Staggered Scheduling of Estimation and Fusion in Long-Haul Sensor Networks

Qiang Liu and Xin Wang
Stony Brook University
Stony Brook, NY 11794
Email: {qiangliu,xwang}@ece.sunysb.edu

Nageswara S. V. Rao
Oak Ridge National Laboratory
Oak Ridge, TN 37831
Email: raons@ornl.gov

***Abstract*—In long-haul sensor networks, sensors are remotely deployed over a large geographical area to perform certain tasks. We study a class of such networks where sensors take measurements of one or more dynamic targets and send state estimates of the target(s) to a fusion center via long-haul satellite links. The severe loss and delay over the satellite channels can easily reduce the chance that an estimate is successfully received by the fusion center, thereby limiting the potential information fusion gain and resulting in suboptimal accuracy performance of the fused estimates. In this work, starting with the temporal-domain staggered estimation for an individual sensor, we explore the impact of the so-called intra-state prediction and retrodiction on estimation errors. We also investigate the effect of such estimation scheduling across different sensors on the spatial-domain fusion performance, where the sensors retain the same estimation frequency, but with possibly asynchronous estimation instants staggered over time. In particular, the impact of communication delay and loss on such scheduling is explored by means of numerical and simulation studies that demonstrate the validity of our analysis.**

***Index Terms*—Long-haul sensor networks, state estimate fusion, asynchronous and staggered estimation, staggered interval, intra-state and inter-state prediction and retrodiction, mean-square-error (MSE) performance, reporting latency.**

I. Introduction

Networked sensing systems can be found in a multitude of real-world applications, especially in detecting and/or monitoring of the states of dynamic targets. In particular, we consider a class of such systems – the so-called long-haul sensor networks – where the sensors are deployed to cover a very large geographical area, such as a continent or even the entire globe. Examples of such long-haul networks can be found in tasks such as the monitoring of greenhouse gas emissions using airborne and ground sensors [1], processing of global cyber events using cyber sensors distributed over the Internet [2], space exploration using a network of telescopes [12], and target detection and tracking for air and missile defense [5].

In a typical long-haul sensor network tracking/monitoring application, a remote sensor measures certain parameters of interest from the dynamic target(s) on its own, and then sends either the measurements directly, or the state estimates it derives from the measurements, to the fusion center. The fusion center serves to collect data from multiple sensors and fuse these data to obtain global estimates periodically at specified time instants. In this work, we focus on satellite-based long-haul sensor networks, where the sensors send out their time-stamped state estimates, rather than the raw measurement data, over the satellite links to a remote fusion center. In the end, a global estimate is expected to possess better quality levels in terms of its improved accuracy performance over that of the individual sensors; this effect is often referred to as the fusion gain.

Unfortunately, many challenges exist in such satellite-based long-haul sensor network estimation and fusion applications. Because of the long distance, often on a scale of tens of thousands of miles, the signal propagation time is rather significant compared to that in short-range communications. For example, the round-trip time (RTT) for signal propagation with a geostationary earth orbit (GEO) satellite is well over a half second [14]. More importantly, communication over the satellite links is characterized by sporadic high bit-error rates (BERs) and burst losses. The losses incurred during transmission or resulting from the message drop due to occasional high BERs could further reduce the number of reliable estimates available at the fusion center, which in turn degrades the fusion performance, since the global estimates cannot be promptly and accurately finalized by the fusion center. This can eventually result in failures to comply with the system requirements on the worst-case estimation error and/or maximum reporting delay, both crucial elements in systems calling for nearly real-time performance.

In the literature, some studies have attempted to address estimation and/or fusion under variable communication loss and/or delay conditions. The quality of information (QoI) performance is evaluated in [17] under network loss and delay for a target following the Brownian motion model, where measurements are selected based on the information gain calculated as a function of known probability loss and delay profiles. An upper bound of the loss rate has been derived in [6], above which the estimation error goes unbounded. Some studies, including [11], [18], [19], have addressed the so-called out-of-order-measurement (OOSM) issue – where an OOSM is defined as a measurement that has been generated earlier but arrives later – and their common goal is to update the current state estimate with an earlier measurement without reordering the measurements and recalculating the state estimator recursively. In these studies, the data will finally arrive albeit the random delay. More recently, a few studies [9], [10], [13] have exploited retransmission techniques to recover some of the lost messages over time so that the effect of information loss can be somewhat mitigated. A dynamic online selective fusion mechanism based on the projected information gain is proposed in [8] so that the final time for fusion is dynamically determined depending on if enough information has arrived at the fusion

center. Despite the above research efforts, there have been hardly any studies on temporal-domain scheduling design for improving the tracking performance in the context of state estimation/fusion applications.

In this work, we introduce a new concept, *staggered scheduling*. Different from conventional work where the time instants for sensors to generate estimates and the time for the fusion center to create fusion reports are deemed to coincide, we consider *asynchronous* and *staggered* estimation and fusion where there is *a time shift* (we call it "staggered time" in this work) between the two types of time instants. The fusion center can take advantage of the time difference to perform *intra-state* prediction and retrodiction to improve the quality of the fused estimates at desired reporting time. This scheduling can be carried out over multiple sensors as well. Correspondingly, the fusion center needs to fuse such multi-sensor state estimates with various time stamps.

The main goal of this work is to investigate the effect of such asynchronous and staggered estimation/fusion on the final accuracy performances under variable communication loss and delay conditions. In addition to the scheduling of sensors to generate estimates at specific time instants, the unexpected delay and loss of the communication network also naturally contribute to the time difference between sensor estimation and fusion time. Our results indicate that the fusion center can exploit staggered scheduling and opportunistically apply *intra-state* prediction and retrodiction to improve the fusion performance based on the link-level loss and delay conditions.

The remainder of this paper is organized as follows: Sec. II reviews the filtering fundamentals, highlighting the filtering methods as applied by the fusion center. Next in Sec. III, we introduce the basic ideas behind the staggered estimation by the technique of intra-state prediction and retrodiction. In Sec. IV, we explore the effect of such staggered design, in the context of network communication loss and delay, on the performance of one-sensor estimation. Similar studies are carried out for the two-sensor scenario in Sec. V where performance of two types of fusers is investigated. The paper concludes in Sec. VI.

II. Prediction and Retrodiction by the Fusion Center

This section provides a brief overview on filtering basics as well as the prediction and retrodiction techniques performed by the fusion center, which serves as the fundamentals for our staggered estimation/fusion design.

1) Filtering Basics: The goal of a state estimator is to extract the state information $\mathbf{x}$ from measurements $\mathbf{z}$ that is corrupted by noise; this is done by running a filter that outputs the state estimate $\hat{\mathbf{x}}$ and its associated error covariance matrix $\mathbf{P}$. Essentially, *filtering* is considered as a means to reduce uncertainty from noisy measurement data. In many recursive filtering algorithms, such as the well-known Kalman filtering (KF), each recursion is conceptualized as two distinct phases, namely, "predict" and "update". In the "predict" phase, the state estimate from the previous time step is used to produce an estimate of the current state, which is also known as the a priori state estimate. Subsequently, in the "update" phase, such a priori prediction is combined with the latest measurement to refine the state estimate, resulting in the a posteriori estimate that is supposed to possess improved quality in terms of reduced estimation error. Normally, these two phases alternate and the system state at a certain time instant is predicted directly from the preceding posterior estimate. In this work, we assume the sensors can take measurements and then in turn generate and send out their state estimates in a timely manner; it is the communication loss and delay between any sensor and the fusion center that may result in unavailable state estimates at the fusion center. Doing so allows us to focus mainly on designing information processing algorithms at the fusion center to improve the performance.

2) Prediction by the Fusion Center: We consider prediction performed not by a sensor during its regular recursive filtering, but by the fusion center. The purpose is largely different even though the two may share the same "prediction" equation. Since the fusion center does not have access to measurements, it needs the sensors to communicate their processed state estimates for subsequent fusion. However, due to severe loss and delay, the desired state estimates are not always available; in this case, the fusion center may simply *interpolate* the unavailable estimates by plugging in its own predicted estimates from earlier ones, using known or learned state evolution models. Hence, the prediction by the fusion center is used to counteract the effect of communication degradations. Due to the system uncertainty – often characterized by varying process noise levels – prediction alone often results in higher estimation errors compared to the estimates generated and sent by the sensors; this is the very reason measurements have to be taken regularly in order to maintain desired tracking performance. Nevertheless, to realize the fusion gain, it is still preferable to use predicted estimates for a sensor rather than discard the sensor's potential information altogether [4].

3) Retrodiction by the Fusion Center: Estimation of a target state at a particular time based on measurements collected beyond that time is called retrodiction or smoothing. Retrodiction improves the accuracy of the estimates, thanks to the use of more information, at the cost of extra delay. The vast majority of the existing literature studies have considered retrodiction only from the perspective of an individual sensor; the effect of retrodiction in the context of state fusion has been largely unexplored except in our earlier studies [8], [9]. Since retrodiction calls for the availability of subsequent data to the ones of interest, the inherent link delay over a long-haul network entails that the fusion center can exploit the opportunities for potential retrodiction to improve the accuracy of the fused estimate. Conventionally, an estimate is retrodicted only when it actually arrives as in many OOSM-related studies [11], [18], [19]. However, in our design, the fusion center opportunistically interpolates the missing estimates – that is, to "fill in the blanks" – from the available estimates before, using prediction, and after, with retrodiction. Of course, an available estimate can be retrodicted using its following estimates too – as in the conventional use of retrodiction – as long as the associated fused estimate has not been finalized by the fusion center. This has the potential benefit to speed up the process of finalizing the global estimates – since the fusion center does not have to wait for the actual missing estimates to finally arrive – and hence reduce the chance of missing the stipulated reporting deadline.

III. Staggered Estimation: An Overview

In this work, we propose a new estimation structure, *staggered estimation*, which exploits the temporal relationship between estimates at different time instants to improve the estimation performance. Before presenting the new concept, we first introduce some basic terminology and the conventional estimation structure.

To formulate the estimation and fusion process, we consider that a stream of globally fused estimates is reported at a regular time interval of T, which also coincides with the estimation interval at the sensors as well. Suppose the (continuous) time of interest is nT, where n is a positive integer, then due to the stationarity of the above interval T, in subsequent analysis, the time instant will also be conveniently referred to as the time (step) n. Based on the estimates sent by the sensors, the fusion center can perform prediction and retrodiction to form component state estimates for fusion. For a given sensor, depending on what estimates sent from the sensor have been received, the fusion center may report one of the following types of estimates, corresponding to a time n:

a) $\hat{\mathbf{x}}_{n|n}$, the "default" estimate sent from the sensor;
b) $\hat{\mathbf{x}}_{n^-}$, the predicted estimate;
c) $\hat{\mathbf{x}}_{n^-|n+1}$, the predicted & retrodicted estimate; and
d) $\hat{\mathbf{x}}_{n^+|n+1}$, the retrodicted estimate.

In the cases a) and d), the sensor's estimate for time n is successfully received by the fusion center; whereas in the other two cases, this estimate is missing and hence prediction over one or multiple steps[1] by the fusion center is performed first. As the system uncertainty accumulates over time, the estimation error often increases with the number of prediction steps that have accrued, which means that $\hat{\mathbf{x}}_{n|n-2}$ is a worse estimate than $\hat{\mathbf{x}}_{n|n-1}$. On the other hand, the presence of the sensor's estimate for step $n+1$ – in c) and d) – helps improve the quality of the estimate for time n. The improvement is on top of the predicted estimate in the case c) but on the already received sensor estimate in d).

The prediction and retrodiction techniques discussed above are schematically shown in Fig. 1(a), where the estimation interval T and its integer multiples serve as the basic time units for prediction and retrodiction. We conveniently name this as *inter-state* prediction and retrodiction.

Instead of forming the reports ideally at the same time instants as those when the sensor generate the estimates, which is difficult to achieve in reality, we consider a new staggered scheduling method shown in Fig. 1(b). With this method, a sensor is scheduled to generate its estimates following the same estimation interval T but at time instants different from the ones at which the fusion center creates reports. As a result, we allow both the prediction and retrodiction to be performed over a period of time that is a fraction of the estimation interval T. Hence, we have the *intra-state* prediction and retrodiction[2].

In Fig. 2, an example consisting of three different estimation schedules is shown. In the figure, the red dotted lines denote

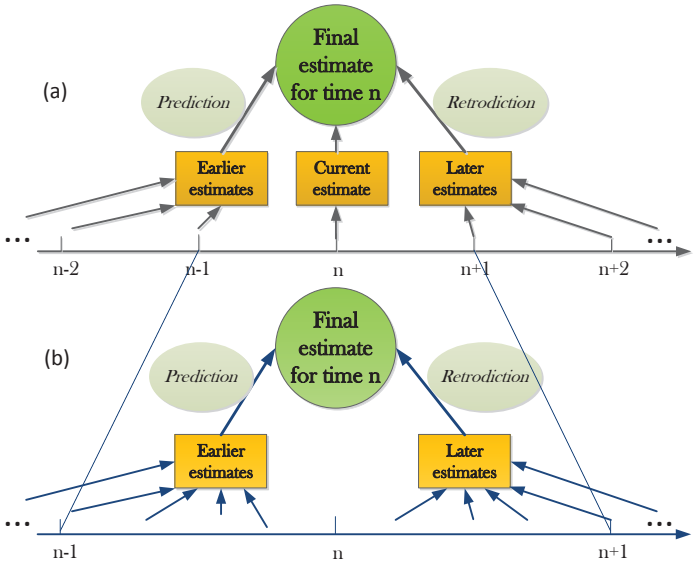

Fig. 1: Prediction and retrodiction: the estimate at time n is to be obtained. By convention as in (a), only estimates at steps $n-1$, $n+1$, etc. are to be used; however, in this study, we consider an estimate is generated at a time different from nT, by a fraction of the estimation interval, as shown in (b).

the common time instants of interest for the fusion center (i.e., the time instants whose state estimates are to be finalized and reported) and the green bars indicate the times where the estimates are generated. In (a), the standard estimation schedule is shown, where the estimation time at the sensors and fusion time at the fusion center always coincide[3]. In Fig. 2(b), an estimate with time-stamp $(n-0.2)T$ is sent out by the sensor, upon initial reception, the fusion center can perform a 0.2-step prediction to form the estimate report for time instant nT; next, if the subsequent estimate from the same sensor – now with time-stamp $(n+0.8)T$ – arrives before the reporting deadline (which is assumed to be one estimation interval T here), the fusion center can further perform a 0.8-step retrodiction for an improvement in accuracy over the previous predicted estimate. On the other hand, the estimation time in Fig. 2(c) always lags its preceding fusion time by $0.2T$, resulting in a 0.8-step prediction and a 0.2-step retrodiction when both estimates are available. In the figure, τ (tau) values are shown as the gap between the fusion time and the time stamp of the latest generated estimate. In all, when a sensor does not directly report its estimates for the time instants of interest but expects the fusion center to generate the corresponding estimates on its own for further fusion, we consider the scheduling as both "asynchronous" – from the perspective of the fusion center – and "staggered".

Albeit conceptually simple, the effects of this staggered scheduling on estimation and fusion performance are not readily predictable. In the next two sections, we will investigate its potential benefits for both one-sensor and two-sensor scenarios under variable communication loss and delay conditions.

[1]If the preceding estimate for step $n-1$ is available, then one-step prediction is in place; otherwise, multi-step prediction is necessary.

[2]Of course, the extension of this intra-state filtering can be realized by superimposing an estimation interval (and its multiples) on top of the fractional period τ.

[3]To avoid confusion, the term "fusion time" refers to the time of interest whose state estimate needs to be generated by the fusion center; whereas "reporting time" refers to the time when final fusion occurs.

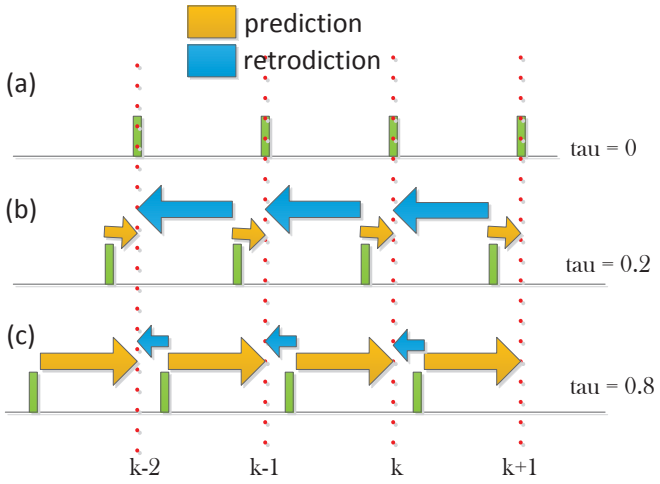

Fig. 2: Staggered estimation scheduling: (a) the standard schedule, where prediction and retrodiction over multiples of the common period T can be performed; (b) staggered estimation, where the sensor takes measurements $0.2T$ earlier than the following fusion time; (c) staggered estimation, where the sensor takes measurements $0.2T$ later than the preceding fusion time. The τ values are shown as the gap between the fusion time and the time stamp of the latest generated estimate.

IV. Staggered Estimation with One Sensor

In this section, we focus on staggered estimation scheduling where the sensor system is composed of only one sensor. Had the communication between the sensor and the fusion center been perfect with the standard synchronous scheduling being used, the fusion center would simply "take over" the sensor state estimates. With communication loss and delay, however, the fusion center may have a different view of the state evolution from that of the sensor due to the use of prediction and retrodiction. We first introduce the communication loss and delay model; and then we analyze the probabilities of generating different types of estimates by a certain deadline and show the impact of scheduling on estimation performance. Numerical studies are carried out at the end of the section using a well-known target motion model.

A. Link Communication Loss and Delay Profiles

The communication loss and delay characteristics are determined by the long-haul link conditions. In this work, we assume that each message sent by a sensor is lost en route to the fusion center with probability p that is independent of other messages. The latency that a message undergoes before arriving at the fusion center may consist of the initial detection and measurement delay, data processing delay, propagation delay, and transmission delay, among others. We suppose a pdf $f(t)$ can model the overall delay t that a message experiences before being successfully received by the fusion center. One typical example is that of the shifted exponential distribution:

$$f(t) = \frac{1}{\mu}\exp^{-\frac{t-T_I}{\mu}}, \text{ for } t \geq T_I. \tag{1}$$

in which T_I serves as the common link and processing delay, which is the minimum delay that a message must experience to reach the fusion center, and μ is the mean of the random delay beyond T_I that can be affected by factors such as weather and terrain.

B. Staggered Scheduling with One Sensor

In this subsection, we explore the effects of different staggered estimation schedules in the presence of communication loss and delay. By analyzing the probabilities of having different types of estimates as inputs to a certain fuser, we calculate the approximate error performance and compare it against the actual position mean-square-error (MSE) performance from numerical studies in the next subsection.

1) Probabilities for Obtaining Different Types of Estimates: We first consider the specific condition under which a certain number of retrodiction rounds can potentially take place. Suppose the interval between the time of interest and the preceding sensor estimation time is τ, where $0 \leq \tau < T$; in other words, the time stamp of the preceding estimates is $nT - \tau$. Suppose the reporting deadline for time nT is $nT + D$ (i.e., with a maximum lag D); then in order to possibly perform at least one round of retrodiction, an estimate must be generated after time nT and arrive at the fusion center by $nT + D$. Since the time stamp of the estimate following time nT is $(n+1)T - \tau$, accounting for the minimum delay T_I as in Eq. (1), the earliest arrival time $(n+1)T-\tau+T_I$ should be no later than the deadline $nT+D$; on the other hand, to have only up to one round of retrodiction, the estimate generated at time $(n+2)T - \tau$ should arrive later than $nT + D$. Combining both constraints, we have the condition for both the reporting lag D and the scheduling lag τ with up to one round of retrodiction. In fact, this result can be easily extended to multi-round retrodiction, as stated below:

Proposition 4.1: To have up to l ($l \geq 1$) rounds of retrodiction, the reporting lag D should satisfy the following condition:

$$lT + T_I - \tau \leq D < (l+1)T + T_I - \tau. \tag{2}$$

Without loss of generality, in the following analysis, T and D are given as 1 s and 1.5 s respectively, with the common link communication and pre-processing delay T_I set as 0.5 s. This is the situation where in the standard scheduling scheme, the deadline for reporting one estimate happens to be the very earliest time the subsequent estimate arrives, namely, $D = T + T_I$. Also, it is easy to verify that $l = 1$.

Given the link statistics introduced in the last subsection, the probability that a sensor estimate is successfully received by the fusion center within time t since being generated is $(1-p)F(t)$. It is easy to verify that the amount of time it takes for the two estimates, one immediately preceding nT and the other following it, to be delivered to the fusion center before the deadline, are $D+\tau$ and $D-T+\tau$, respectively. As such, we have the following probabilities of using different types of estimates by the deadline:

a) $\hat{\mathbf{x}}_{n|n-\tau}$:
$(1-p)F(D+\tau)(1-(1-p)F(D-T+\tau))$;

b) $\hat{\mathbf{x}}_{n^-}$:
$(1-(1-p)F(D+\tau))(1-(1-p)F(D-T+\tau))$;

c) $\hat{\mathbf{x}}_{n^-|n+1-\tau}$:
$(1-(1-p)F(D+\tau))(1-p)F(D-T+\tau)$; and

d) $\hat{\mathbf{x}}_{n^+|n+1-\tau}$:
$(1-p)^2F(D+\tau)F(D-T+\tau)$.

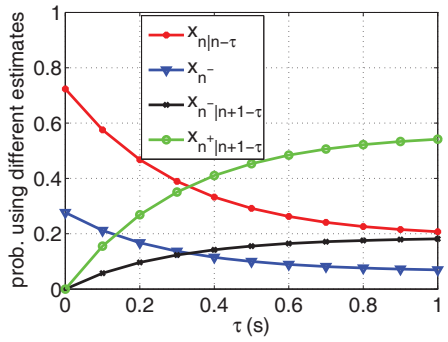

Fig. 3: Probabilities of using different types of estimates at the deadline with variable staggered estimation interval τ, where loss rate $p = 0.25$, $T = 1$, and $D = 1.5$.

Similar notations for these types of estimates were first introduced in Sec. III, now with the exception that the staggered interval τ is added to the subscripts to reflect the time difference. Note that when $\tau = 0$, the results are simply reduced to those under standard scheduling. In the cases b) and c), the minus signs denote that the estimate generated at $n - \tau$ is not available at the fusion center; as such, these probabilities have also incorporated the scenarios where prediction over a longer time span by the fusion center has taken place.

Given a pre-determined set of estimation interval T and deadline D values, the question of interest arises: how would different τ values impact the estimation performance at the fusion center? In Fig. 3, the probabilities of eventually using different types of estimates by the fusion center are plotted for a loss rate of $p = 0.25$. As can be easily seen in the figure, as the staggered interval τ moves from 0 all the way up to T, the probability of obtaining $\hat{\mathbf{x}}_{n^+|n+1-\tau}$ goes up while that of using the predicted state $\hat{\mathbf{x}}_{n^-}$ decreases. Among the four, these two represent the best and worst estimates respectively in terms of the estimation error. Increasing τ would then seem to improve the estimation performance when only these two types of estimates are considered. However, the other two types of estimate, $\hat{\mathbf{x}}_{n|n-\tau}$ and $\hat{\mathbf{x}}_{n^-|n+1-\tau}$, change in reverse directions too as τ shifts, and it is not immediately clear which of the two has overall better accuracy performance [9].

However, the above analysis does not capture the actual behavior of the estimate to be finalized by the fusion center, since intra-state prediction and/or retrodiction has to be applied when τ shifts away from zero, thereby affecting the behavior of all four types of estimates. In what follows, we will explore the error profiles with staggered scheduling under perfect communications. Then we will combine them with the above probabilistic analysis to derive the approximate estimation error performance. To do so, it is necessary to introduce the system and measurement models first.

C. Quantitative Results

1) System and Measurement Models: In most 2-D or 3-D tracking applications, the same motion model is used for each coordinate, and the motion along each coordinate is assumed to be decoupled from other coordinates [4]; therefore, we consider a kinematic model in one *generic* coordinate[4]. The motion uncertainty can be modeled by the process noise – also defined in one generic coordinate – that is continuous in time as a white stochastic process with a certain power spectral density (PSD). The continuous-time system has a state vector $\mathbf{x}$ consisting of position, velocity, and acceleration,

$$\mathbf{x} = [p \quad v \quad a]^T \tag{3}$$

where the derivative of acceleration is modeled by a zero-mean white jerk process noise. The system dynamics described by a discrete-time state equation with a sampling period T are given by

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{w}_k, \tag{4}$$

where the state transition matrix $\mathbf{F}$ is defined as

$$\mathbf{F} = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}. \tag{5}$$

The process noise $\mathbf{w}_k$ is stationary over time with its covariance matrix $\mathbf{Q}$

$$\mathbf{Q} = \begin{bmatrix} T^5/20 & T^4/8 & T^3/6 \\ T^4/8 & T^3/3 & T^2/2 \\ T^3/6 & T^2/2 & T \end{bmatrix} q, \tag{6}$$

where $q \triangleq \sigma_w^2 = \mathbb{E}[\mathbf{w}_k \mathbf{w}_k^T]$ is the noise power spectral density.

The above motion model is named the continuous-time Wiener process acceleration (CWPA) model as in [3], [4]. It is appropriate for targets with longer maneuvering intervals (or shorter sampling intervals). When the noise PSD q is relatively small, it is also called the nearly constant acceleration (NCA) model. We will consider cases with both large and small noise PSDs. In this kinematic model, the sampling interval T appears in both the stationary state transition matrix $\mathbf{F}$ and the noise covariance matrix $\mathbf{Q}$. Conventionally, this T serves as the estimation and fusion intervals as well.

We consider a simplified measurement model in which the sensor directly measures the position state of interest and hence only the position estimate

$$z_k = \mathbf{H}\mathbf{x}_k + v_k \tag{7}$$

is available, where $\mathbf{H}$, defined as

$$\mathbf{H} = [1 \quad 0 \quad 0], \tag{8}$$

is the measurement matrix, and the Gaussian measurement noise v has the following autocorrelation:

$$\mathbb{E}[v_k v_j] = R\delta_{kj} \triangleq \sigma_v^2 \delta_{kj}, \tag{9}$$

where $\delta_{(\cdot)}$ is the Kronecker delta function.

[4]In other words, motion along different directions in a particular coordinate system (such as the common "east-north-up" system) is independent and can be mapped onto corresponding orthogonal axes.

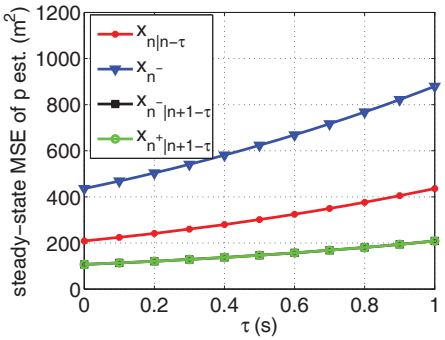

Fig. 4: Steady-state position estimate mean-square-error (MSE) performance with variable staggered interval τ values

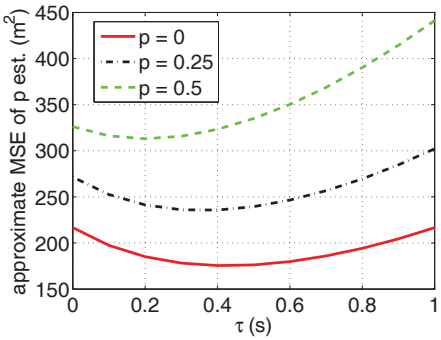

Fig. 5: Approximate position estimate MSE performance with variable staggered interval τ and loss rate p values: only prediction across time of up to $2T$ is considered

2) Approximate Estimation Error: In this work, the fusion center applies the Rauch-Tung-Streibel (RTS) retrodiction algorithm [15] to obtain the retrodicted state estimates. With the previously established models, the steady-state behavior [15] of the sensor estimate can be found analytically via Ricatti equation recursion or more conveniently from simulations. In our subsequent analysis, we assume that $q = 1$ and $\sigma_v = 20$. In Fig. 4, the steady-state error performance of different types of estimates under variable τ values is displayed. Again, with our parameter setup, only the sensor estimates generated at $n+1-\tau$, if available, can be used by the fusion center for retrodiction. Another assumption used in generating the plots is that no bursty loss is present; that is, the number of prediction steps is constrained strictly under two. For example, in Cases b) and c) in the last subsection, the minus sign would mean that only the immediately preceding estimate is not received, but not the ones before.

From the plots, as τ gradually shifts away from 0, all types of estimates experience increased steady-state estimation errors. Recall that under the steady-state condition, a sensor estimate has the same theoretical MSE guarantee regardless of its time of origin. Suppose two adjacent sensor estimates are successfully delivered to the fusion center (as in the case where $\hat{\mathbf{x}}_{n^+|n+1-\tau}$ can be obtained); as τ increases, the intra-state prediction step size is lengthened and retrodiction step size shortened, resulting in increased estimation error. This relationship holds true for all other cases as well. Another interesting observation is that the two cases with $\hat{\mathbf{x}}_{n^+|n+1-\tau}$ and $\hat{\mathbf{x}}_{n^-|n+1-\tau}$ have nearly identical steady-state performance. This means that had the communications been perfect, the frequency that a sensor communicates its estimates (but with the same estimation frequency on tap) can be reduced by half without causing tangible performance degradations.

Finally, we calculate the expected estimation MSE performance as the probabilistic combination of steady-state MSEs of different types of estimates. More specifically, the expected MSE with a certain τ choice is computed as the summation of the probabilities of obtaining all four types of estimates, such as those shown in Fig. 3, times the corresponding steady-state position MSEs found in Fig. 4. This result is "approximate" at best in that the probabilities themselves may have included the cases where a string of losses occur. The results are plotted in Fig. 5 with three different link-level loss rates, namely 0%, 25%, and 50%. Interestingly, across all cases, the estimation errors decrease initially as τ shifts away from zero, and then increases. For validation of the results, however, we also need to test the *actual* estimation error performance via Monte-Carlo simulations.

3) Actual Estimation Error Performance: The same set of parameters are used to generate the actual position estimate MSE performance as shown in Fig. 6. Comparing it with Fig. 5, we can observe the following: First, the above approximation by probabilistic combination becomes increasingly erroneous as the loss rate increases. When there is no or little loss, the off-line probabilistic values serve as a good approximation of the actual error profile; however, as p increases, bursty losses become more commonplace, which was not reflected in the steady-state MSE values in Fig. 4, resulting in overly optimistic approximation when the loss becomes severe (as in the $p = 0.5$ case in the figure). Also the minimum estimation error is somewhat skewed in the approximation. Nevertheless, a common time across cases where the minimum estimation errors can be found happens to be around $\tau = 0.4$ s. Here, at zero loss rate, the standard scheduling results in a nearly 30% higher estimation MSE compared to the value obtained at $\tau = 0.4$ s; even at a 25% loss rate, standard scheduling still yields 20% more errors compared to its staggered counterpart. As the loss becomes even higher, the improvement from staggered scheduling in terms of the percentage of error reduction becomes less prominent as the fusion center encounters more difficulties receiving estimates regardless of their time of origin. But overall, the error reduction performance showcases the major advantage of scheduling sensor estimation activity in a staggered manner.

V. Staggered Estimation and Fusion with Two Sensors

Having explored the error performance with one sensor, we now move on to the two-sensor case, where information fusion of the sensor estimates also needs to be accounted for. First we briefly introduce the fusers to be used here and then again investigate the estimation error performance via simulations.

A. Fusers

It is a well known fact that the common process noise in measuring the motion of any target results in correlation among estimates generated by multiple sensors. The cross-covariance is the term that describes this spatial correlation. However, it

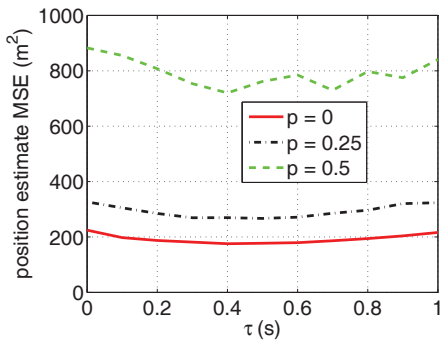

Fig. 6: Actual position estimate MSE performance with variable staggered interval τ and loss rate p values

is in general a very challenging task to derive the exact cross-covariance terms in practice. We consider two types of fusers where the fused estimate can be obtained directly with no cross-covariance calculation needed.

1) Linear Fuser without Cross-Covariance: In tracking applications, the track-to-track fuser (T2TF) [4] is a linear fuser that is optimal in the linear minimum mean-square error (LMMSE) sense. The fused state estimate $\hat{\mathbf{x}}_F$ and its error covariance $\mathbf{P}_F$ are defined for two sensors [4] as:

$$\hat{\mathbf{x}}_F = \hat{\mathbf{x}}_1 + (\mathbf{P}_1 - \mathbf{P}_{12})(\mathbf{P}_1 + \mathbf{P}_2 - \mathbf{P}_{12} - \mathbf{P}_{21})^{-1}(\hat{\mathbf{x}}_2 - \hat{\mathbf{x}}_1) \quad (10)$$

$$\mathbf{P}_F = \mathbf{P}_1 - (\mathbf{P}_1 - \mathbf{P}_{12})(\mathbf{P}_1 + \mathbf{P}_2 - \mathbf{P}_{12} - \mathbf{P}_{21})^{-1}(\mathbf{P}_1 - \mathbf{P}_{21}) \quad (11)$$

where $\hat{\mathbf{x}}_i$ and $\mathbf{P}_i$ are the state estimates and error covariance from sensor i, respectively, and $\mathbf{P}_{ij} = \mathbf{P}_{ji}^T$ is the error cross-covariance between sensors i and j. However, if the sensor errors are correlated but the cross-covariance is unavailable, one may assume that the cross-covariance is zero in order to apply this linear fuser, even though the result will be suboptimal. The fuser would then reduce to a simple convex combination of the state estimates:

$$\mathbf{P}_F = (\mathbf{P}_1^{-1} + \mathbf{P}_2^{-1})^{-1} \quad (12)$$

$$\hat{\mathbf{x}}_F = \mathbf{P}_F(\mathbf{P}_1^{-1}\hat{\mathbf{x}}_1 + \mathbf{P}_2^{-1}\hat{\mathbf{x}}_2) \quad (13)$$

2) Fast Covariance Intersection (CI) Algorithm: Another sensor fusion method is the covariance intersection (CI) algorithm. The intuition behind this approach comes from a geometric interpretation of the problem. If one were to plot the covariance ellipses for $\mathbf{P}_F$ (defined as the locus of points $\{\mathbf{y} : \mathbf{y}^T\mathbf{P}_F^{-1}\mathbf{y} = c\}$ where c is some constant), the ellipses of $\mathbf{P}_F$ are found to always lie within the intersection of the ellipses for $\mathbf{P}_1$ and $\mathbf{P}_2$ for all possible choices of $\mathbf{P}_{12}$ [7]. The intersection is characterized by the convex combination of sensor covariances:

$$\mathbf{P}_F = (\omega_1\mathbf{P}_1^{-1} + \omega_2\mathbf{P}_2^{-1})^{-1} \quad (14)$$

and the corresponding sensor fusion for the CI algorithm is

$$\hat{\mathbf{x}}_F = \mathbf{P}_F\left(\omega_1\mathbf{P}_1^{-1}\hat{\mathbf{x}}_1 + \omega_2\mathbf{P}_2^{-1}\hat{\mathbf{x}}_2\right), \quad \omega_1 + \omega_2 = 1 \quad (15)$$

where $\omega_1, \omega_2 > 0$ are weights to be determined (e.g., by minimizing the determinant of $\mathbf{P}_F$).

More recently, Wang and Li [16] proposed a fast CI algorithm where the weights are found based on an information-theoretic criterion so that ω_1 and ω_2 can be solved for analytically as follows:

$$\omega_1 = \frac{D(p_1, p_2)}{D(p_1, p_2) + D(p_2, p_1)} \quad (16)$$

where $D(p_A, p_B)$ is the Kullback-Leibler (KL) divergence from $p_A(\cdot)$ to $p_B(\cdot)$, and $\omega_2 = 1 - \omega_1$. When the underlying estimates are Gaussian, the KL divergence can be computed as:

$$D(p_i, p_j) = \frac{1}{2}\left[\ln\frac{|\mathbf{P}_j|}{|\mathbf{P}_i|} + \mathbf{d}_X^T\mathbf{P}_j^{-1}\mathbf{d}_X + Tr(\mathbf{P}_i\mathbf{P}_j^{-1}) - k\right] \quad (17)$$

where $\mathbf{d}_X = \hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j$, k is the dimensionality of $\hat{\mathbf{x}}_i$, and $|\cdot|$ denotes the determinant. This fast-CI algorithm will be used for a quantitative comparison against the above simple linear fuser with unavailable cross-covariances.

B. Estimation Error Performance with Two Sensors and Staggered Scheduling

Two sensors are assumed to have the same measurement noise profile. In this case, we need to consider all the combinations of different staggered intervals for both sensors – relative to the reporting time instants at the fusion center – denoted as τ_1 and τ_2, respectively. Probabilistic analysis similar to that in the last section can be carried out for both sensors. However, our focus here is to analyze the Monte Carlo simulation results as shown in Figs. 7 and 8, in which results for the simple linear fuser and the fast-CI fuser are plotted respectively.

From the figures, all generated three-dimensional surfaces resemble a sheet with downward-curved center portions, the extension of the earlier one-sensor estimation performance. We observe that for nearly all cases, the fast-CI fuser outputs estimates that are of slightly worse quality than those generated by the simple linear fuser[5]. Also the fast-CI fuser is more sensitive to the changes in the loss rate. The increase in estimation MSE with a more lossy link is more dramatic in the CI fuser. Another common feature across the cases is that the standard scheduling $\tau_1 = \tau_2 = 0$ happens to result in the highest estimation error. Comparing the results here to those in the one-sensor case shown in Fig. 6, we can see that both fuser outputs have MSEs that are more than half of those values with one sensor only, reflecting the effect of the common process noise and cross-covariance.

Although not easily discernible in the figures here, a more "microscopic" examination of the numerical results reveals the effect of cross-covariance in staggered scheduling design. Individually, at $\tau = 0.4$ s, the fusion center can expect the least estimation error from either of the two sensors. However, the case where $\tau_1 = \tau_2 = 0.4$ s does not achieve the best fuser outputs; another point close by does. This observation can be construed as the reduction of cross-covariance by staggering the estimation time across sensors. If the two sensors take samples at the same time (even at optimal $\tau = 0.4$ s), the cross-covariance is the highest; as the time separation in between increases, so does the reduction of correlation. This reduction of cross-covariance over time was also observed in one of our earlier numerical studies [9] and is especially of interest for further investigation.

[5]Note that if $\mathbf{P}_1 = \mathbf{P}_2$, then $\omega_1 = \omega_2 = 0.5$, and the resulting fused estimate will be equivalent to that from Eq. (13) but with an inflated error covariance matrix (increased by a factor of 2).

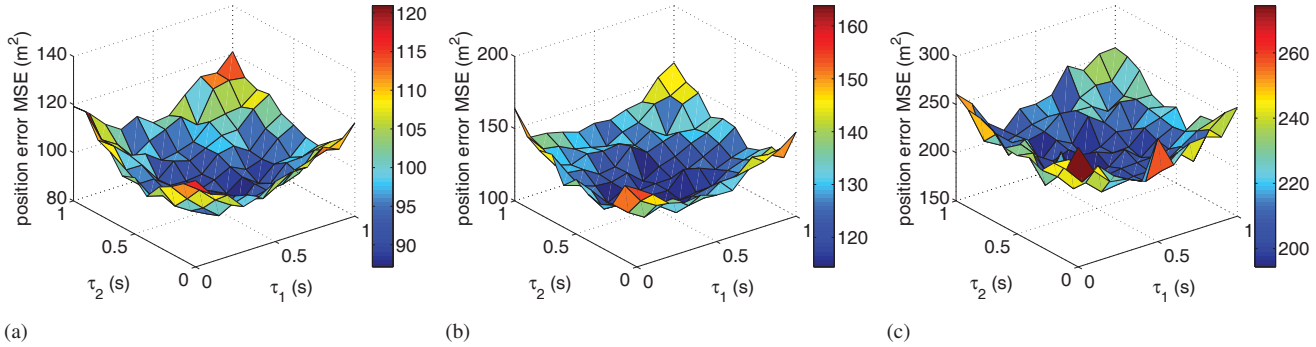

Fig. 7: Actual position estimate MSE performance with two-sensor linear fusion and varying staggered interval τ and loss rate p: (a) $p = 0$; (b) $p = 0.25$; (c) $p = 0.5$

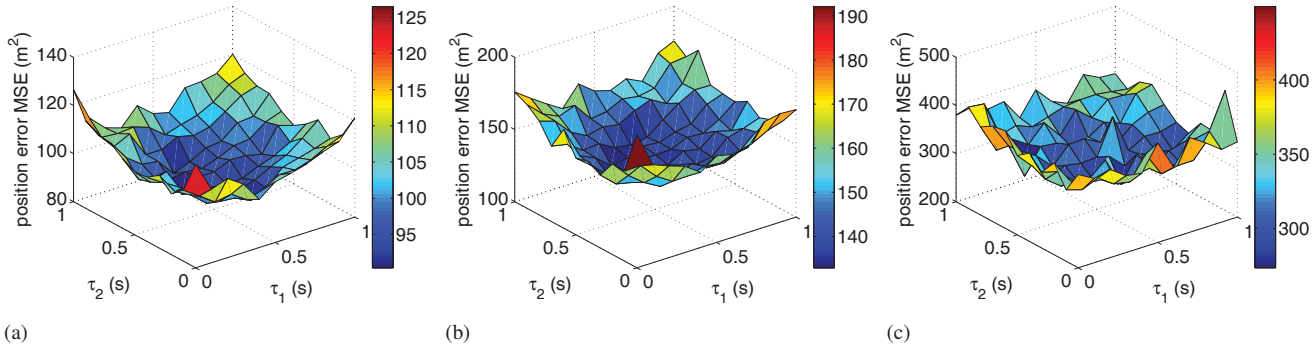

Fig. 8: Actual position estimate MSE performance with two-sensor fast-CI fusion and varying staggered interval τ and loss rate p: (a) $p = 0$; (b) $p = 0.25$; (c) $p = 0.5$

VI. Conclusion

.

In this work, we studied how the fusion center can exploit staggered scheduling and opportunistically apply intra-state prediction and retrodiction to improve the fusion performance based on the link-level loss and delay conditions. Tracking performances in one- and two-sensor cases are provided that demonstrate the advantages of staggered scheduling design. Extensions of this study include incorporation of more sensors as well as more complex target maneuvering models. Also of interest are dynamic scheduling algorithms based on the instantaneous communication condition.

References

[1] Earth systems research laboratory, national oceanic and atmospheric administration. //www.esrl.noaa.gov/gmd/ccgg/index.html.

[2] Network telescope research, 2011. www.caida.org/research/security/telescope/.

[3] Y. Bar-Shalom, T. Kirubarajan, and X. R. Li. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., New York, NY, 2002.

[4] Y. Bar-Shalom, P. K. Willett, and X. Tian. *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishers, 2011.

[5] W. Boord and J. B. Hoffman. *Air and Missile Defense Systems Engineering*. CRC Press, 2013.

[6] A. Chiuso and L. Schenato. Information fusion strategies and performance bounds in packet-drop networks. *Automatica*, 47:1304–1316, Jul. 2011.

[7] S. J. Julier and J. K. Uhlmann. *General Decentralized Data Fusion with Covariance Intersection, ser. Handbook of Multisensor Data Fusion*. CRC Press, 2001.

[8] Q. Liu, X. Wang, and N. S. V. Rao. Fusion of state estimates over long-haul sensor networks under random delay and loss. In *Proceedings of 31st IEEE International Conference on Computer Communications (INFOCOM 2012), Mini-conference*, pages 2968–2972, Orlando, FL, Mar. 2012.

[9] Q. Liu, X. Wang, N. S. V. Rao, K. Brigham, and B. V. K. Vijaya Kumar. Fusion performance in long-haul sensor networks with message retransmission and retrodiction. In *9th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS 2012)*, Las Vegas, NV, Oct. 2012.

[10] Q. Liu, X. Wang, N. S. V. Rao, K. Brigham, and B. V. K. Vijaya Kumar. Performance of state estimate fusion in long-haul sensor networks with message retransmission. In *Proc. Information Fusion (FUSION), 2012 15th International Conference on*, pages 719–726, Singapore, Singapore, Jul. 2012.

[11] M. Mallick and K. Zhang. Optimal multiple-lag out-of-sequence measurement algorithm based on generalized smoothing framework. In *Proc. SPIE, Signal and Data Processing of Small Targets*, San Diego, CA, Apr. 2005.

[12] University of Southampton. Global network of new-generation telescopes will track astrophysical events as they happen, *ScienceDaily*, Jan. 2011.

[13] N. S. V. Rao, K. Brigham, B. V. K. Vijaya Kumar, Q. Liu, and X. Wang. Effects of computing and communications on state fusion over long-haul sensor networks. In *Proc. Information Fusion (FUSION), 2012 15th International Conference on*, pages 1570–1577, Singapore, Singapore, Jul. 2012.

[14] D. Roddy. *Satellite Communications*. McGraw-Hill, 2006.

[15] D. Simon. *Optimal state estimation: Kalman, H [infinity] and nonlinear approaches*. Wiley-Interscience, 2006.

[16] Y. Wang and X. Li. Distributed estimation fusion with unavailable cross-correlation. *Aerospace and Electronic Systems, IEEE Transactions on*, 48(1):259–278, Jan. 2012.

[17] W. Wei, T. He, C. Bisdikian, D. Goeckel, and D. Towsley. Target tracking with packet delays and losses - qoi amid latencies and missing data. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on*, pages 93 –98, Mar.-Apr. 2010.

[18] K. Zhang, X. R. Li, and Y. Zhu. Optimal update with out-of-sequence measurements. *Signal Processing, IEEE Transactions on*, 53(6):1992–2004, Jun. 2005.

[19] S. Zhang, Y. Bar-Shalom, and G. Watson. Tracking with multisensor out-of-sequence measurements with residual biases. In *Information Fusion (FUSION), 2010 13th Conference on*, pages 1–8, Edinburgh, United Kingdom, Jul. 2010.