \left[ \left( \underline{b}_l^T(i) \underline{X}_s(i) \right)^2 - \left( \underline{b}_k^T(i) \underline{X}_s(i) \right)^2 \right] \tag{5.19}$$

$$= \frac{1}{2} \ln \frac{A_l(t_i)}{A_k(t_i)} + \frac{1}{2} \left[ \frac{B_{0, a_l, i-1}^2(t_i)}{A_l(t_i)} - \frac{B_{0, a_k, i-1}^2(t_i)}{A_k(t_i)} \right], \tag{5.20}$$

$$\begin{aligned} \text{Var} \left[ \alpha_{k,l}(i) \right] &= 1 - \left( \underline{b}_k^T(i) \underline{b}_l(i) \right)^2 + \frac{1}{\sigma^2} \left[ \left( \underline{b}_k^T(i) \underline{X}_s(i) \right)^2 + \left( \underline{b}_l^T(i) \underline{X}_s(i) \right)^2 \right. \\ &\quad \left. - 2 \left( \underline{b}_k^T(i) \underline{b}_l(i) \right) \left( \underline{b}_k^T(i) \underline{X}_s(i) \right) \left( \underline{b}_l^T(i) \underline{X}_s(i) \right) \right] \end{aligned} \tag{5.21}$$

$$\approx \left[ \frac{B_{0, a_l, i-1}(t_i)}{A_l^{1/2}(t_i)} - \frac{B_{0, a_k, i-1}(t_i)}{A_k^{1/2}(t_i)} \right]^2. \tag{5.22}$$

In order to prove theorem 5, we first show the following lemma.

### Lemma 2

Let  $\underline{m}$  be an  $(i + 1) \times 1$  Gaussian random vector with probability density  $N(\underline{X}_s, \sigma^2 I)$ , where  $\underline{X}_s$  is an  $(i + 1) \times 1$  vector. Let  $B$  be an  $(i + 1) \times (i + 1)$  symmetric matrix and let the random variable

$$y = \underline{m}^T B \underline{m}. \tag{a.23}$$

Then the mean and the variance of  $y$  are given by

$$E [ y ] = \sigma^2 \text{Tr}(B) + \underline{X}_s^T B \underline{X}_s , \quad (\text{a.24})$$

$$\text{Var} [ y ] = 2\sigma^4 \text{Tr}(B^2) + 4\sigma^2 \underline{X}_s^T B^2 \underline{X}_s , \quad (\text{a.25})$$

where the symbol 'Tr' denotes the trace of a matrix.

**Proof of lemma 2:** Since  $B$  is symmetric, there exists an  $(i + 1) \times (i + 1)$  orthonormal matrix  $Q$  and an  $(i + 1) \times (i + 1)$  diagonal matrix  $\Lambda = \text{diag} ( \lambda_0, \lambda_1, \dots, \lambda_i )$ , such that

$$B = Q \Lambda Q^T , \quad (\text{a.26})$$

where  $\{ \lambda_p \}_{p=0}^i$  are the eigenvalues of  $B$ . Let  $\underline{x} = Q^T \underline{m} / \sigma$ . Obviously,  $\underline{x}$  is a Gaussian random vector with density  $N(\underline{\mu}, I)$ , where  $\underline{\mu} = Q^T \underline{X}_s / \sigma$ . Denote the  $p$ -th element of  $\underline{x}$  and  $\underline{\mu}$  as  $x_p$  and  $\mu_p$ , respectively, then  $\{ x_p^2 \}_{p=0}^i$  are independent noncentral  $\chi^2$  random variables with one degree of freedom and noncentrality parameter  $\{ \mu_p^2 \}_{p=0}^i$ . The mean and the variance of  $x_p^2$  are given by [Kotz82]

$$E [ x_p^2 ] = 1 + \mu_p^2 , \quad (\text{a.27})$$

$$\text{Var} [ x_p^2 ] = 2(1 + 2\mu_p^2) . \quad (\text{a.28})$$

From (a.23) and (a.26), we have

$$y = \sigma^2 \sum_{p=0}^i \lambda_p x_p^2 . \quad (\text{a.29})$$

Thus, the mean of  $y$  is given by

$$\begin{aligned} E [ y ] &= \sigma^2 \sum_{p=0}^i \lambda_p (1 + \mu_p^2) \\ &= \sigma^2 \left( \sum_{p=0}^i \lambda_p + \sum_{p=0}^i \lambda_p \mu_p^2 \right) . \end{aligned} \quad (\text{a.30})$$

Since  $\{ x_p^2 \}_{p=0}^i$  are independent , the variance of  $y$  is given by

$$\begin{aligned} \text{Var} [ y ] &= 2\sigma^4 \sum_{p=0}^i \lambda_p^2 (1 + 2\mu_p^2) \\ &= 2\sigma^4 \left( \sum_{p=0}^i \lambda_p^2 + 2 \sum_{p=0}^i \lambda_p^2 \mu_p^2 \right) . \end{aligned} \quad (\text{a.31})$$

From (a.26) and the definition of  $\underline{\mu}$ , the followings are true.

$$\text{Tr}(B) = \sum_{p=0}^i \lambda_p , \quad (\text{a.32})$$

$$\text{Tr}(\mathbf{B}^2) = \sum_{p=0}^i \lambda_p^2, \quad (\text{a.33})$$

$$\underline{\mathbf{X}}_s^T \mathbf{B} \underline{\mathbf{X}}_s = \sigma^2 \sum_{p=0}^i \lambda_p \mu_p^2, \quad (\text{a.34})$$

$$\underline{\mathbf{X}}_s^T \mathbf{B}^2 \underline{\mathbf{X}}_s = \sigma^2 \sum_{p=0}^i \lambda_p^2 \mu_p^2. \quad (\text{a.35})$$

Consequently, the claim in (a.24) and (a.25) are true. Q.E.D.

**Proof of theorem 5:** Let

$$\mathbf{B}(i) = \underline{\mathbf{b}}_l(i) \underline{\mathbf{b}}_l^T(i) - \underline{\mathbf{b}}_k(i) \underline{\mathbf{b}}_k^T(i), \quad (\text{a.36})$$

then, from (5.17), we have

$$\alpha_{kl}(i) = \frac{1}{2} \ln \frac{A_l(t_i)}{A_k(t_i)} + \frac{1}{2\sigma^2} \underline{\mathbf{m}}^T(i) \mathbf{B}(i) \underline{\mathbf{m}}(i). \quad (\text{a.37})$$

From the definition of  $\mathbf{B}(i)$  and using part (1) and (2) of lemma 1, the followings are true.

$$\text{Tr}(\mathbf{B}(i)) = 0 \quad (\text{a.38})$$

$$\text{Tr}(\mathbf{B}^2(i)) = 2 (1 - (\underline{\mathbf{b}}_k^T(i) \underline{\mathbf{b}}_l(i))^2) \quad (\text{a.39})$$

$$\approx 0 \quad (\text{a.40})$$

$$\underline{\mathbf{X}}_s^T(i) \mathbf{B}(i) \underline{\mathbf{X}}_s(i) = |\underline{\mathbf{b}}_l^T(i) \underline{\mathbf{X}}_s(i)|^2 - |\underline{\mathbf{b}}_k^T(i) \underline{\mathbf{X}}_s(i)|^2 \quad (\text{a.41})$$

$$\underline{\mathbf{X}}_s^T(i) \mathbf{B}^2(i) \underline{\mathbf{X}}_s(i) = |\underline{\mathbf{b}}_l^T(i) \underline{\mathbf{X}}_s(i)|^2 + |\underline{\mathbf{b}}_k^T(i) \underline{\mathbf{X}}_s(i)|^2 - 2 (\underline{\mathbf{b}}_k^T(i) \underline{\mathbf{b}}_l(i)) (\underline{\mathbf{b}}_k^T(i) \underline{\mathbf{X}}_s(i)) (\underline{\mathbf{b}}_l^T(i) \underline{\mathbf{X}}_s(i)) \quad (\text{a.42})$$

$$\approx [|\underline{\mathbf{b}}_k^T(i) \underline{\mathbf{X}}_s(i) - \underline{\mathbf{b}}_l^T(i) \underline{\mathbf{X}}_s(i)|^2] \quad (\text{a.43})$$

Using lemma 2, part (3) of lemma 1 and (5.7), the claims in this theorem are true. Q.E.D.

### Theorem 6

If  $i > a_k + d_m$  and  $i > a_1 + d_m$ , then the followings are true.

(i) if  $a_l > a_k$  then  $V_{00, a_l, i-1}(t_i) > V_{00, a_k, i-1}(t_i)$ ,  $V_{00, a_l, i}(t_i) > V_{00, a_k, i}(t_i)$  and  $A_l(t_i) > A_k(t_i)$ .

(ii) if  $a_l > a_k$  then  $V_{mm, a_l, i-1}(t_i) > V_{mm, a_k, i-1}(t_i)$  and  $V_{mm, a_l, i}(t_i) > V_{mm, a_k, i}(t_i)$ , for  $m > 0$  and large  $i$ .

**Proof:** (i) By applying the fact (i) of theorem 3 with  $N = i - a_k$  and  $m = 0$  to (5.3.4), we have

$$V_{00, a_k, i-1}(t_i) = \frac{\sigma^2}{i - a_k} + \sum_{p=1}^{d_m} \frac{(2p+1)\sigma^2}{i - a_k} \prod_{r=1}^p \frac{i - a_k + r}{i - a_k - r} \quad (\text{a.44})$$



Since  $a_1 > a_k$ ,  $(i - a_1) > dm$  and  $(i - a_k) > dm$ , we can show that  $\frac{1}{i - a_1} > \frac{1}{i - a_k}$  and

$\frac{i - a_1 + r}{i - a_1 - r} > \frac{i - a_k + r}{i - a_k - r}$ , for  $r = 1, \dots, dm$ . Consequently, from (a.44),  $V_{00, a_1, i-1}(t_i) > V_{00, a_k, i-1}(t_i)$ . From

(5.7), we have  $A_1(t_i) > A_i(t_i)$ . Similarly, we can prove that  $V_{00, a_1, i}(t_i) > V_{00, a_k, i}(t_i)$ .

(ii) Using the fact (iii) of theorem 3 with  $N = i - a_k$ , we can approximate  $V_{mm, a_k, i-1}(t_i)$  in (5.3.4), for  $m > 0$ , as

$$V_{mm, a_k, i-1}(t_i) = \frac{\sigma^2}{\Delta^{2m}} \sum_{p=m}^{dm} \frac{(2p)! (2p+1)!}{(p!)^2 ((p-m)!)^2} (i - a_k + 1)^{-(2m+1)} \quad (\text{a.45})$$

If  $a_1 > a_k$ , then  $(i - a_1 + 1)^{-(2m+1)} > (i - a_k + 1)^{-(2m+1)}$ . It implies that  $V_{mm, a_1, i-1}(t_i) > V_{mm, a_k, i-1}(t_i)$ . Similarly, we can prove that  $V_{mm, a_1, i}(t_i) > V_{mm, a_k, i}(t_i)$ . Q.E.D.

## Reference

- [Bog187]  
P.L. Bogler, "Tracking a maneuvering target using input estimation", IEEE Trans. Aerosp. Electro. System, vol. AES-23, May 1987, pp. 298-310.
- [Boll85]  
R.C. Bolles and H.H. Bakers, "Epipolar-plane image analysis: a technique for analyzing motion sequences", in Workshop on computer vision: representation and control, Oct. 1985.
- [Broi86a]  
T.J. Broida and R.Chellappa, "Estimation of object motion parameters from noisy images", IEEE PAMI, Jan 1986 , pp. 90-99.
- [Broi86b]  
T.J. Broida and R. Chellappa, "Kinematics and structure of a rigid object from a sequence of noisy images", IEEE workshop on motion 1986, pp. 95-100.
- [Chan84]  
C-B. Chang and J. A. Tabaczynski, "Application of state estimation to target tracking", IEEE Trans. Auto. Control, vol. AC-29, No. 2, Feb. 1984, pp. 98-109.
- [Chin79]  
L. Chin, "Advances in adaptive filtering", in *Advances in Control and Dynamic Systems*, Vol. 15, C. T. Leondes, Ed. New York: Academic 1979.
- [Gelb74]  
A. Gelb, Ed., *Applied Optimal Estimation*. Cambridge, MA: MIT Press, 1974.
- [Iu89a]  
S.-L. Iu and K. Wohn, "Estimation of 3-D motion and structure based on a temporally-oriented approach with the method of regression", IEEE Workshop on visual motion, March 1989, pp. 273-281.
- [Iu89b]  
S.-L. Iu and K. Wohn, "Recovery of 3-D motion of a single particle", SPIE Symposium on Machine Vision, November 1989.
- [Jazw70]  
A.H. Jazwinski, *Stochastic Processes and Filtering Theory*. New York: Academic, 1970.
- [Kotz82]  
S. Kotz and N.L. Johnson, *Encyclopedia of statistical Sciences*, New York: Wiley, 1982.
- [Kuma89]  
R.V.R. Kumar, A. Tirumalai and R. C. Jain, "A non-linear optimization algorithm for the estimation of structure and motion parameters", IEEE Workshop on visual motion, March 1989, pp. 136-143.
- [Mayb81]  
P.S. Maybeck, R.L. Jensen, and D.A. Harnly, "An adaptive extended Kalman filter for target image tracking", IEEE Trans. Aerosp. Electron. Syst., vol. AES-17, March 1981, pp. 172-180.
- [Mayb82]  
P. S. Maybeck, *Stochastic models, estimation and control*, vol. 1-2, Academic Press, 1982.
- [Magi65]  
D. T. Magill, "Optimal adaptive estimation of sampled stochastic process", IEEE Trans. Auto. Control, Oct. 1965, pp. 434-439.
- [Moos79]  
R.L. Moose, H.F. Vanlandingham and D.H. McCabe, "Modeling and estimation for tracking maneuvering targets", IEEE Trans. Aerosp. Electron. Syst., vol. AES-15, May 1979, pp. 448-456.
- [Peeb70]  
P. Z. Peebles, "Alternative approach to the prediction of polynomial signals in noise from discrete data", IEEE T-AES, July 1970, pp. 534-543.
- [Rals65]  
A. Ralston, *A first course in numerical analysis*, McGraw-Hill Co., 1965.
- [Simm86]  
Simmon Haykin, *Adaptive Filter Theory*, Prentice Hall, 1986
- [Sing70]  
R.A. Singer, "Estimating optimal tracking filter performance for manned-maneuvering targets", IEEE trans. Aerosp. Electron. Syst., vol. AES-6., July 1970, pp. 473-483.
- [Weng87]  
J. Weng, T.S. Huang and N. Akuja, "3-D motion estimation, understanding and prediction from noisy image sequences", IEEE PAMI, May 1987, pp. 370-389.

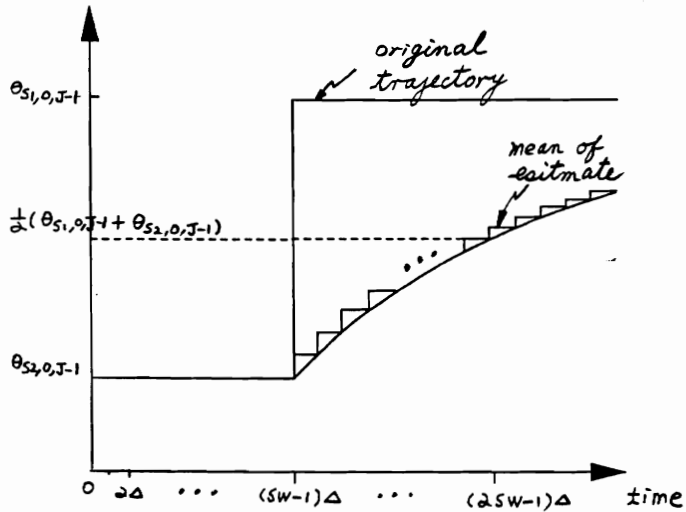


Fig. 1. Mean of the estimate  $\hat{X}_{0, J_0, J-1}(t_{J-1})$  and the original trajectory  $X_s(t)$  for example 1,  $J_0 = 0$ .

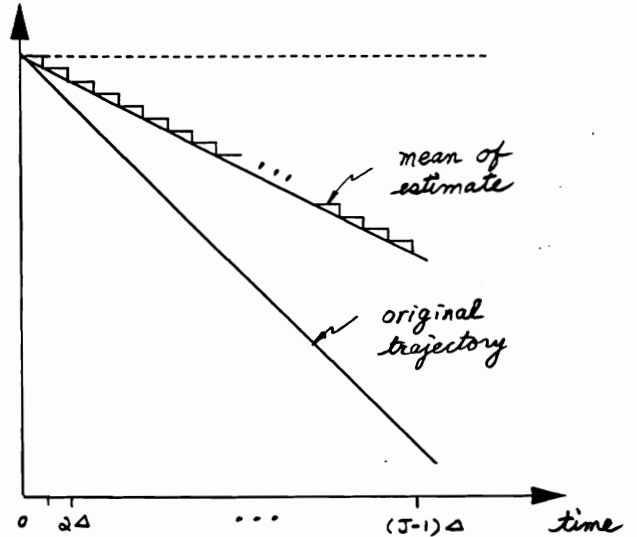


Fig. 2. Mean of the estimate  $\hat{X}_{0, J_0, J-1}(t_{J-1})$  and the original trajectory  $X_s(t)$  for example 2,  $J_0 = 0$ .

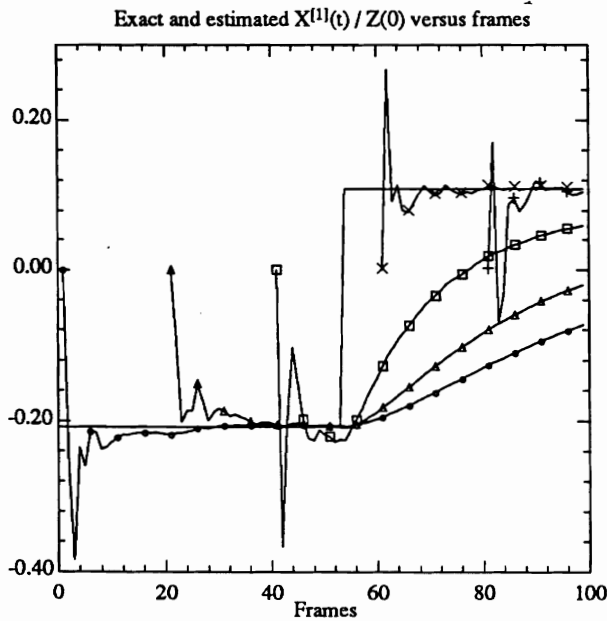


Fig. 3a. Exact and estimated  $X^{(1)}(t) / Z(0)$  versus number of frames for experiment 1. Estimators are triggered at samples 0 (O), 20 ( $\Delta$ ), 40 ( $\square$ ), 60 ( $\times$ ) and 80 (+).

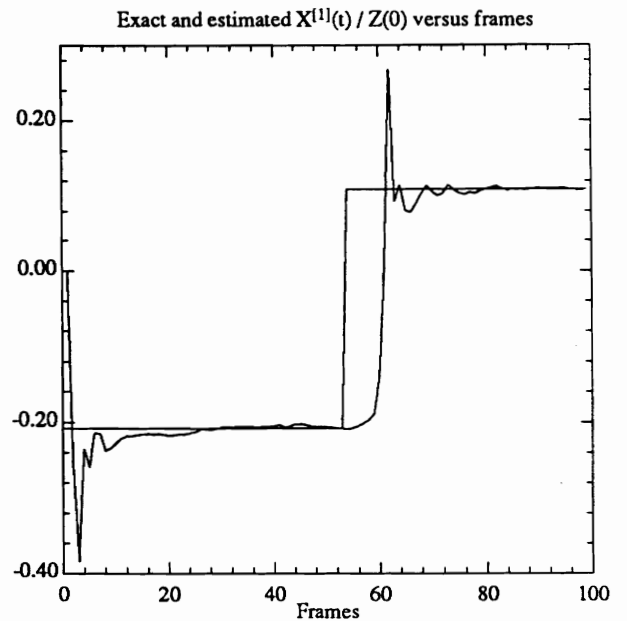


Fig. 3b. Exact and estimated  $X^{(1)}(t) / Z(0)$  versus number of frames by using FLAT MMF, for experiment 1.

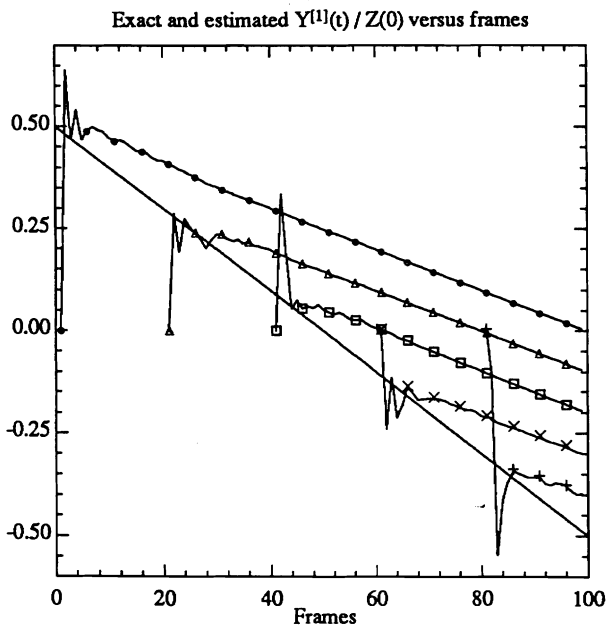


Fig. 4a. Exact and estimated  $Y^{(1)}(t) / Z(0)$  versus number of frames for experiment 2. Estimators are triggered at samples 0 (○), 20 (△), 40 (□), 60 (×) and 80 (+).

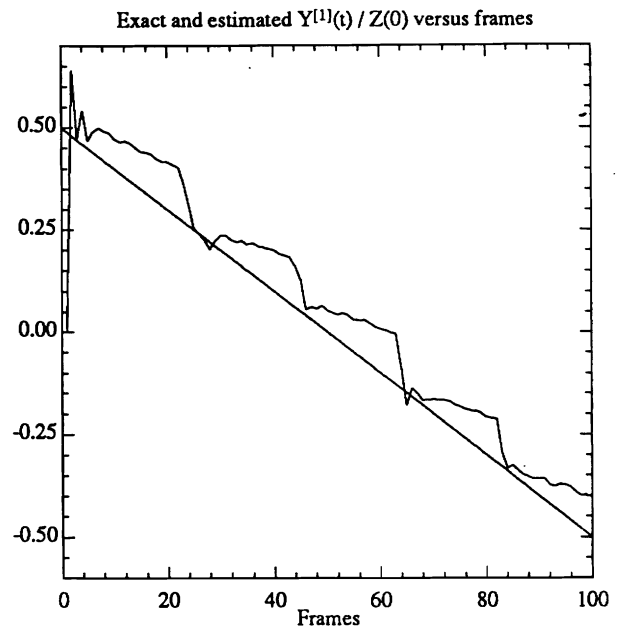


Fig. 4b. Exact and estimated  $Y^{(1)}(t) / Z(0)$  versus number of frames by using FLAT MMF, for experiment 2.

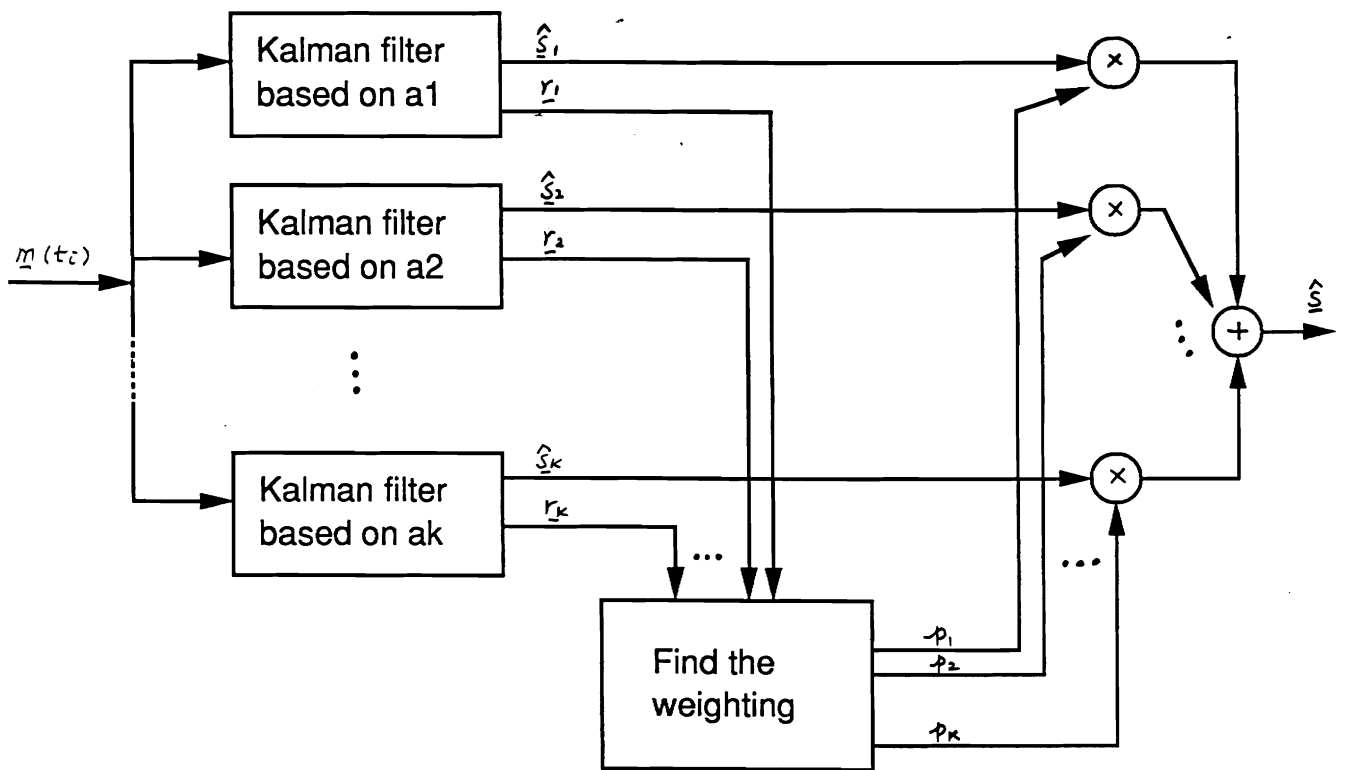


Fig. 5. Block diagram of multiple model filter.

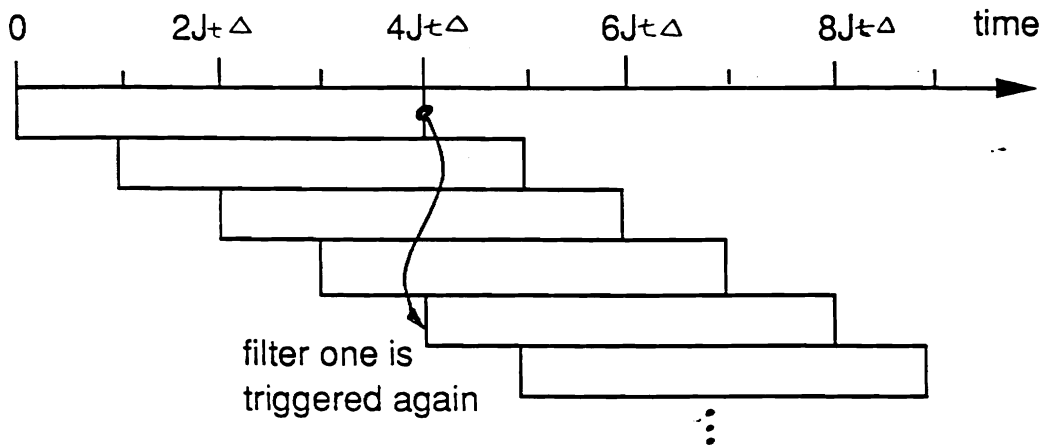


Fig. 6. Timing of FLAT MMF for  $K=4$ .

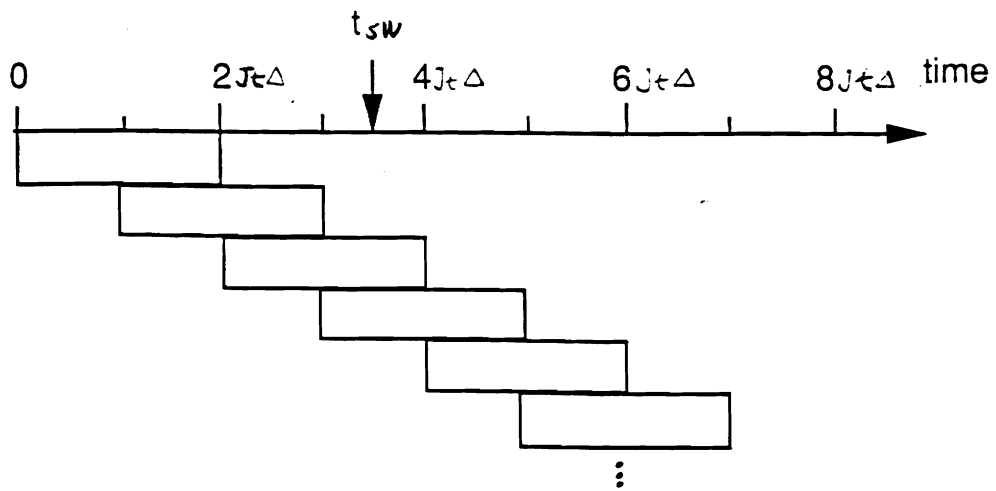


Fig. 7. Timing of the two parallelly operated filters and the switching time of the trajectory.

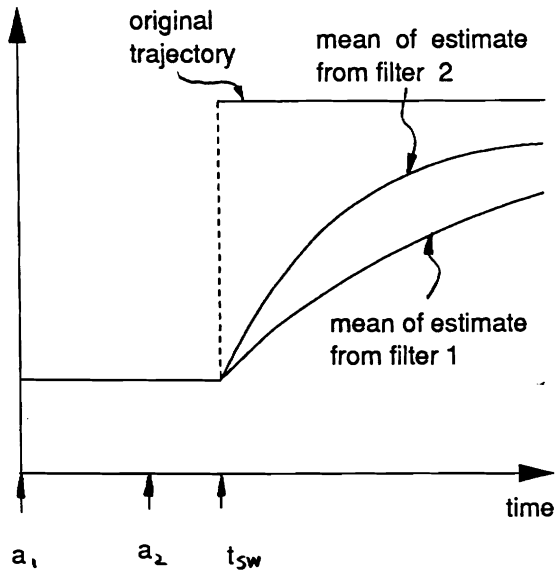


Fig. 8a. Example that filter 2 always provides better estimate for the analysis of FLAT MMF on parameter jumping,  $J_0 = 0$ .

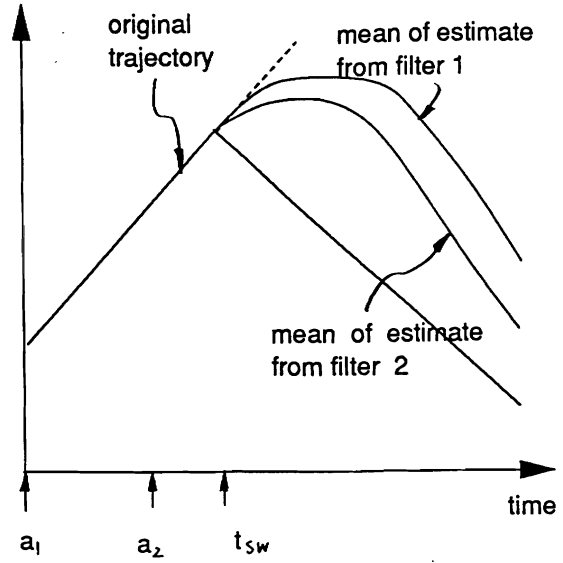


Fig. 8b. Example that filter 2 always provides better estimate for the analysis of FLAT MMF on parameter jumping,  $J_0 = 0$ .

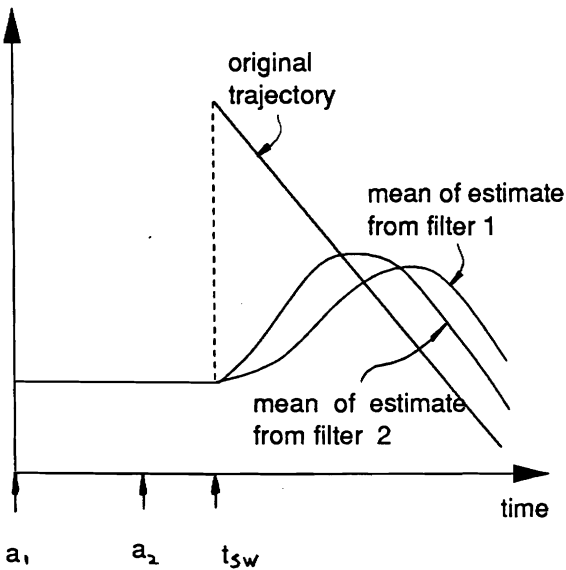


Fig. 8c. Counter-example that filter 2 always provides better estimate for the analysis of FLAT MMF on parameter jumping,  $J_0 = 0$ .

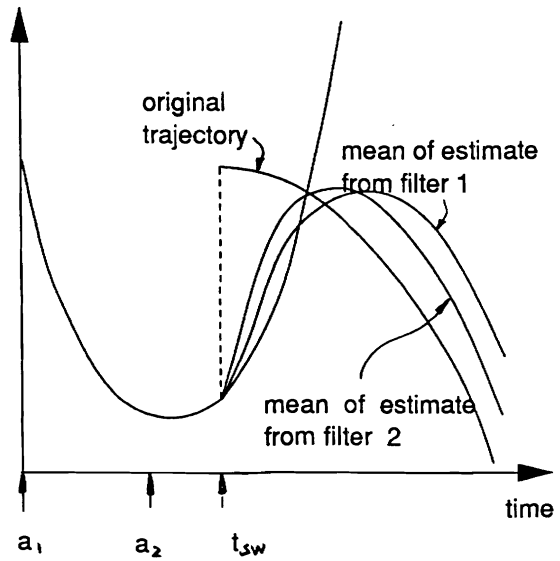


Fig. 8d. Counter-example that filter 2 always provides better estimate for the analysis of FLAT MMF on parameter jumping,  $J_0 = 0$ .

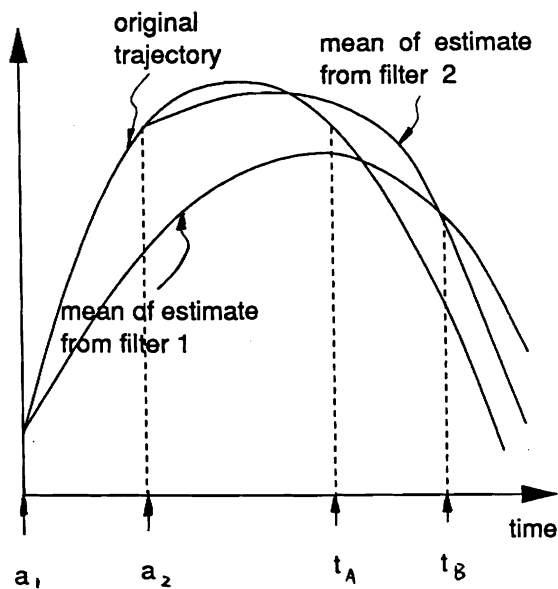


Fig. 9. Counter-example that filter 2 always provides better estimate for the analysis of FLAT MMF on undermodeling,  $d_m = 0$ ,  $d_s = 2$ ,  $J_0 = 0$ .

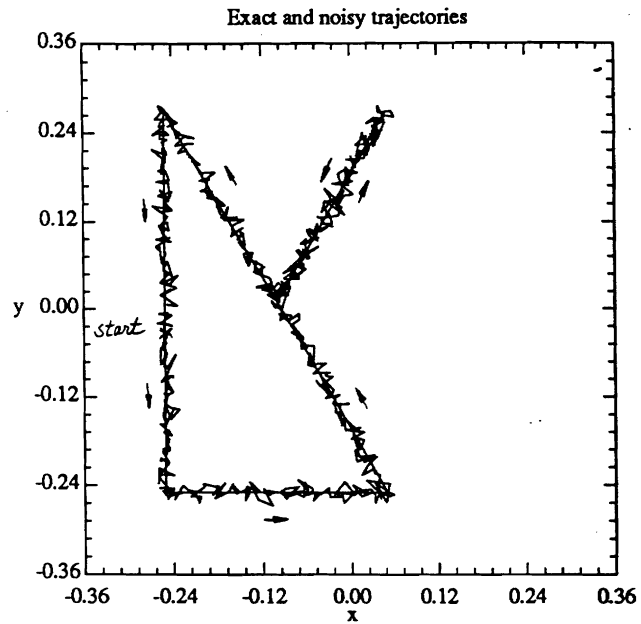


Fig. 10a. Exact and noisy trajectories for experiment 3. Standard deviation of the noise is 2.5 pixels.

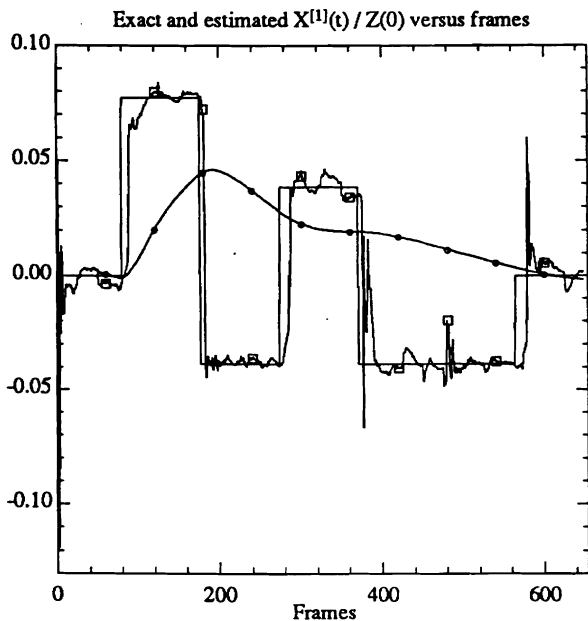


Fig. 10b. Exact and estimated  $X^{(1)}(t) / Z(0)$  versus number of frames by using normal filter (O) and FLAT MMF ( $\square$ ), for experiment 3.

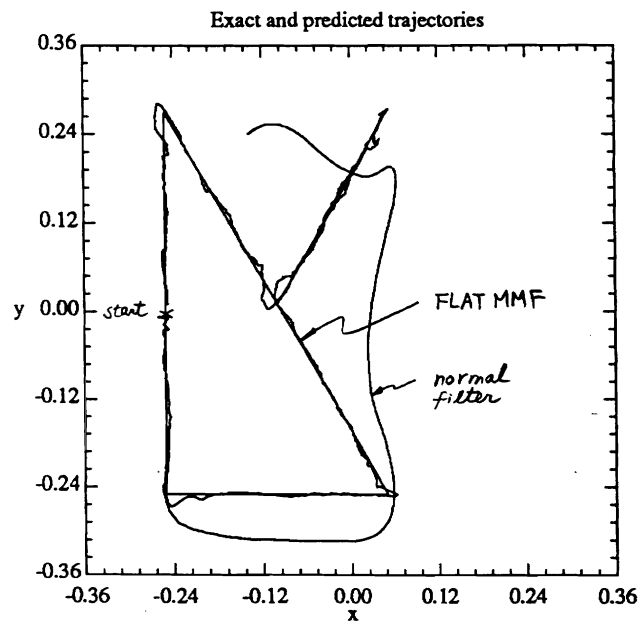


Fig. 10c. Exact and predicted trajectories by using normal filter and FLAT MMF, for experiment 3.

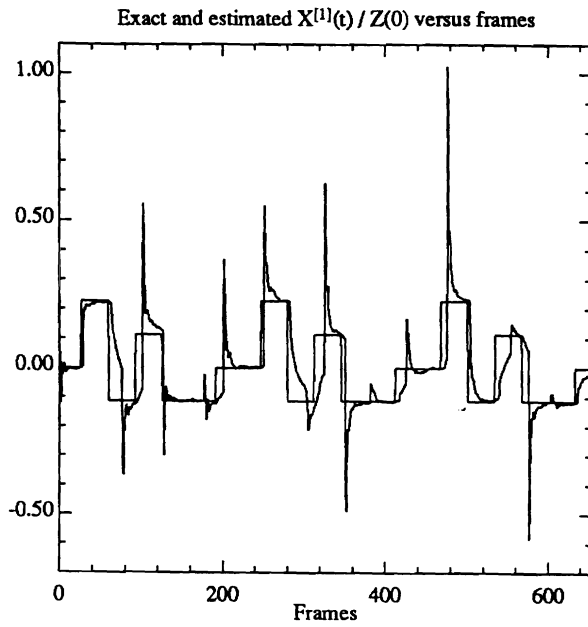


Fig. 11a. Exact and estimated  $X^{(1)}(t) / Z(0)$  versus number of frames by using FLAT MMF with two filters, for experiment 3 (faster motion).

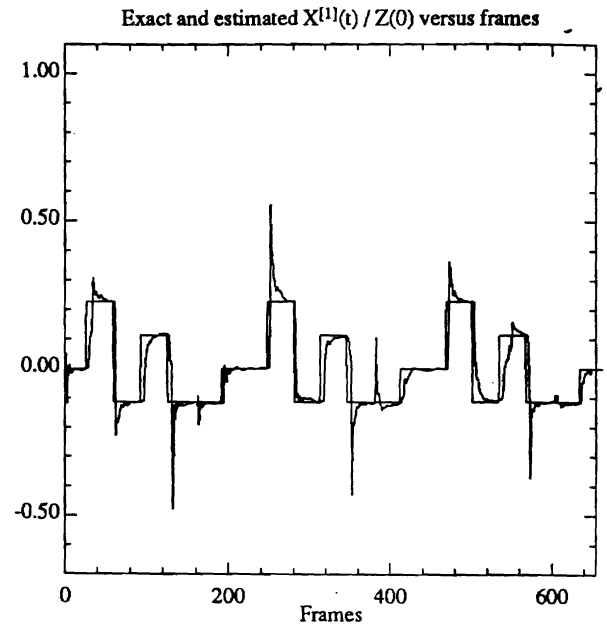


Fig. 11b. Exact and estimated  $X^{(1)}(t) / Z(0)$  versus number of frames by using FLAT MMF with five filters, for experiment 3 (faster motion).

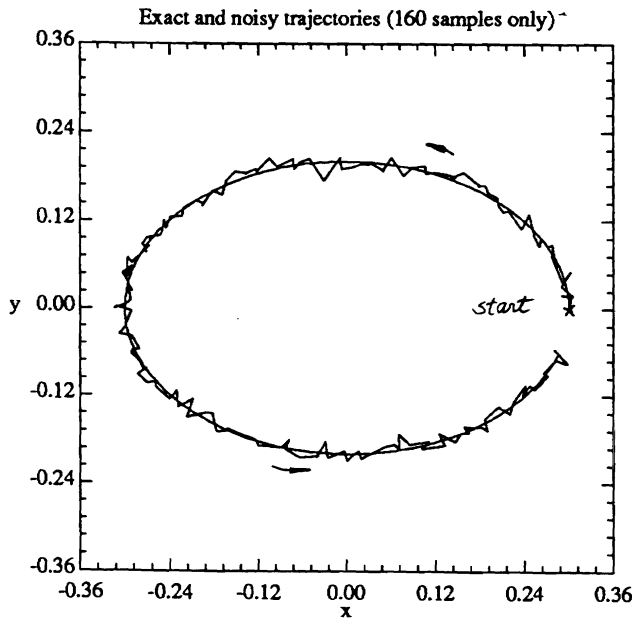


Fig. 12a. Exact and noisy trajectories for experiment 4. Standard deviation of the noise is 2.5 pixels.

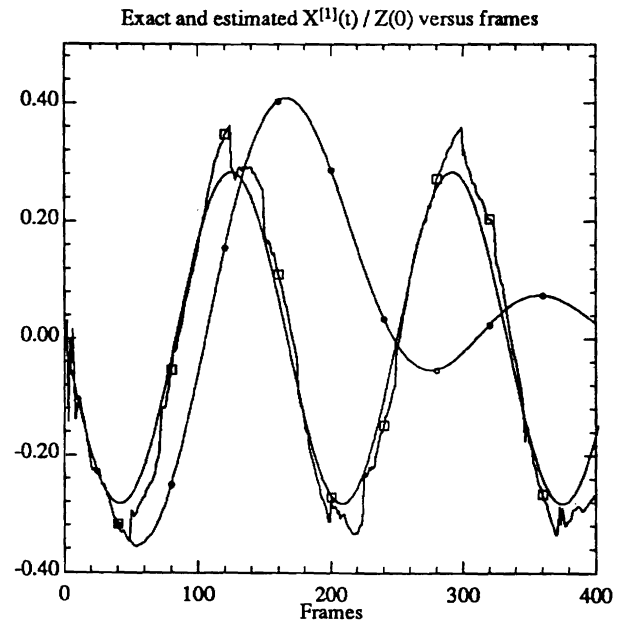


Fig. 12b. Exact and estimated  $X^{(1)}(t) / Z(0)$  versus number of frames by using normal filter (O) and FLAT MMF (□), for experiment 4.



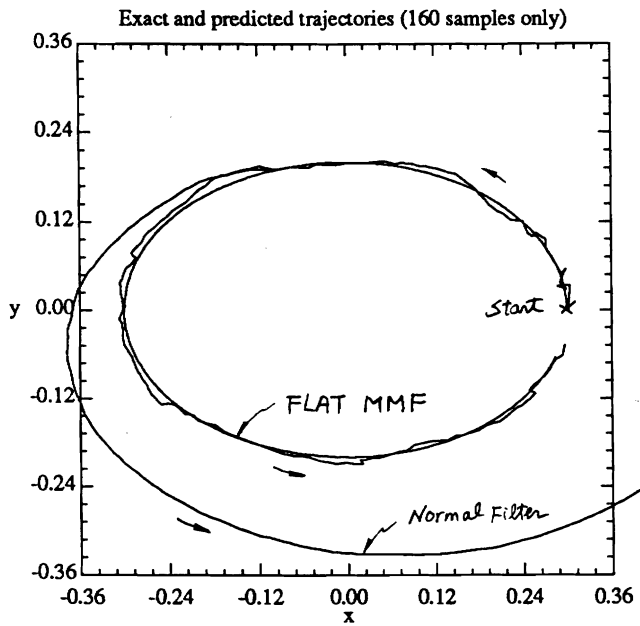


Fig. 12c. Exact and predicted trajectories by using normal filter and FLAT MMF, for experiment 4.

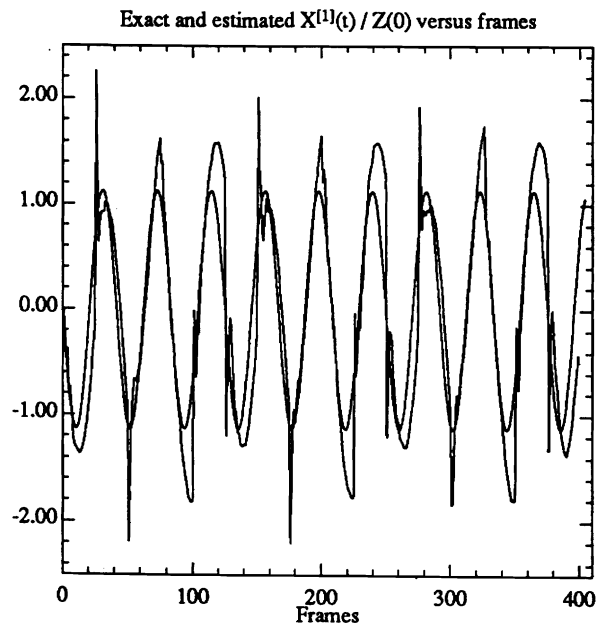


Fig. 13a. Exact and estimated  $X^{(1)}(t) / Z(0)$  versus number of frames by using FLAT MMF with two filters, for experiment 4 (faster motion).

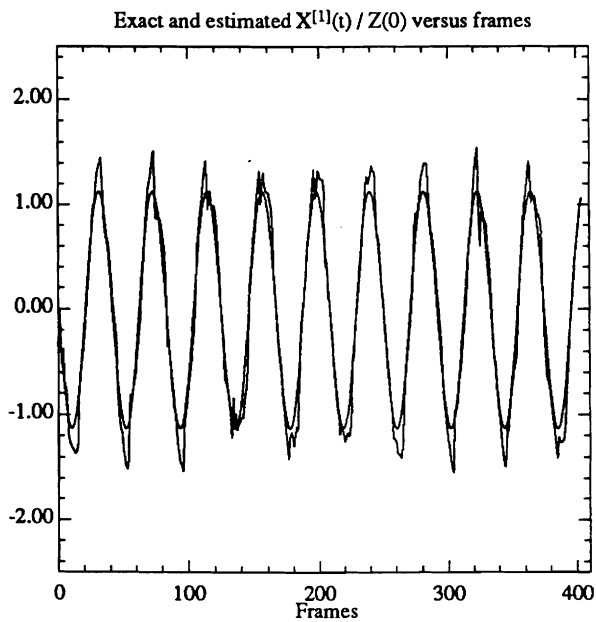


Fig. 13b. Exact and estimated  $X^{(1)}(t) / Z(0)$  versus number of frames by using FLAT MMF with five filters, for experiment 4 (faster motion).