

Article

Exploring the Exponentially Decaying Merit of an Out-of-Sequence Observation

Josiah Yoder ^{1,*}, Stanley Baek ² , Hyukseong Kwon ³ and Daniel Pack ⁴

¹ Department of Electrical Engineering and Computer Science, Milwaukee School of Engineering, Milwaukee, WI 53202, USA

² Department of Electrical and Computer Engineering, University of Michigan-Dearborn, Dearborn, MI 48128, USA; stanbaek@umich.edu

³ HRL Laboratories, Malibu, CA 90265, USA; hkwon@hrl.com

⁴ College of Engineering and Computer Science, University of Tennessee at Chattanooga, Chattanooga, TN 37403, USA; daniel-pack@utc.edu

* Correspondence: yoder@msoe.edu; Tel.: +1-765-444-9631

Received: 10 April 2018; Accepted: 11 June 2018; Published: 15 June 2018



Abstract: It is well known that in a Kalman filtering framework, all sensor observations or measurements contribute toward improving the accuracy of state estimation, but, as observations become older, their impact toward improving estimations becomes smaller to the point that they offer no practical benefit. In this paper, we provide a practical technique for determining the merit of an old observation using system parameters. We demonstrate that the benefit provided by an old observation decreases exponentially with the number of observations captured and processed after it. To quantify the merit of an old observation, we use the filter gain for the delayed observation, found by re-processing all past measurements between the delayed observation and the current time estimate, a high cost task. We demonstrate the value of the proposed technique to system designers using both nearly-constant position (random walk) and nearly-constant velocity (discrete white-noise acceleration, DWNA) cases. In these cases, the merit (that is, gain) of an old observation can be computed in closed-form without iteration. The analysis technique incorporates the state transition function, the observation function, the state transition noise, and the observation noise to quantify the merit of an old observation. Numerical simulations demonstrate the accuracy of these predictions even when measurements arrive randomly according to a Poisson distribution. Simulations confirm that our approach correctly predicts which observations increase estimation accuracy based on their delay by comparing a single-step out-of-sequence Kalman filter with a selective version that drops out-of-sequence observations. This approach may be used in system design to evaluate feasibility of a multi-agent target tracking system, and when selecting system parameters including sensor rates and network latencies.

Keywords: Kalman filter; out-of-sequence observation; delayed measurements; selective filtering; delayed Kalman gain

1. Introduction

In many estimation applications, sensor observations or measurements are received out-of-sequence (OOS); that is, the observations arrive at the filter in a different order than they were measured. A few examples include missile guidance with delayed measurements [1], multiple unmanned vehicles [2,3], teleoperation systems [4], and networked control systems [5]. Although standard filtering techniques (such as the Kalman filter) cannot be used directly when data arrive out-of-sequence, several techniques for incorporating these delayed observations have been developed. When a delayed observation arrives at a filter, an out-of-sequence filter can use that observation to improve the accuracy of the estimate.

There are a few reasons why it may be beneficial to simply drop some of the delayed estimates instead of processing them with the OOS filter. First, optimal out-of-sequence filters must re-process all previous estimates to find the current estimate. The more an OOS observation has been delayed, the longer it will take to process. Second, approximate OOS filters (those whose runtime is constant with respect to how delayed an observation is) provide no guarantee that the delayed observation will actually improve the performance. In fact, as our simulations show, delayed observations can actually hurt their performance! For these filters, dropping a delayed observation not only reduces processing time, it also has the potential to improve the accuracy of the filter's estimate.

1.1. Prior Work

Out-of-sequence observations can be incorporated optimally with the costly approach of re-processing all newer observations every time a delayed observation arrives. This requires storing all observations received in chronological order. In some cases, this is the only way to achieve this optimal result [6]. In other cases, additional assumptions can be made which simplify the out-of-sequence tracking problem [6]. For example, an algorithm proposed in [7,8] incorporates a single out-of-sequence observation optimally, assuming that the measurement time and covariance of the observation are known in advance. Even if nothing is known about the observation before it arrives, it can still be optimally fused using the A1 algorithm of [9], as long as the out-of-sequence observation arrives within the last sampling period (The last sampling period is the time between the second-most and most recently measured observations that have arrived at a filtering center).

Some out-of-sequence filters save processing time by providing an approximate rather than optimal estimate based on the out-of-sequence observations. The B1 and C1 algorithms from [9] are examples of these. Several tradeoffs between the complexity of both storage and computation and the accuracy of the algorithm are explored in [6], including a linearly constrained optimal estimate while storing only the latest state estimate with its covariance. Three new algorithms for the general out-of-sequence filtering problem are presented in [10]. For summaries of the current literature regarding filtering techniques with time delayed measurements, see [10–14].

While all out-of-sequence observations can theoretically be used to improve estimation accuracy with an optimal OOS filter, if a measurement is old enough it will have an insignificant contribution to the estimate of the current state. Under certain conditions, it has been demonstrated experimentally that observations may have no practical contribution to state estimation accuracy after as few as two to five time-steps [15,16]. In such cases, it is computationally more efficient to discard observations that would provide no practical contribution. Then, one may ask what factors determine whether an observation should be discarded. A variety of experiments have demonstrated that increasing randomness in an object's motion, decreasing observation noise, and increasing the amount of delay of an observation all decrease its contribution to estimation accuracy [15–17]. The experiments of [15] suggest that the impact of an observation decreases exponentially with its delay, but it is still obscure why the decrease would be exponential when the motion and measurement uncertainties increase only linearly or quadratically.

The experiments in [15,16] suggest that we should focus our attention on the ratio of an object's motion, relative to the observations that are made of it. (This dimensionless quantity is known as the maneuvering index [6,9,18,19]). These experiments show that the maneuvering index plays a critical role in determining the value of an out-of-sequence observation. To improve the estimation accuracy and stability of an OOS filter, OOS observations can be dropped if they will not contribute to the estimate. Tasoulis et al. propose an automatic technique for determining whether to keep an observation based on a hypothesis test comparing covariances fusing all measurements with maximum lag l and lag $l + 1$ [15]. The measurement is only fused if there is a significant improvement in the covariance according to a predetermined significance level. In [20], an approach is presented to determine which observations to incorporate to minimize tracking error within an average processing-time constraint.

1.2. Problem Statement

In summary, out-of-sequence observations are known to be of limited value if they have been delayed by too long. Dropping some old observations improves computational efficiency while maintaining nearly-optimal performance [15–17].

However, how do we determine when an observation has been delayed too long? Is there a way that we can gain some intuition about when we should drop an observation and when we should keep it? It is known that the value of an observation increases as the measurement time decreases, as its own observation noise decreases, as the target's motion becomes more predictable, and as the other measurements' observation noise increases. However, is there a way to incorporate these into a closed-form expression to estimate the impact an observation has on the accuracy of the current estimate? In this paper, we make a step toward this goal.

Our goal in this paper is to go beyond proposing yet another selective filter with an ad-hoc heuristic. We desire to provide a practical way of thinking about selective filtering that will help to explain why delayed observations behave the way they do.

1.3. Contributions

In particular, we would like to convince the reader that the usefulness (or merit) of a delayed observation decays exponentially with time. Traditionally, filters operate by correcting the current prediction of the state based on the error in the prediction of the observation (We explain the notation used here in Section 2.1), i.e., $\hat{x} = \hat{x}^p + K(y - H\hat{x}^p)$.

This allows us to reduce the problem to the estimation of an appropriate gain, K . Furthermore, the gain applied to a delayed observation by an ideal filter decreases exponentially with time as we demonstrate in Section 2. For the case where the maneuverability index is very low $\lambda \ll 1$, the magnitude of the gain K is proportional to $(1 - \lambda)^{T_d/\Delta T}$, where λ is the maneuvering index discussed previously in Section 1.1, T_d is the time delay of the observation, and ΔT is the mean time between observations. That is, the gain decreases exponentially with the time since the observation (measured in terms of how many measurements have been received since that time) and increases as the maneuverability decreases.

Unless an out-of-sequence approach models this exponentially-decaying nature of K (whether explicitly or implicitly), its performance will be hurt by old observations. As we demonstrate in Section 4, for an approximate filters, incorporating some delayed observations and dropping others can achieve better the estimation accuracy than would be achieved by incorporating either all or none of the delayed observations.

The key contributions of this paper are:

- We propose using the gain given by a simple optimal OOS filter to the delayed observation to estimate the merit of a past observation.
- For the case where the maneuverability index is very low $\lambda \ll 1$, this gain is proportional to $(1 - \lambda)^{T_d/\Delta T}$.
- We propose that observations where the ratio of the merit to a new observation is below a fixed threshold be dropped. For a very low maneuverability target moving according to a random walk, this merit leads to a threshold of $T_d = \frac{\Delta T p}{\lambda}$, that is, a threshold that increases linearly with the mean sampling rate and decreases linearly with the maneuverability of the target.
- Through synthetic simulations, we demonstrate that the proposed technique provides a good estimate of when observations no longer have merit across the full practical spectrum of the maneuverability index.

Although we provide an approximation of the ideal Kalman gain, we are not proposing a new OOS algorithm here. We intend this approach to be used with an existing OOS filters, although it can be used with any OOS filter available. This approach complements existing approaches by providing

an alternative perspective on the weight/gain that an old observation should be given, but it does not provide a technique for estimating the full matrix gain, only a bound on the elements within that matrix.

1.4. Assumptions

This work is focused on a Kalman filter that is operating in steady-state. During the initial stages, when the estimate covariances are still high, it could be desirable to incorporate more delayed observations or even all of them. An assumption we make in the derivation of the delayed gain (in Section 2) is that only a single observation has been delayed. During the simulations (in Section 4) we delay all observations randomly and discuss the impact this has on the derivation. Our simplest results apply in the case where $\lambda \ll 1$. A similar assumption is made in [20]. This case is particularly interesting for OOS filters, because the less maneuverable a target is, the more valuable out-of-sequence measurements become. Nevertheless, this approach does not require this assumption to be made—it can be applied for any value of the parameter λ .

2. The Delayed Kalman Gain

In this section, we propose a Kalman gain be used as a metric of the merit of a delayed observation. In particular, we use the gain that would be given to an observation if we applied the optimal but costly approach of re-processing all newer observations every time a delayed observation arrived (The “newer observations” here are all the observations *measured after* the OOS measurement which *arrived before* it). Although the delayed Kalman gain cannot be computed exactly in practice without storing and reprocessing all measurements [6], we can obtain a good estimate of its magnitude using a formula that we present later in this section.

2.1. Preliminaries

For a nearly-constant velocity model, the maneuvering index (or tracking index) is defined as $\lambda = \sigma_Q \Delta T^2 / \sigma_R$, where ΔT is the time between observations, σ_Q is the standard deviation of the process noise, and σ_R is the standard deviation of the measurement noise. Although the form of the maneuvering index depends on the modeling assumptions, it always includes the ratio σ_Q / σ_R [6,9,21].

We use a classic state transition and observation model:

$$\begin{aligned} x_{n+1} &= Fx_n + w_n & w_n &\sim \mathcal{N}(0, Q) \\ y_n &= Hx_n + v_n & v_n &\sim \mathcal{N}(0, R) \end{aligned}$$

where F is the state transition matrix, H is the observation matrix, w and v are uncorrelated Gaussian vectors with zero mean and covariances Q and R , respectively, x_n is the state at the n^{th} time-step, and y_n is the observation at time n . (When all subscripts in an equation are the same, we omit them for simplicity.) Given this formulation, the Kalman prediction and update steps are

$$\begin{aligned} \hat{x}_n^p &= F\hat{x}_{n-1} \\ P_n^p &= FP_{n-1}F^T + Q \\ \hat{x}_n &= \hat{x}_n^p + K_n(y_n - H\hat{x}_n^p) \\ K_n &= P_n^p H^T (HP_n^p H^T + R)^{-1} \\ P_n &= (I - K_n H)P_n^p \end{aligned}$$

where P is the covariance of the estimate \hat{x} and K_n is the Kalman gain. We use a super-script p to denote the predictions (e.g., \hat{x}^p) and the absence of superscript p to denote the updated estimates that include the most recent observations (e.g., \hat{x}).

In the remainder of the paper, we refer only to the estimates of the state, \hat{x} , and not the state’s true value, x . In the sections that follow, we represent \hat{x} by x , and \hat{x}^p by x^p .

2.2. The Output of the Kalman Filter as Weighted Sum of Observations

The estimate produced by the linear Kalman filter is a linear combination of the observations y_1, \dots, y_n with weights (or gains) determined by the system parameters. We can see this by expanding the recursive formulas for the estimated states, both predicted x^p and updated x ,

$$\begin{aligned}x_n^p &= Fx_{n-1} \\x &= Ky + (I - KH)x^p,\end{aligned}$$

to obtain

$$x_n = K_n y_n + (I - K_n H) F K_{n-1} y_{n-1} + (I - K_n H) F (I - K_{n-1} H) F K_{n-2} y_{n-2} + \dots \quad (1)$$

$$= K_n y_n + \sum_{j=1}^{n-1} \left\{ \prod_{i=0}^{j-1} [(I - K_{n-i} H) F] K_{n-j} y_{n-j} \right\} \quad (2)$$

where $n > 0$ and we assume the initial state estimate $x_0 = 0$. The estimate for step n takes the form of a linear combination of the inputs y_n, y_{n-1}, \dots, y_1 where term y_{n-j} is weighted by matrix $\prod_{i=0}^{j-1} ((I - K_{n-i} H) F) K_{n-j}$.

2.3. Steady-State Kalman Gain

To understand how the weight, $\prod_{i=0}^{j-1} ((I - K_{n-i} H) F) K_{n-j}$, changes as the number of observations j increases, we focus on the situation where Q and R are constant, as the K_n 's will converge to a limiting value K . (In Section 3, we show numerically that this is also a good approximation when Q changes because observations are made non-periodically.) In order to find this value, it is necessary to solve the discrete algebraic Riccati equation (DARE),

$$C^p = F(C^p - C^p H^T (H C^p H^T + R)^{-1} H C^p) F^T + Q, \quad (3)$$

where C^p is the steady-state covariance of the prediction.

In general, the solutions to the DARE correspond to the roots of a polynomial, and thus cannot be expressed in a closed form for a sufficiently high order system (e.g., [22,23]). Nevertheless, for many applications, the DAREs have analytic solutions, and we look at two examples in the following sections.

The expanded state estimate (2) simplifies considerably when substituting the value of K after convergence for all the K_n 's, to

$$x_n = \sum_{j=0}^{n-1} ((I - KH) F)^j K y_{n-j} = \sum_{j=0}^{n-1} K_{n,j} y_{n-j} \quad (4)$$

where $K_{n,j}$ is the effective gain of the Kalman filter for all past observations. The effective gain $K_{n,j}$ is usually smaller than the original gain K_{n-j} applied to an observation because it is reduced by the down-weighting factor $((I - KH) F)^j$. In other words, although the ideal Kalman filtering may originally apply a high gain to an observation when it is first made, it is multiplied by gains that reduce its magnitude with each successive observation that is measured after it.

The simple linear combination (4) includes the gains that the ideal Kalman filter gives to all observations when the filter is in steady-state. This is also the gain that the optimal but costly buffered and re-filtered approach gives to a delayed observation. From (4), the gain for a delayed observation is

$$K_{n,j} = ((I - KH) F)^j K$$

where the observation is delayed by j time-steps. Given only the time-delay of an observation, T_d , we can compute the number of time-steps $j = T_d/\Delta T$ between when the observation was made and our most recent estimate time, and estimate the time-delayed gain

$$K_n(T_d) = ((I - KH)F)^{T_d/\Delta T} K, \quad (5)$$

where ΔT is the time between observations. This gain, $K_n(T_d)$ is the metric that we propose for the merit of a delayed observation. We recommend selecting delayed observations whose $K_n(T_d)$ value is greater than some threshold, and rejecting the rest.

In general, $K_n(T_d)$ is a matrix quantity, and requires the solution to the discrete algebraic Riccati equation. In Sections 2.5 and 2.6, we examine this gain for random walks and nearly-constant-velocity walks. Based on the decaying exponential bound for the function(s) making up the closed-form solution for $K_n(T_d)$ in these cases, we suggest how to select a time-delay threshold to decide whether to select a delayed observation to be included into an out-of-sequence filter. Although the general $K_n(T_d)$ cannot always be found analytically, it is still bounded by a decaying exponential envelope, as examined in the next section.

2.4. Exponential Behavior of the Time Delayed Gain $K_n(T_d)$

The time-delayed gain $K_n(T_d) = ((I - KH)F)^{T_d/\Delta T} K$ has norms bounded by a decaying exponential function because the magnitude of all eigenvalues μ of $(I - KH)F$, is less than one ($|\mu| < 1$), for all systems where (F, H) is observable and (F, Q) is reachable. As a result, the impact of a time-delayed observation will decrease exponentially as the delay time increases. Numerical methods can be used to determine the eigenvalues of $(I - KH)F$, and the largest eigenvalue (the one with magnitude closest to unity) will play the dominant role in determining the maximum delay time.

The time-delayed gain $K_n(T_d)$ provides insights into the merit of a time-delayed observation based on the system parameters F, Q, H , and R as well as the delay time, T_d . In the general case, these variables are hidden in the solution to the DARE, but in many practical cases, there are closed-form solutions that provide further insights, as we shall show in the remainder of this section. In the general case, $K_n(T_d)$ can be used to decide which time-delayed observations are worth incorporating using an out-of-sequence filter.

2.5. Scalar Case

In many practical cases, it is possible to find a closed-form solution to the discrete algebraic Riccati equation (DARE). In the scalar case, when $f = 1$ and $h = 1$, the limiting value of the steady-state gain k (sometimes known as α) is given by [18]

$$k = \frac{1}{2}(-\lambda^2 + \sqrt{\lambda^4 + 4\lambda^2}) \quad (6)$$

where $\lambda = \sigma_Q \sqrt{\Delta T} / \sigma_R$ is the maneuvering index for a continuous random walk (also known as a Brownian motion or Wiener process), σ_Q is the standard deviation of the state transition noise (in physical dimensions [length]/sqrt([time])), and σ_R is the standard deviation of the observation noise (in physical dimensions [length]). All other lower-case variables are the one-dimensional counterparts of the capitalized full-matrix versions.

With this k , we can compute the gain of the delayed observation using the scalar form of (5)

$$k_n(T_d) = (1 - k)^{T_d/\Delta T} k \quad (7)$$

Supposing that we only want to keep observations with a minimum gain of k_{min} , we can compute the maximum delay time to be

$$T_d = \Delta T \frac{\log(k_{min}/k)}{\log(1 - k)} \quad (8)$$

We use this equation in the experiments (in Section 4) to decide when to simply drop an old observation, saving computing time with only minor change in estimation performance.

When the motion model is very good (that is, when $\sigma_Q \ll \sigma_R$ and therefore the maneuvering index is small, $\lambda \ll 1$), the approximations $k \approx \lambda$ applies, and $k_n(T_d) = (1 - \lambda)^{T_d/\Delta T} k$. This knowledge of the low maneuvering index is significant for system design. For example, suppose that we choose to keep observations where $k_{min}/k \geq 0.6$, and suppose that the maneuvering index is $\lambda = 0.1$. Then we will keep all observations whose maximum delay is $T \geq T_d = \Delta T \frac{\log 0.6}{\log(1-0.1)} = \Delta T 4.85$, or about five time-steps.

2.6. Discrete White-Noise Acceleration (DWNA) Case

The DARE can also be solved in a closed-form for the common system models used to estimate positions and velocities, often called the constant velocity, the nearly-constant velocity, or the discrete white-noise acceleration model. In this model, $H = [1 \ 0]$, $F = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}$, $Q = \begin{bmatrix} \Delta T^4/4 & \Delta T^3/2 \\ \Delta T^3/2 & \Delta T^2 \end{bmatrix} \sigma_Q^2$, and $R = \sigma_R^2$. The steady-state Kalman gain is a function of the maneuvering index $\lambda = \sigma_Q \Delta T^2 / \sigma_R$, ([19,21]),

$$K = \begin{bmatrix} \alpha \\ \beta/\Delta T \end{bmatrix} \quad (9)$$

where

$$\alpha = \frac{1}{8}(-\lambda^2 - 8\lambda + (\lambda + 4)\sqrt{\lambda^2 + 8\lambda}) \quad (10)$$

and

$$\beta = \frac{1}{4}(\lambda^2 + 4\lambda - \lambda\sqrt{\lambda^2 + 8\lambda}) \quad (11)$$

The general form, $K_n(T_d) = ((I - KH)F)^{T_d/\Delta T} K$ reduces to

$$K_n(T_d) = \begin{bmatrix} 1 - \alpha & \Delta T(1 - \alpha) \\ -\beta/\Delta T & 1 - \beta \end{bmatrix}^j \begin{bmatrix} \alpha \\ \beta/\Delta T \end{bmatrix} \quad (12)$$

The matrix $(I - KH)F$ has eigenvalues

$$1 - \frac{\alpha + \beta}{2} \pm \frac{1}{2}\sqrt{(\alpha + \beta)^2 - 4\beta} \quad (13)$$

and as stated in Section 2.4, both eigenvalues have magnitude less than 1, and the matrix gain has a decaying exponential envelope.

The random-walk and position-and-velocity cases are easily extended to multiple dimensions. As long as there is no mixing between the dimensions, the formulas from these sections can be applied to each dimension independently.

3. Numerical Validation

In many applications where observations are delayed, the observations are also made asynchronously.

To validate the gain formulas provided in the previous section when observations are not made according to a fixed schedule, we compute these gains numerically in a scenario where observations occur according to a Poisson process. Our simulation uses a Kalman filter that maintains the delayed gain $k_{j,l}$ and predicted delayed gain $k_{j,l}^p$ of all observations. We computed these gains using Algorithm 1.

Algorithm 1 Computation of ideal Kalman filter gains on Poisson-distributed sequence. The gains do not depend on the actual measurements, so there is no need to include them in the calculations.

```

 $k_{1,1}^p = 1$ 
for  $j$  from 1 to  $n$ 
   $k_{j,0} = c_j^p / (h^2 c_j^p + \sigma_R^2)$ 
   $c_j = (1 - k_j h) c_j^p$ 
   $k_{j,l} = (1 - k_j h) k_{j,l}^p, l \in 1, \dots, j$ 
   $c_{n+1}^p = c_n + \Delta T_n \sigma_Q^2$ 
   $k_{j+1,l}^p = k_{j,l+1}, l \in 0, \dots, j$ 
end

```

This algorithm tracks the gain (or weight) of each observation on the current estimate. The estimate x_n can be expressed as $x_n = \sum_{l=0}^{n-1} k_{n,l} y_{n-l} + k_{n,n} x_0$ where $k_{n,l}$ is the gain of the observation delayed by l time-steps on the estimate at time n , and the prediction x_n^p can be expressed as $x_n^p = \sum_{l=0}^{n-1} k_{n,l}^p y_{n-l} + k_{n,n}^p x_0$, where $k_{n,l}^p$ is the gain of the observation delayed by l time-steps on the prediction at time n . This code is used to experimentally calculate the true delayed gains that would have been applied to an estimate. Because the gains do not depend on the actual observations, there is no need to compute the observations.

When the observations are uniformly spaced, our filter computes the delayed Kalman gain $k_n(T_d)$ exactly. When observations are not uniformly spaced, the state transition noise $\Delta T_n \sigma_Q^2$ will change from time-step to time-step, and k and $k_n(T_d)$ will never converge exactly.

To apply (7) to Poisson-distributed observations, we simply compute the expected number of observations j that would have arrived in the time period T since the observation came in:

$$j \approx T / \Delta T_p$$

where the rate parameter of the Poisson distribution is $1 / \Delta T_p$. In this case, the average time between observations will be ΔT_p in the place of a fixed time ΔT .

Figures 1 and 2 illustrate the result of experiments comparing $K_n(T_d)$ computed with a fixed ΔT with the actual gains computed for varying ΔT when observations arrive according to a Poisson distribution. Figure 1 illustrates a situation in which the process noise is significantly lower than the observation noise ($\sigma_Q \ll \sigma_R$). In this situation, we can use a simpler form of k to get a very good approximation of the weight of each observation. The figure shows the result of four example runs in this situation.

When σ_Q is closer to σ_R , the approximation is still fairly accurate. We examined this in the experimental runs illustrated in Figure 2. We show the result of numerous runs. Because of variations in the timing introduced by the Poisson distribution, the delayed Kalman gain is not always the same. Nevertheless, $K_n(T_d)$ as computed in (5) with a fixed ΔT gives a good approximation of the actual gain given to delayed observations when they are processed in-order.

Because $K_n(T_d)$ gives a good estimate of the gain applied to an old observation even when observations are not measured periodically, it can be applied to a wide variety of systems when determining the cut-off time threshold for out-of-sequence observations.

Figure 3 shows the result of a similar experiment using a discrete white-noise acceleration model instead of a random walk.

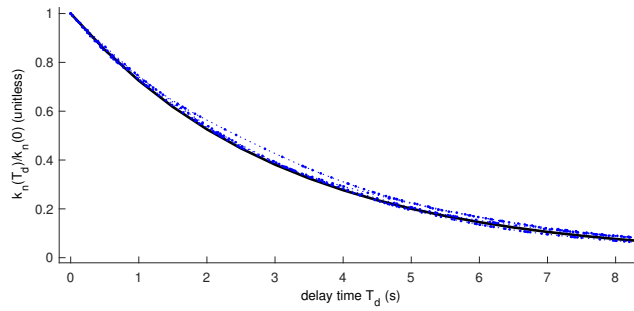


Figure 1. Low- λ approximation. Dashed blue lines with dots: Four experimental runs. Thick black line: $(1 - \frac{\sigma_Q \sqrt{\Delta T_P}}{\sigma_R})^{T/\Delta T_P}$, where $\Delta T_P = 0.1, \sigma_Q^2 = 0.01, \sigma_R^2 = 1, h = 1, f = 1$.

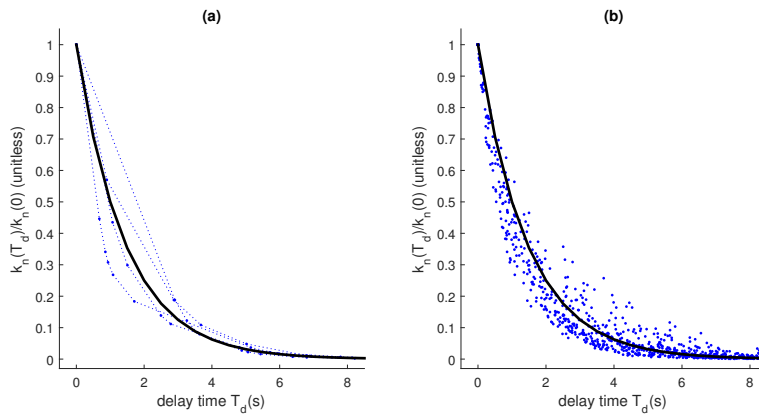


Figure 2. Typical- λ approximation. Dashed blue lines/dots: many runs. Thick black line: $(1 - k_n(\Delta T_P) h)^{T_d/\Delta T_P}$, where $k_n(T_d)$ is given by (6), $\lambda = 1, \sigma_Q^2 = 0.5, \sigma_R^2 = 1, h = 1, f = 1$ (a) five runs; (b) 100 runs displayed without connecting lines.

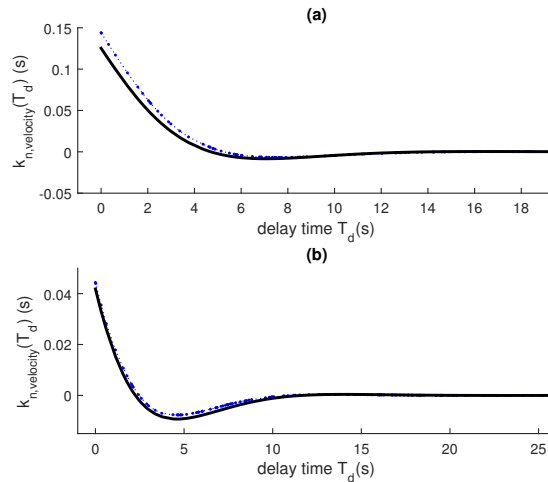


Figure 3. Approximation with a velocity component. Dashed blue lines/dots: One run. Thick black line: $(1 - KH)^{T/\Delta T} K$, where K is found by solving the DARE. $\Delta T = 1/5, Q = 0.01 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, R = 1, H = [1 \ 0], F = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}$. (a) position; (b) velocity.

4. Simulations and Discussion

4.1. Nearly-Constant Position (Random Walk)

The gain $K_n(T_d)$ not only provides intuition into the exponentially-decreasing value of a time-delayed observation but also can be used to set the time-delayed threshold, for example, using (8) from Section 2.5.

In this section, we run a series of experiments demonstrating that the threshold given by (8) accurately predicts which delayed observations contribute to an out-of-sequence filter's performance.

We use a simple Kalman filter (KF) implementation that drops stale observations, an out-of-sequence filter (Bar-Shalom et al.'s A/I [24]), and a *selective* A/I (SA/I) under a variety of operating conditions. For each run, we first produced N random steps of a Poisson process with rate parameter $1/\Delta T_p$. At each event in the Poisson process, we simulated a linear Markov process where the noise added between the steps is proportional to the time elapsed ($w \sim \mathcal{N}(0, \Delta t \sigma_Q^2)$). We then simulated observations delayed according to an exponential distribution with mean ΔT_D . The observations have Gaussian noise with covariance r added to them.

We processed these observations with the three filters mentioned above. The KF drops all late observations, the A/I incorporates all observations, and the SA/I selectively incorporates observations based on the threshold T_{stale} (Algorithm 2). The SA/I filter incorporates an observation if its time delay satisfies $T_d = t_e - t_m < T_{stale}$, where t_e is the time of the most recent estimate, and t_m is the measurement time of the delayed observation. When the SA/I incorporates out-of-sequence measurements, it uses the A/I technique.

Algorithm 2 A selective filter uses an existing out-of-sequence filter to incorporate observations that are newer than a fixed threshold.

```

field  $T_d$ 
field oosFilter
field lastEstTime
function init( $T_d$ ):
    field.T_d =  $T_d$ 
end
function update(time, observation):
    if  $time - lastEstTime < T_d$ :
        oosFilter.update(time, observation)
        lastEstTime = time
    end
end

```

We compare the estimates produced by each of the filters at each of the N time-steps with the true state at the filter's estimate time. The three filters use exactly the same estimate time—the maximum measurement time of the observations currently received. When an out-of-sequence measurement comes in, all the filters keep their current estimate time instead of going back to the measurement time of the out-of-sequence observation. They either update their estimate by incorporating the out-of-sequence observation, or simply keep their previous estimate if they choose to drop the out-of-sequence observation. For each experiment, we performed ten runs of 5000 steps and computed the root mean square error (RMSE) of each method for each run. When computing the RMSE error, we compared the estimates with the true states at the same time.

Using (8) with a k_{min}/k value of 0.6, we compute the threshold for keeping the observations shown in each figure. The value of 0.6 was selected experimentally, and shows a good match across the wide range of maneuvering indexes used here.

Figure 4 illustrates the results of these experiments. In each case, the value of the cut-off time T_{stale} accurately predicts the threshold where the performance of the SA/I filter meets the performance of

the A/I filter, even though this threshold changes by two orders of magnitude across the examples considered. This illustrates that the time-delayed gain we propose accurately determines the threshold for selecting the most useful delayed observations.

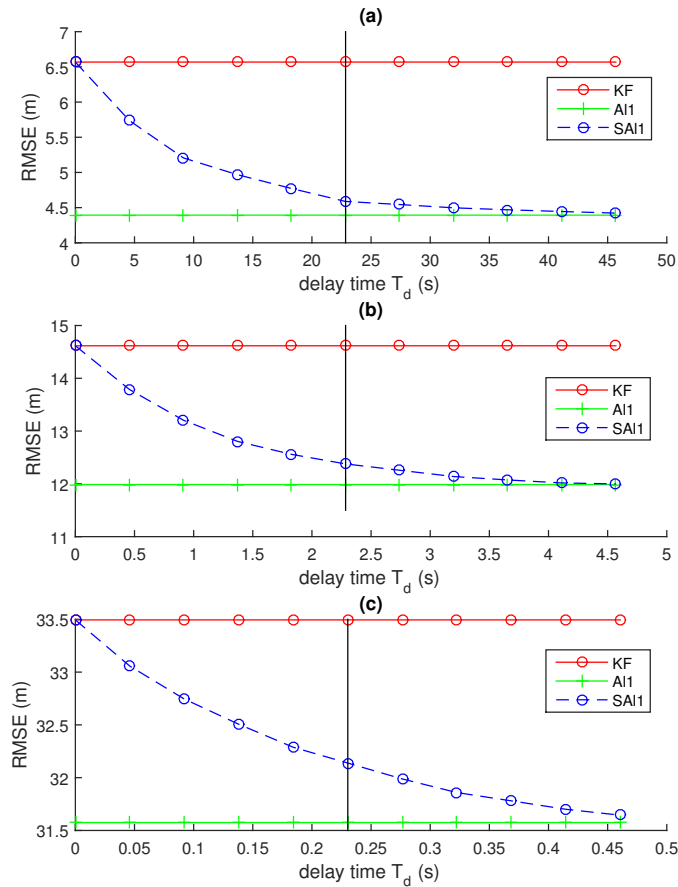


Figure 4. Performance of selective out-of-sequence filters as a function of the cut-off time T_{stale} . The vertical line represents the theoretical cut-off T_d computed using (8) with a k_{min}/k value of 0.6. (a) $\sigma_Q = 0.5, \sigma_R = 50, \Delta T_D = 50$; (b) $\sigma_Q = 5, \sigma_R = 50, \Delta T_D = 4$; (c) $\sigma_Q = 50, \sigma_R = 50, \Delta T_D = 0.75$. In all runs, $f = 1, h = 1, \Delta T_P = 0.2$.

Figures 5 and 6 illustrate the same experiments as Figure 4, but take the experiments further by examining the performance of the algorithm as the mean observation delay ΔT_D varies across several orders of magnitude. Each curve in the figure represents a run of the experiment with a different value for ΔT_D . The two figures show the same curves at different time-scales.

Figures 5 and 6 show that the points at which the performance of the SAI1 first reaches the mid-point between the performance of the KF that drops all observations and the SAI1 that improves all observations is fairly consistent despite significant change in the mean delay ΔT_D . It also shows that point where the SAI1 first reaches the performance of the AI1 is consistent as well.

The key thing to observe in these figures is the difference in the horizontal (time) axis between parts a, b, c, and d of the figure. Despite the maneuverability changing over three orders of magnitude, the predicted T_d is within less than one order of magnitude of the point where the SAI1 first reaches the performance of the AI1. This ranges from dozens of time steps in Figure 5a to less than 1 time step in Figure 5d.

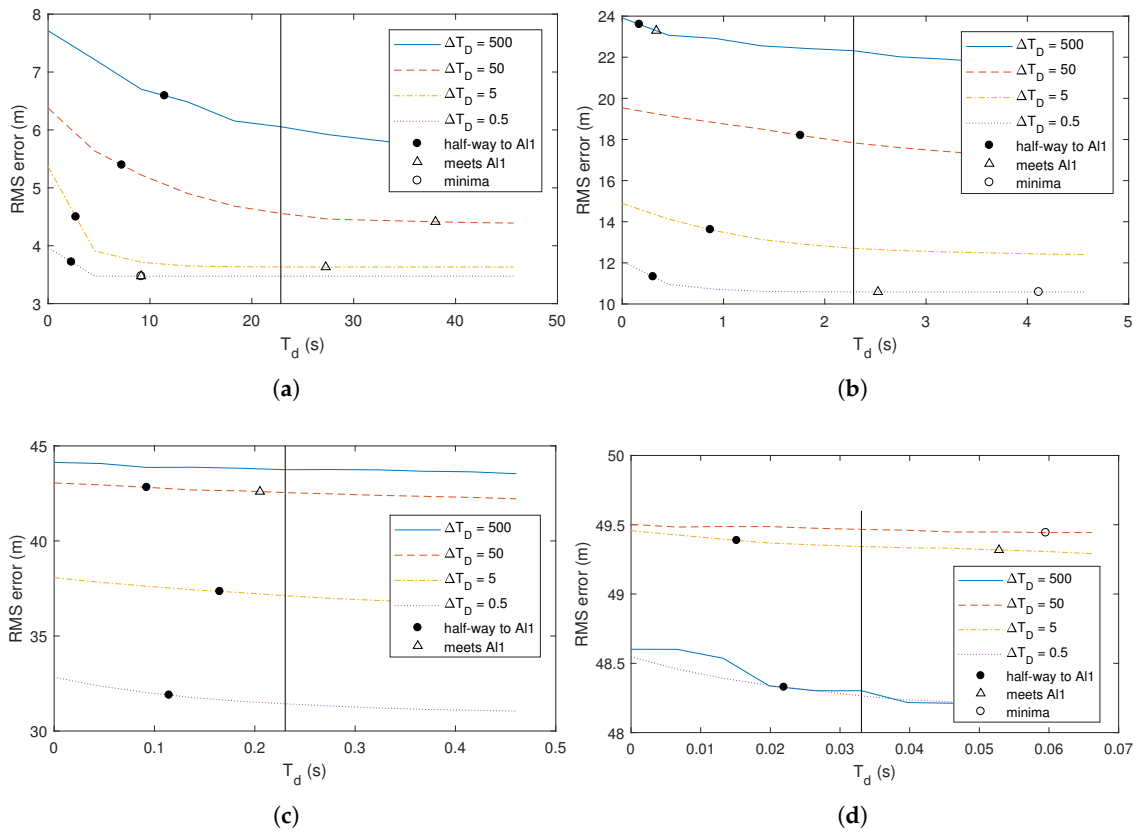


Figure 5. Performance of selective out-of-sequence filters as a function of the cut-off time T_{stale} for a wide range of ΔT_D viewing a narrow time-scale. The vertical line represents the theoretical cut-off T_d computed using (8) with a k_{min}/k value of 0.6. (a) $\sigma_Q = 0.5, \sigma_R = 50$; (b) $\sigma_Q = 5, \sigma_R = 50$; (c) $\sigma_Q = 50, \sigma_R = 50$; (d) $\sigma_Q = 500, \sigma_R = 50$. In all runs, $f = 1, h = 1, \Delta T_P = 0.2$.

Figure 6 illustrates that the cut-off threshold to achieve minimum error (marked with a circle (o) in the figure) often increases as the mean delay ΔT_P increases. This could seem odd, as the proposed threshold T_d is not a function of ΔT_P . However, as ΔT_P surpasses T_d , many observations are never incorporated by the filter because they have been delayed by more than ΔT_P when they arrive. One of the assumptions we made in the derivation of ΔT_P is that only one observation has been delayed, and that all the observations that were made after it have already been incorporated into the filter. When many of the observations are discarded because they come later than T_d , it changes the steady-state gain of the Kalman filter. As a result, if many observations are being discarded, the cut-off point should be determined based on the mean time between observations that are actually used rather than all those that are made. If the mean delay ΔT_P increases, the number of observations that are actually used decreases, increasing the mean time between measurements ΔT_P and the optimal cutoff T_d .

Similarly, when ΔT_P is significantly lower than T_d , it can be observed that the performance of the SA11 reaches the performance of the AI1 at a time less than T_d . This is to be expected because all observations have less delay than T_d and should all be incorporated.

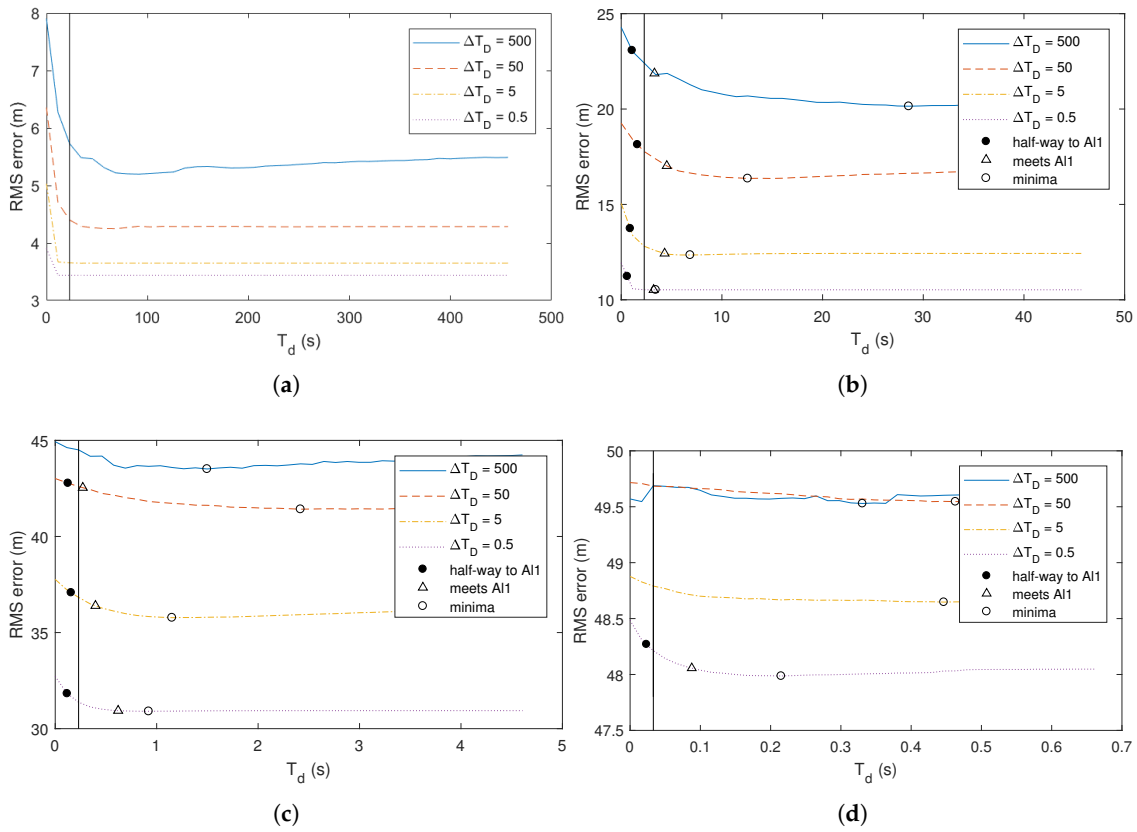


Figure 6. Performance of selective out-of-sequence filters as a function of the cut-off time T_{stale} for a wide range of ΔT_D viewing a narrow time-scale. The vertical line represents the theoretical cut-off T_d computed using (8) with a k_{min}/k value of 0.6. (a) $\sigma_Q = 0.5, \sigma_R = 50$; (b) $\sigma_Q = 5, \sigma_R = 50$; (c) $\sigma_Q = 50, \sigma_R = 50$; (d) $\sigma_Q = 500, \sigma_R = 50$. In all runs, $f = 1, h = 1, \Delta T_P = 0.2$.

4.2. Arrowhead Path

The white Gaussian process model, while convenient, is not a realistic model for predicting target motion. To investigate selective filtering when a target is following a nonlinear trajectory, we performed an experiment with the target following a deterministic non-linear path. The path is the arrowhead given by the parametric equations

$$x(t) = \begin{bmatrix} \rho \sin(\omega t + \sin \omega t) \\ \rho \cos(\omega t + \cos \omega t - \cos 1) \end{bmatrix} \quad (14)$$

with radius $\rho = 100$ m, nominal speed $s = 5$ m/s, and angular speed $\omega = s/\rho = 0.05$ rad/s (Illustrated as the red/light-gray line with dots in Figure 7).

For tracking, we used a random walk model (without a velocity component), with the two spatial dimensions decoupled. To determine the model noise, we measured the actual standard deviation of the motion noise along each dimension, and used the maximum noise, with rounding. This resulted in a process noise model variance of $q = 20$ m²/s. To generate the position observations, we added Gaussian noise with variance $r = 2000$ m². We used a regular time-step of $\Delta T_P = 1$ s (not a Poisson process), and added delay to the observations according to an exponential distribution with mean delay $\Delta T_D = 5$ s. With a k_{min}/k value of 0.6, and from (8), the time threshold to keep a stale observation is $T_{stale} = 5.11$, as illustrated by the vertical line in Figure 8. As in the previous section, we averaged the performance of the filter over 30 runs of 10,000 steps.

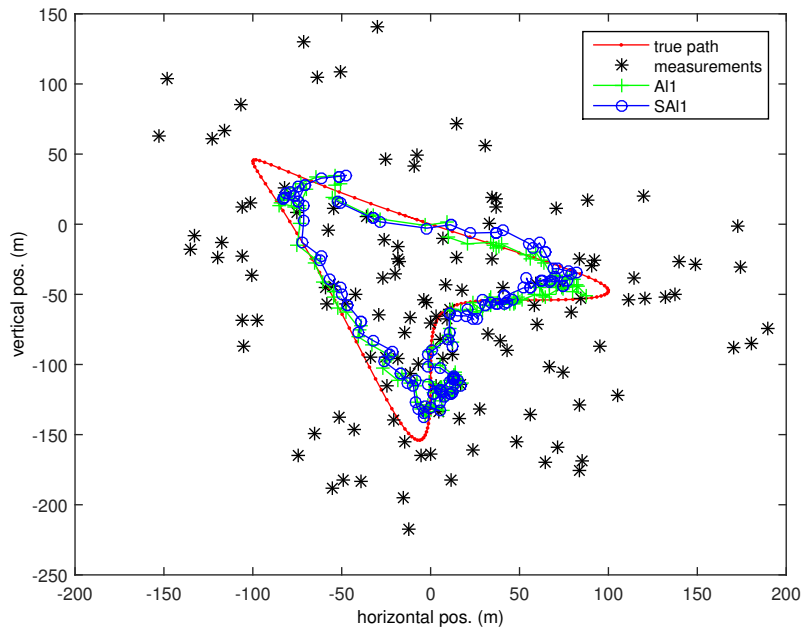


Figure 7. Arrowhead Path with random walk tracker.

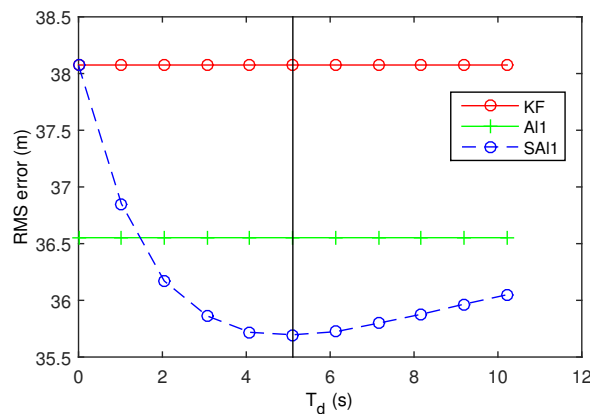


Figure 8. Performance of selective out-of-sequence filters for Arrowhead Path.

In these experiments, we are using an approximate out-of-sequence filter, the AI1 filter, rather than the expensive approach of buffering and re-processing observations when a delayed observation arrives. The AI1 filter is significantly faster than the buffered approach, but it does not guarantee an optimal result. Unlike the ideal Kalman filter, a delayed observation can actually hurt the performance of the AI1 filter.

We see how delayed observations can hurt a filter without optimal guarantees in Figure 8 where the selective AI1 (SAI1) filter has better performance at $T = T_{stale}$ than the AI1 filter incorporating all observations. Figure 9 shows how the AI1 filter lags more as it comes around the eastern end of the track (where the horizontal position reaches its maximum), because it is suboptimally incorporating older observations that the SAI1 does not. Increasing the threshold beyond T_{stale} actually hurts the performance of the SAI1 filter as it too incorporates older observations suboptimally. Contributing to the problem is the fact that the Kalman filter makes an assumption that the motion steps are random and uncorrelated when they are in fact deterministic and highly correlated.

Despite the invalid assumptions of the tracking model, our approach gives a good rough estimate of how much delay observations can have and still contribute significantly to estimation accuracy.

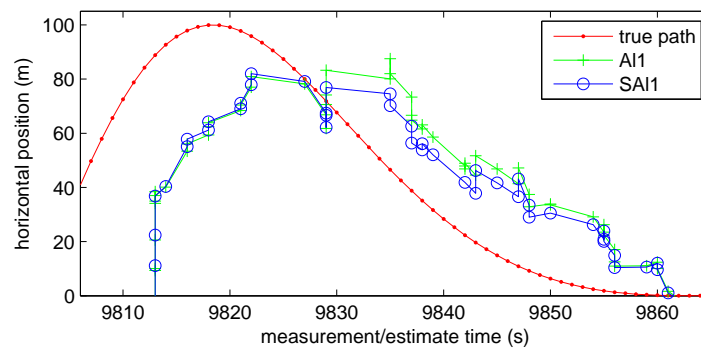


Figure 9. Performance of SAI1 filter compared with AI1 filter.

4.3. Extension to Nonlinear Filter

To investigate our approach in more general cases, we implemented an extended Kalman filter (EKF) for tracking a target that is following the arrowhead trajectory as follows

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ v_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} x_k + \Delta v_k \cos(\theta_k) \\ y_k + \Delta v_k \sin(\theta_k) \\ v_k \\ \theta_k \end{bmatrix} + \omega_k \quad (15)$$

$$\mathbf{z}_k = \mathbf{C}\mathbf{x}_k + \nu_k \quad (16)$$

where $\mathbf{x}_k = [x_k \ y_k \ v_k \ \theta_k]^T$ is the state vector at k , \mathbf{z} is the observation vector, (x_k, y_k) is the location of the target, v_k and θ_k are, respectively, the speed and heading of the target, $\omega_k \sim \mathcal{N}(0, \mathbf{Q}_k)$ is a zero mean Gaussian random vector with covariance \mathbf{Q}_k , representing the system disturbance, and $\nu_k \sim \mathcal{N}(0, \mathbf{R}_k)$ is a zero mean Gaussian random vector with covariance \mathbf{R}_k , representing the measurement noise. We assume that all states are completely measurable, i.e., $\mathbf{C} = \mathbf{I}^n$, where \mathbf{I}^n is the $n \times n$ identity matrix.

In this experiment, we have designed the EKF to drop delayed measurements if the delay time is greater than T_d . Otherwise, the EKF recursively processes them. For example, $T_d = 0$ means the filter does not process any delayed observations and $T_d = \infty$ means the filter processes all measurements, and therefore, the state estimates become identical to the case with no delay.

The root mean square error (RMSE) between the true paths and estimated trajectories with respect to the maximum delay (T_d) is shown in Figure 10a. The solid blue line is the average of 30 runs and the gray band indicates 1 standard deviation from the average. As it is expected, the RMSE exponentially approaches the minimum RMSE where we incorporate all delayed measurements. On the other hand, the computational time (Figure 10b) increases almost linearly, which indicates that there is not much merit to incorporate very old observations. It is evident that there is no closed-form solution like (5) in this case because the model is not linear time-invariant. Consequently, it is difficult to analytically find T_{stale} . However, this experiment shows that there is limited improvement on state estimates after $T_d > 5$ and it would be safe to set the threshold time to be $T_{stale} = \sim 5$ s, which corresponds to our analysis with the linear counterpart discussed in Section 4.2.

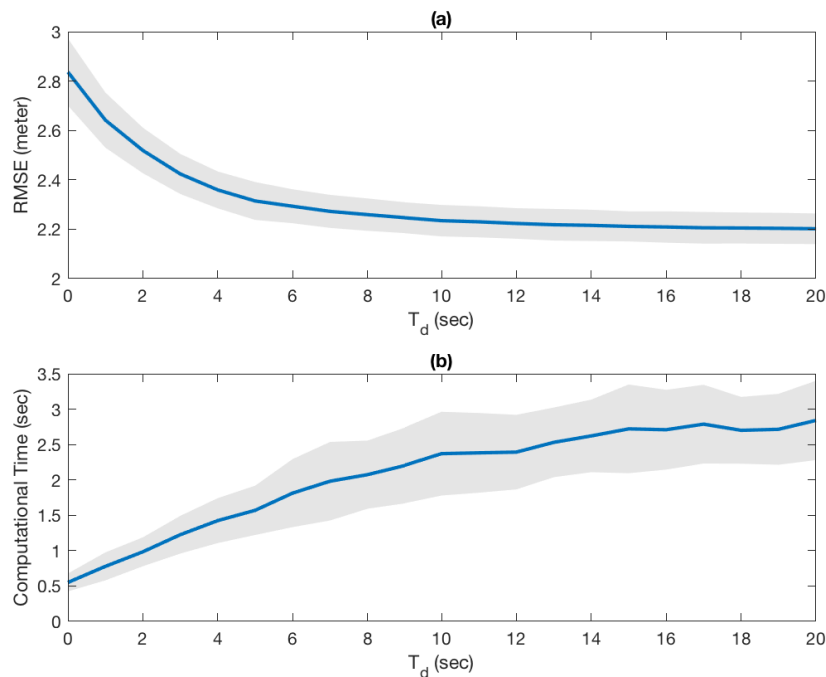


Figure 10. Performance of Extended Kalman filter. (a) (RMSE) between the true paths and estimated trajectories with respect to the maximum delay; (b) Computational time with respect to the maximum delay. The gray bands indicate 1 standard deviation from the mean.

5. Conclusions

This paper introduces a measure of the merit of a delayed observation based on the gain given by the ideal filter that reprocesses all newer observations every time a delayed observation arrives. We have demonstrated that this gain is a realistic estimate even when random measurement times perturb the system, and have demonstrated that this metric provides a simple and effective way to decide which observations to keep in a selective out-of-sequence filter.

It is useful to know when a delayed observation will not improve performance. The maximum delay time can be used to set the size of buffers if recent observations are stored to improve accuracy. The maximum observation delay can be used to decide whether an out-of-sequence filter will improve performance or not. Finally, the formulas provided in this paper can be used throughout the design of a tracking system system to predict how system parameters such as measurement delay, network latency, target speed, sensor accuracy, or sensing rate will impact system performance.

We believe that the intuition provided by this our approach is fundamental to the design of out-of-sequence filters. It provides critical intuition as to why some old observations do not improve performance, and others do. We hope that it will help shape the development of future OOS filters, whether they are selective or not.

Although the derivations in this paper have used linear systems, we have demonstrated that the theory here also lends intuition to the design of nonlinear variations on the Kalman filter such as the Extended Kalman Filter (EKF), Unscented Filter and, more generally, any Sigma-Point Kalman Filter (SPKF). As systems become more and more non-linear, the analysis above will become more and more of an approximation. Nevertheless, we expect that for many non-linear systems, the delayed Kalman gain is still exponentially decreasing. Investigating to what extent this is true in general nonlinear systems is the plan for our future work.

Author Contributions: J.Y. conceived the main idea of the merit of delayed observation. J.Y., S.B. and H.K. designed and performed the experiments and analyses. D.P. supervised all the tasks related to this work. J.Y. and S.B. prepared the manuscript.

Acknowledgments: The authors began this work while at the Academy Center for Unmanned Aerial Systems Research in the Department of Electrical and Computer Engineering, United States Air Force Academy, USAFA, CO 80840.

Conflicts of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Kim, S.H.; Choi, H.L. An Out-of-sequence Measurement Fusion Method for Guidance Filtering with Delayed Measurements. *Int. J. Control Autom. Syst.* **2018**, *16*, 512–521. [[CrossRef](#)]
2. Wang, X.; Qin, W.; Bai, Y.; Cui, N. Cooperative target localization using multiple UAVs with out-of-sequence measurements. *Aircraft Eng. Aerosp. Technol.* **2017**, *89*, 112–119. [[CrossRef](#)]
3. Liu, C.; Eben, L.; Yang, D.; Hedrick, K. Distributed Bayesian Filter Using Measurement Dissemination for Multiple Unmanned Ground Vehicles With Dynamically Changing Interaction Topologies. *J. Dyn. Syst. Meas. Control* **2017**, *140*, 030903. [[CrossRef](#)]
4. Ohno, Y.; Uozumi, S.; Ohnishi, K. Compensation of out-of-order delivery using Kalman Filter for teleoperation system. In Proceedings of the 2014 IEEE 13th International Workshop on Advanced Motion Control (AMC), Yokohama, Japan, 14–16 March 2014; pp. 488–493. [[CrossRef](#)]
5. Chen, Z.; Feng, X.; Zhou, F. Extended Kalman filter for networked control systems based on measurements summation. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 4542–4547. [[CrossRef](#)]
6. Zhang, K.; Rong Li, X.; Zhu, Y. Optimal update with out-of-sequence measurements. *IEEE Trans. Signal Process.* **2005**, *53*, 1992–2004. [[CrossRef](#)]
7. Alexander, H.L. State estimation for distributed systems with sensing delay. In *SPIE Proceedings of Data Structures and Target Classification, Orlando, FL, USA, 1 April 1991*; International Society for Optics and Photonics: Bellingham, WA, USA, 1991; Volume 1470, pp. 103–111.
8. Larsen, T.; Andersen, N.; Ravn, O.; Poulsen, N. Incorporation of time delayed measurements in a discrete-time Kalman filter. In Proceedings of the 37th IEEE Conference on Decision and Control, Tampa, FL, USA, 18 December 1998; Volume 4, pp. 3972–3977.
9. Bar-Shalom, Y. Update with out-of-sequence measurements in tracking: Exact solution. *IEEE Trans. Aerosp. Electron. Syst.* **2002**, *38*, 769–777. [[CrossRef](#)]
10. Zhang, S.; Bar-Shalom, Y. Optimal Update with Multiple Out-of-Sequence Measurements with Arbitrary Arriving Order. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 3116–3132. [[CrossRef](#)]
11. Van der Merwe, R. Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models. Ph.D. Thesis, Oregon Health & Science University, Portland, OR, USA, 2004.
12. Xu, S.; Doğançay, K.; Hmam, H. 3D AOA target tracking using distributed sensors with multi-hop information sharing. *Signal Process.* **2018**, *144*, 192–200. [[CrossRef](#)]
13. Liu, L.; Yang, A.; Zhou, W.; Tao, Q.; Tu, X.; Yue, J. *Filtering for Stochastic Systems with Transmission Delays and Out-of-Order Packets. Intelligent Computing, Networked Control, and Their Engineering Applications*; Yue, D., Peng, C., Du, D., Zhang, T., Zheng, M., Han, Q., Eds.; Springer: Singapore, 2017; pp. 497–506.
14. Singh, A.K.; Date, P.; Bhaumik, S. A Modified Bayesian Filter for Randomly Delayed Measurements. *IEEE Trans. Autom. Control* **2017**, *62*, 419–424. [[CrossRef](#)]
15. Tasoulis, D.K.; Adams, N.M.; Hand, D.J. Selective fusion of out-of-sequence measurements. *Inf. Fusion* **2010**, *11*, 183–191. [[CrossRef](#)]
16. Zhou, H.; Deng, Z.; Xia, Y.; Fu, M. Update with Out-of-Sequence Measurements. *Nurocomputing* **2016**, *214*, 212–225. [[CrossRef](#)]
17. Li, Z.; Zhang, H.; Mu, D.; Guo, L. Random Time Delay Effect on Out-of-Sequence Measurements. *IEEE Access* **2016**, *4*, 7509–7518. [[CrossRef](#)]
18. Kalata, P.R. The tracking index: A generalized parameter for α - β and α - β - γ target trackers. *IEEE Trans. Aerosp. Electron. Syst.* **1984**, *20*, 174–181. [[CrossRef](#)]
19. Chen, H.; Kirubarajan, T.; Bar-Shalom, Y. Performance Limits of Track-to-Track Fusion versus Centralized Estimation: Theory and Application. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 386–400. [[CrossRef](#)]

20. Oreshkin, B.N.; Liu, X.; Coates, M.J. Efficient delay-tolerant particle filtering. *IEEE Trans. Signal Process.* **2011**, *59*, 3369–3381. [[CrossRef](#)]
21. Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation: Theory, Algorithms, and Software*; Wiley & Sons: New York, NY, USA, 2001.
22. Kojima, C.; Takaba, K.; Kaneko, O.; Rapisarda, P. A characterization of solutions of the discrete-time algebraic Riccati equation based on quadratic difference forms. *Linear Algebra Its Appl.* **2006**, *416*, 1060–1082. [[CrossRef](#)]
23. Rojas, A.J. On the Discrete-Time Algebraic Riccati Equation and Its Solution in Closed-Form. *IFAC Proc. Vol.* **2011**, *44*, 162–167. [[CrossRef](#)]
24. Bar-Shalom, Y.; Chen, H.; Mallick, M. One-step solution for the multistep out-of-sequence-measurement problem in tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2004**, *40*, 27–37. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).