

Staggered Scheduling of Sensor Estimation and Fusion for Tracking Over Long-Haul Links

Qiang Liu, *Member, IEEE*, Nageswara S. V. Rao, *Fellow, IEEE*, and Xin Wang, *Member, IEEE*

Abstract—Networked sensing can be found in a multitude of real-world applications. We focus on the communication- and computation-constrained long-haul sensor networks, where sensors are remotely deployed over a vast geographical area to perform certain tasks. Of special interest is a class of such networks where sensors take measurements of one or more dynamic targets and send their state estimates to a remote fusion center via long-haul satellite links. The severe loss and delay over such links can easily reduce the amount of sensor data received by the fusion center, thereby limiting the potential information fusion gain and resulting in suboptimal tracking performance. In this paper, 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 then investigate the effect of such estimation scheduling across different sensors on the spatial-domain fusion performance, where the sensing time epochs across sensors are scheduled in an asynchronous and staggered manner. In particular, the impact of communication delay and loss as well as sensor bias 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, intra-state and inter-state prediction and retrodiction, mean-square-error (MSE) and root-mean-square-error (RMSE) performance, reporting latency.

I. INTRODUCTION

NETWORKED sensing, an enabling core technology for many modern-day applications, has become ubiquitous over the past decade. Systems have been deployed for tasks such as greenhouse gas emissions monitoring using airborne and ground sensors [1], global cyber events processing using cyber sensors distributed over the Internet [15], space exploration using a network of telescopes [18], and target detection and tracking for air and missile defense using radar and sonar [3]. In a typical tracking scenario, a remote sensor with

sensing, data processing, and communication capabilities 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 a fusion center. The fusion center serves to collect data from multiple sensors and fuse the data to obtain global estimates periodically at specified time instants. A “global” estimate derived by the fusion center is expected to be of better quality in terms of its improved accuracy performance over that of the individual sensors, and this effect is often referred to as the fusion gain.

Certain networks are often called long-haul sensor networks because of the long-range communications entailed to cover a very large geographical area, such as a continent or even the entire globe. In this work, we focus on long-haul sensor networks where the sensors send out their time-stamped state estimates over the satellite links to a remote fusion center. 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. For example, the round-trip time (RTT) with a geostationary earth orbit (GEO) satellite is well over a half second [20]. 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. As a result, 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. An upper bound of the loss rate has been derived in [4], above which the estimation error goes unbounded. Some studies, including [13] and [25]–[27], 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 would eventually arrive despite the random delay. More recently, a few studies [11], [19] have exploited retransmission techniques to recover some of the lost messages over time so

Manuscript received April 25, 2016; accepted May 19, 2016. Date of publication June 1, 2016; date of current version July 6, 2016. This work was supported by the Mathematics of Complex, Distributed, Interconnected Systems Program, Office of Advanced Computing Research, U.S. Department of Energy, and the SensorNet Project within the Office of Naval Research, through Oak Ridge National Laboratory managed by UT-Battelle, LLC for the U.S. Department of Energy under Contract No. DE-AC05-00OR22725, and through Stony Brook University under NSF Award CNS 1247924 and NSF Award ECCS 1231800. The associate editor coordinating the review of this paper and approving it for publication was Prof. Okayay Kaynak.

Q. Liu and N. S. V. Rao are with the Oak Ridge National Laboratory, Computer Science and Mathematics Division, Oak Ridge, TN 37831 USA (e-mail: liuq1@ornl.gov; raons@ornl.gov).

X. Wang is with the Department of Electrical and Computer Engineering, State University of New York at Stony Brook, Stony Brook, NY 11794 USA (e-mail: x.wang@stonybrook.edu).

Digital Object Identifier 10.1109/JSEN.2016.2575099

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 [10] so that the final time for fusion is dynamically determined depending on whether enough information has arrived at the fusion center. An information feedback mechanism has been proposed in [9] where fused estimates are fed back to a subset of sensors in order to improve their information quality, and in turn, the overall fusion performance.

Sensing scheduling is a broad term that pertains to any action that can effect changes in the attributes or parameters during the sensing process [6]. Many of the existing studies on wireless sensor networks have considered energy and power savings as an essential task for prolonging the lifespan of the entire network. Formulations such as the myopic- and non-myopic sensor management schemes [8] do not lead to generally applicable solutions as they suffer from the “curse of dimensionality” due to the exponentially increasing computational complexity over time. An algorithm for scheduling and control of passive sensors is similarly proposed in [5] that aims to maximize an information measure in the sensor measurements. An optimization problem for estimation scheduling has been studied in [17] where a uniform scheduling technique is found to be optimal with steady-state sensors and an average error metric. Recently, [12] has proposed an optimization solution that aims to balance the requirements between estimation accuracy and total sensor activations over one time period. However, none of these studies have accounted for varying tracking requirements and are generally not amenable to communication-constrained settings as all the sensor data must be present for the manager to make global decisions.

There have been few studies on temporal-domain scheduling design for improving the tracking performance in the context of state estimation/fusion applications. Our focus in this paper is not on finding the optimal solutions (e.g., minimum errors) for every possible combination of network and sensor conditions in a communication- and computation-constrained environment; rather, of interest is how to opportunistically use these constraints to schedule sensing over time that can potentially improve the estimation and fusion performance. Different from conventional approaches, where the time instants to generate state estimates at the sensors and the fused state at the fusion center coincide, we consider *staggered* estimation and fusion where there exists a time shift – the “staggered time” – between the two. 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 final reporting time. Such scheduling can be carried out across 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 tracking for different types of target trajectories under variable communication loss/delay conditions and sensor bias levels.

The remainder of this paper is organized as follows: Section II is an overview of the system model, from modeling the target and measurements, to generating the estimates at

the sensors and fusing the estimates at the fusion center. Next in Section III, the prediction and retrodiction performed at the fusion center is highlighted, which serves as the basis of staggered scheduling. In Section IV, we present staggered estimation via the technique of intra-state prediction and retrodiction. In Section V, we carry out a study using a simplified trajectory case that demonstrates the major advantages of such opportunistic staggered scheduling in improving estimation performance for both one- and two-sensor scenarios. Simulation results of a maneuver target tracking application are shown and analyzed in Section VI before we conclude the paper in Section VII.

II. SYSTEM MODEL

We present the target and sensor measurement models as well as the estimation and fusion algorithms in this section.

A. Target Model

The most popular target motion models in the tracking domain are the kinematic state models that are obtained by setting a certain derivative of the position to be equal to a zero-mean white noise [2]. The process noise in each model describes the motion uncertainty that complicates the estimation process. We consider a trajectory that consists of two basic types of motion: straight-line and turn movements. The straight line and turn components are described by the continuous white-noise acceleration (CWNA) and coordinated turn (CT) models respectively. Such a combination has been used in the literature, such as in [24]. In this study, we consider tracking that occurs in horizontal dimensions (for both straight-line and turn segments). This is typically the case for motion over land or water; but even for aircraft motion, real trajectories suggest that the vertical motion can be easily decoupled from that in the horizontal plane [16].

1) *Continuous White-Noise Acceleration (CWNA) Model:* The discretized continuous white noise acceleration model is a commonly used motion model in which an object moving in a generic coordinate ξ is assumed to be traveling at a near constant speed. The discrete-time state equation is given by $\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{w}_k$, where (dropping the time index k), $\mathbf{x} = [\xi \ \dot{\xi}]^T$ is a two-dimensional vector representing the position and velocity, and \mathbf{F} is known as the transition matrix $\mathbf{F} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}$, where T is the sampling period of the sensor.¹ The discrete-time process noise is \mathbf{w}_k and its covariance is $\mathbf{Q} = \tilde{q} \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix}$, where \tilde{q} (often assumed to be constant over time) is the power spectral density (PSD) of the underlying continuous-time white stochastic process.

We extend the above model to 2-D tracking with orthogonal coordinates ξ and η . The evolution of the state vector $\mathbf{x} = [\xi \ \dot{\xi} \ \eta \ \dot{\eta}]^T$ is described as

$$\mathbf{x}_{k+1} = \begin{bmatrix} \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \end{bmatrix} \mathbf{x}_k + \mathbf{w}_k, \quad (1)$$

¹A superscript T always denotes the transpose of a vector or matrix.

where \mathbf{w}_k is the process noise whose covariance matrix is given by

$$\mathbf{Q}_k = \begin{bmatrix} \tilde{q}_\xi \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix} & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & \tilde{q}_\eta \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix} \end{bmatrix}. \quad (2)$$

2) *Target Maneuver*: The second type of motion occurs when the target performs a maneuver (i.e., a turn). A turn usually follows a pattern known as *coordinated turn* (CT), which is characterized by a near constant turn rate and near constant speeds along both coordinates. The turn rate Ω is incorporated into the motion model by augmenting the state vector for a horizontal motion model: $\mathbf{x} = [\xi \ \dot{\xi} \ \eta \ \dot{\eta} \ \Omega]^T$, which gives rise to the discretized CT model [2], given by

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & \frac{\sin \Omega(k)T}{\Omega(k)} & 0 & -\frac{1 - \cos \Omega(k)T}{\Omega(k)} & 0 \\ 0 & \cos \Omega(k)T & 0 & -\sin \Omega(k)T & 0 \\ 0 & \frac{1 - \cos \Omega(k)T}{\Omega(k)} & 1 & \frac{\sin \Omega(k)T}{\Omega(k)} & 0 \\ 0 & \sin \Omega(k)T & 0 & \cos \Omega(k)T & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times \mathbf{x}_k + \mathbf{w}_k, \quad (3)$$

and the covariance matrix of the process noise \mathbf{w}_k is

$$\mathbf{Q}_k = \begin{bmatrix} \tilde{q}_\xi \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix} & \mathbf{0}_{2 \times 2} & 0 \\ \mathbf{0}_{2 \times 2} & \tilde{q}_\eta \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix} & 0 \\ 0 & 0 & \tilde{q}_\Omega T \end{bmatrix}. \quad (4)$$

In contrast to the CWNA model, the CT model is a nonlinear one if the turn rate Ω is not a known constant. In practice, the linear acceleration noise PSD levels in both dimensions are assumed to be equal; i.e., $\tilde{q}_\xi = \tilde{q}_\eta$. The general guidelines for selecting appropriate levels of these noise parameters can be found in [2].

B. Sensor Measurement Model

A sensor collects measurements of the target range r and azimuth angle a according to the following measurement model [24]:

$$\mathbf{z} = \begin{bmatrix} r \\ a \end{bmatrix} = \begin{bmatrix} \sqrt{\xi^2 + \eta^2} \\ \tan^{-1} \left(\frac{\eta}{\xi} \right) \end{bmatrix} + \begin{bmatrix} w_r \\ w_a \end{bmatrix} \quad (5)$$

where w_r and w_a are independent zero mean Gaussian noises with standard deviations σ_r and σ_a , respectively. Note that this measurement has been normalized to the sensor's own known location.

C. Generating the State Estimates

1) *Conversion of Measurements From Polar to Cartesian Coordinate*: In practice, the measurements are often reported in polar coordinates (as in Eq. (5)) with respect to the

sensor location, whereas common motion models are given in Cartesian coordinates. Therefore, it is necessary that a sensor first converts the polar measurements to Cartesian ones before generating its state estimates. The standard conversion formulas are

$$z_\xi = r \cos a \quad z_\eta = r \sin a, \quad (6)$$

where z_ξ and z_η denote the transformed measurement components along ξ and η axes respectively. However, it is noted that this conversion only applies in restricted conditions as it can introduce a bias after the conversion. A more general *unbiased* conversion rule is given by applying a correction factor as follows [14]:

$$z_\xi^u = e^{\sigma_a^2/2} r \cos a \quad z_\eta^u = e^{\sigma_a^2/2} r \sin a, \quad (7)$$

where σ_a is the standard deviation of the polar azimuth measurement.

2) *Kalman Filtering*: The goal of a state estimator is to extract the state information \mathbf{x} from measurement \mathbf{z} that is corrupted by noise; this is done by running a filter that sequentially outputs the state estimate $\hat{\mathbf{x}}$ and its associated error covariance matrix $\mathbf{P} = \mathbb{E}[(\hat{\mathbf{x}} - \mathbf{x})(\hat{\mathbf{x}} - \mathbf{x})^T]$. In most recursive state estimators, each step is comprised of two distinct phases, namely, “predict” and “update” [21], where the new measurement is used to update the predicted estimate in the latter phase.

The Kalman filters (KFs), the most well-known state estimators, are linear minimum-mean-square-error (LMMSE)-optimal as the trace of \mathbf{P} – characterizing the estimation error – at each step is minimized. The detailed steps and their explanations can be found in [21]. If both system and measurement models are well-defined linear functions, KFs are guaranteed to yield the optimal estimates. In the CWNA model presented earlier, the transition matrix is well-defined and stable over time. Despite the nonlinearity in the polar measurement equation as in Eq. (5), a sensor can regard the transformed Cartesian measurement in Eq. (6) or (7) as the direct measurement, which corresponds to a linear mapping from states to measurements.

On the other hand, when the state transition matrix contains elements of the current state, as in the CT model, the system dynamics are nonlinear. As such, extended Kalman filters (EKFs) can be used to approximate the nonlinearity. This is achieved by using the Jacobians of the nonlinear state transition and measurement functions at the current estimate as the state transition and measurement matrices, respectively, to be plugged into the original KF equations. This process essentially linearizes the non-linear function around the current estimate.

3) *Interacting Multiple Models*: In many practical scenarios, the system characteristics can change over time. As an example, a fighter jet, which normally proceeds with stable flight dynamics, might commence rapid maneuvers when approached by a hostile missile. Such varying system characteristics are not easily described by a single model, thereby calling for a multiple-model approach. A multiple model (MM) can serve as a versatile tool for adapting the state estimation process in dynamic systems where a target

can undergo different types of motion at different times. In particular, the interacting multiple-model (IMM) estimator is considered one of the most cost-effective dynamic MM algorithms. In IMM, at any time, the system state is assumed to be in a number of possible modes that are described by their probabilities. The structure of the system and/or the statistics of the noise can be different from mode to mode. The transition probabilities between modes from one estimation epoch to the next are assumed to follow a Markov chain. For each mode, the underlying filtering process is performed in the same way as described earlier, with the addition of evaluating the probabilities of different modes and interacting and mixing all the modes to generate an overall state estimate and error covariance. The equations and a detailed discussion of design parameters and implementation issues can be found in [2].

In our two-sensor settings, since KF and EKF can be used for CWNA and CT models respectively, we consider Sensor 1 uses an IMM estimator that contains two modes, namely, KF and EKF, with appropriately selected noise levels. On the other hand, in [2], an argument has been made that during maneuvers, if the number of samples are relatively small (say, less than 10), a stand-alone KF with high process noise levels can be used for the maneuver. As such, we consider a heterogenous sensor setting where Sensor 2 uses a KF throughout its filtering process; in other words, it doesn't distinguish between different types of motion. For both sensors, state estimates $\hat{\mathbf{x}}$ and the associated error covariances \mathbf{P} are generated periodically and sent out, in a timely fashion, to the remote fusion center. Besides, when the latter combines the two estimates together according to the fusion algorithms to be presented below, the component estimates should be of the same dimension. At the time of fusion, the fusion center can simply drop the turn rate component from Sensor 1 if necessary.

D. Fusers

It is a well known fact that the common process noise in measuring the motion of a target results in correlation among estimates generated by multiple sensors. The error cross-covariance is the term that describes this spatial correlation. However, it 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 in closed forms with no cross-covariance calculation needed.

1) *Track-to-Track Fuser Without Cross-Covariance*: In tracking applications, the track-to-track fuser (T2TF) is a linear fuser that is optimal in the linear minimum mean-square error (LMMSE) sense. In general, the fused state estimate $\hat{\mathbf{x}}_F$ and its error covariance \mathbf{P}_F are defined for two sensors [2] as:

$$\hat{\mathbf{x}}_F = \hat{\mathbf{x}}_1 + (\mathbf{P}_1 - \mathbf{P}_{12})(\mathbf{P}_1 + \mathbf{P}_2 - \mathbf{P}_{12} - \mathbf{P}_{21})^{-1}(\hat{\mathbf{x}}_2 - \hat{\mathbf{x}}_1) \quad (8)$$

$$\mathbf{P}_F = \mathbf{P}_1 - (\mathbf{P}_1 - \mathbf{P}_{12})(\mathbf{P}_1 + \mathbf{P}_2 - \mathbf{P}_{12} - \mathbf{P}_{21})^{-1}(\mathbf{P}_1 - \mathbf{P}_{21}) \quad (9)$$

where $\hat{\mathbf{x}}_i$ and \mathbf{P}_i are the state estimate 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, when the sensor errors are correlated and 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 be reduced to a simple convex combination of the state estimates:

$$\mathbf{P}_F = (\mathbf{P}_1^{-1} + \mathbf{P}_2^{-1})^{-1} \quad (10)$$

$$\hat{\mathbf{x}}_F = \mathbf{P}_F(\mathbf{P}_1^{-1}\hat{\mathbf{x}}_1 + \mathbf{P}_2^{-1}\hat{\mathbf{x}}_2). \quad (11)$$

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 ellipses of \mathbf{P}_F are found to always contain the intersection of the ellipses for \mathbf{P}_1 and \mathbf{P}_2 for all possible choices of \mathbf{P}_{12} [7], [22]. The intersection is characterized by the convex combination of sensor covariances:

$$\mathbf{P}_F = (\omega_1 \mathbf{P}_1^{-1} + \omega_2 \mathbf{P}_2^{-1})^{-1} \quad (12)$$

$$\hat{\mathbf{x}}_F = \mathbf{P}_F (\omega_1 \mathbf{P}_1^{-1} \hat{\mathbf{x}}_1 + \omega_2 \mathbf{P}_2^{-1} \hat{\mathbf{x}}_2), \quad \omega_1 + \omega_2 = 1 \quad (13)$$

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

Recently, [23] has 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:

$$\omega_1 = \frac{D(p_1, p_2)}{D(p_1, p_2) + D(p_2, p_1)} \quad (14)$$

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:

$$D(p_i, p_j) = \frac{1}{2} \left[\ln \frac{|\mathbf{P}_j|}{|\mathbf{P}_i|} + \mathbf{d}_X^T \mathbf{P}_j^{-1} \mathbf{d}_X + \text{Tr}(\mathbf{P}_i \mathbf{P}_j^{-1}) - k \right] \quad (15)$$

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 T2TF with unavailable cross-covariances.

E. Target Trajectory

The initial state of the target in Cartesian coordinates (with the position in meters and velocity in m/s) is set to be [24] $\mathbf{x}(0) = [x(0) \ \dot{x}(0) \ y(0) \ \dot{y}(0)]^T = [0 \ 0 \ 20000 \ 250]^T$. At $t = 60$ s, the test target starts to take a left turn at a turn rate of $2^\circ/\text{s}$ for 30 s, and then continues straight until $t = 150$ s. The sampling rate of the sensors is once every two seconds, i.e., $T = 2$ s.

III. PREDICTION AND RETRODICTION BY THE FUSION CENTER

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.

A. Prediction by the Fusion Center

In this work, we mostly concern ourselves with prediction performed not by a sensor during its regular recursive filtering, but rather by the fusion center. The purpose is largely different even though the two may share the same form of “prediction” equations. 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 constraints. Due to the system uncertainty characterized by process noise, 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 at the sensors in order to maintain desired tracking performance). Nevertheless, to achieve the fusion gain, at the fusion center, often it is still preferable to use predicted estimates for a sensor rather than discard the sensor’s potential information altogether [11].

B. Retrodiction by the Fusion Center

To recall, estimation of a target state at a particular time based on data 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. 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 [13], [25], [26]. However, in our design, the fusion center opportunistically interpolates the missing estimates – that is, to “fill in the blanks” – from prior estimates using prediction, and subsequent ones 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 can reduce the chance of missing the reporting deadline.

C. Selection of Prediction and Retrodiction Algorithms With Multiple Models

In the context of possibly heterogenous and multiple system models utilized by individual sensors, the fusion center faces the question of which model(s) to use when deriving its predicted/retrodicted values. As a matter of fact, a sensor is not likely to send every filtering parameter update to the fusion center (considering, e.g., the complexity of IMM estimators); therefore, the fusion center can only use its best guess, or out of practicality, use simplified models to interpolate the missing values. Although data thereby generated may not match the quality level of that of the sensor originals, they might still provide enough information to drive up the fusion gain.

We consider the fusion center uses a linear form of prediction and retrodiction (i.e., KF-based) for simplicity, both of which can be easily computed with closed-form expressions. In particular, the fixed-interval Rauch-Tung-Striebel (RTS) retrodiction algorithm [21] is adopted that can help fill in missing data as well as update existing ones using subsequent arrivals.

IV. STAGGERED ESTIMATION: AN OVERVIEW

We explore *staggered estimation* that exploits the temporal relationship between adjacent estimates in order to improve the estimation performance. To formulate the estimation and fusion process, we consider that a stream of globally fused estimates are 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. Given 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 – if necessary – to form component state estimates for fusion. For a given sensor, depending on what estimates sent from this sensor have been received, the fusion center may report one of the following types of estimates, corresponding to 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 steps² 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

²If the preceding estimate for step $n-1$ is available, then one-step prediction is in place; otherwise, multi-step prediction is necessary.

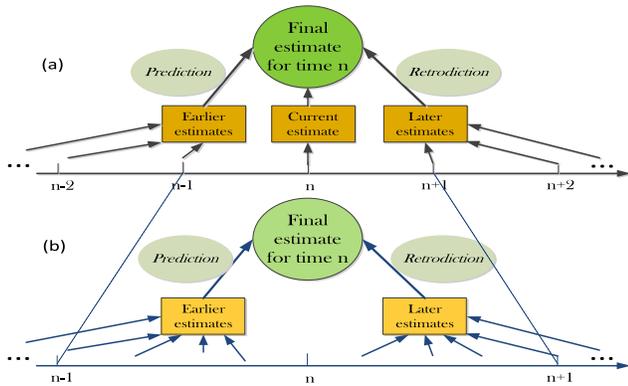


Fig. 1. Prediction and retrodiction: The estimate at time n is to be obtained. (a) Conventionally, prediction and retrodiction occur over integer multiples of the estimation interval. (b) In staggered scheduling, prediction and retrodiction can be carried out over a fraction of the estimation interval.

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 conventional prediction and retrodiction techniques are schematically shown in Fig. 1(a), where the estimation interval T (along with its integer multiples) serves as the base time unit 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 sensors generate the estimates, we consider staggered scheduling 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 finalizes and reports the fused estimates. 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.³

In Fig. 2, an example consisting of three different estimation schedules is shown. In the figure, the red dotted lines denote the common time instants of interest for the fusion center (i.e., the time instants for which 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.⁴ In (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

³Of course, the extension of this intra-state filtering can be realized by superimposing an estimation interval (and its integer multiples) on top of the fractional period τ .

⁴To avoid confusion, the term “fusion time” refers to the time of interest for which a sensor state estimate needs to be generated by the fusion center (because of the staggered scheduling); whereas “reporting time” refers to the time when final fusion and reporting occurs.

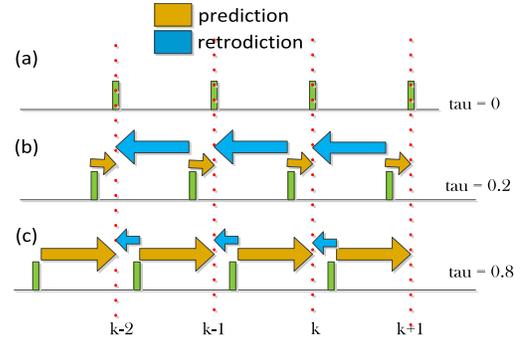


Fig. 2. Staggered estimation scheduling: (a) the standard schedule; (b) staggered estimation, where the sensor takes measurements $0.2T$ earlier than the subsequent fusion time; and (c) staggered estimation, where the sensor takes measurements $0.2T$ later than the preceding fusion time. The τ values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

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 (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 in the figure as the gap – normalized to the estimation interval T – 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 simple conceptually, the effects of this staggered scheduling on estimation and fusion performance are not readily predictable. Next we investigate its potential benefits for both one-sensor and two-sensor estimation scenarios under variable communication loss and delay conditions during straight-line motion.

V. STAGGERED ESTIMATION AND FUSION

In this section, we carry out a study of the staggered estimation and fusion under a simplified single-model motion scenario. To achieve this, we relax the target and sensor measurement settings described in Section II as follows. We consider only the straight-line segment – as described by the CWNA model – in one generic coordinate ζ with the estimation interval set to be $T = 1$ s and the noise PSD $\tilde{q}_\zeta = 1$ m^2/s^3 . The sensor measurement model has been simplified in which the sensor directly measures the Cartesian position of the target and hence the measurement $z_k = \mathbf{H}\mathbf{x}_k + v_k$ is available, where $\mathbf{H} = [1 \ 0]$ is the measurement matrix, and the Gaussian measurement noise v has autocorrelation $\mathbb{E}[v_k v_j] = R \delta_{kj} \triangleq \sigma_v^2 \delta_{kj}$, where $\delta_{(\cdot)}$ is the Kronecker delta function. Both sensors have a $\sigma_v = 20$ m. This study enables us to focus on the effect of staggered scheduling on the estimation and fusion performance in a simplified setting, without considering the impact of such factors as target model uncertainty and sensor heterogeneity, among others.

A. Estimation Performance With One Sensor

We first explore the effect of staggered scheduling on one-sensor estimation. Had the communication between the sensor and the fusion center been perfect, with the standard synchronous scheduling, the fusion center would have simply “taken 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 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:

$$f(t) = \frac{1}{\mu} \exp^{-\frac{t-T_I}{\mu}}, \text{ for } t \geq T_I, \quad (16)$$

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 depends on instantaneous link conditions. We analyze the probabilities of generating different types of estimates by a certain deadline and show the impact of scheduling on estimation performance.

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 sensor estimate 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 fixed initial delay T_I , 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: To have up to l ($l \geq 1$) rounds of retrodiction, the reporting lag D should satisfy the following condition:

$$lT + T_I - \tau \leq D < (l+1)T + T_I - \tau. \quad (17)$$

Without loss of generality, in the following analysis, T and D are given as 1 s and 1.5 s respectively, with T_I set as 0.5 s and μ 0.3 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 $l = 1$.

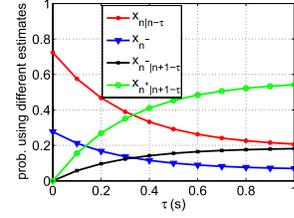


Fig. 3. Probabilities of using different types of estimates at the deadline with variable staggered estimation intervals τ , where $p = 0.25$, $T = 1$, and $D = 1.5$ s.

Given the link statistics introduced earlier, the probability that a sensor estimate is successfully received by the fusion center within time t after being generated is $(1-p)F(t)$. It is easy to verify that the amounts 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:

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

Similar notations for these types of estimates were first introduced in Section IV, 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 $\hat{\mathbf{x}}_{n+|n+1-t}$ 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, $\hat{\mathbf{x}}_{n|n-t}$ and $\hat{\mathbf{x}}_{n-|n+1-t}$, change in reverse directions too as τ shifts, and it is not immediately clear which of the two has overall better accuracy performance [11].

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 types of estimates. In what follows, we explore the error profiles with staggered scheduling under perfect communications. Then we will combine them with the above

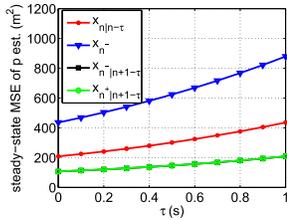


Fig. 4. Steady-state position estimate mean-square-errors (MSEs) with variable staggered interval τ values.

probabilistic analysis to derive the approximate estimation error performance.

B. Quantitative Results

1) *Approximate Estimation Error*: The fusion center applies the Rauch-Tung-Streibel (RTS) retrodiction algorithm [21] to obtain the retrodicted state estimates. With the previously established models, the steady-state behavior of the sensor estimate can be found analytically via Riccati equation recursion or more conveniently from simulations. 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 the cases b) and c) of the previous 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 increasing 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 an 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, under our settings, the frequency that a sensor communicates its estimates (but with the same estimation frequency on tap) can be reduced by half without causing noticeable performance degradation.

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, in which the *actual* error performance could be worse. The results are plotted in

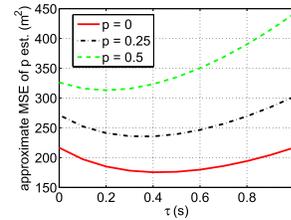


Fig. 5. Approximate position estimate MSEs with variable staggered interval τ and loss rate p values.

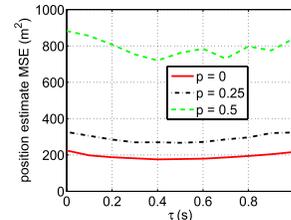


Fig. 6. Actual position estimate MSEs with variable staggered interval τ and loss rate p values.

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.

2) *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 can 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% higher 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.

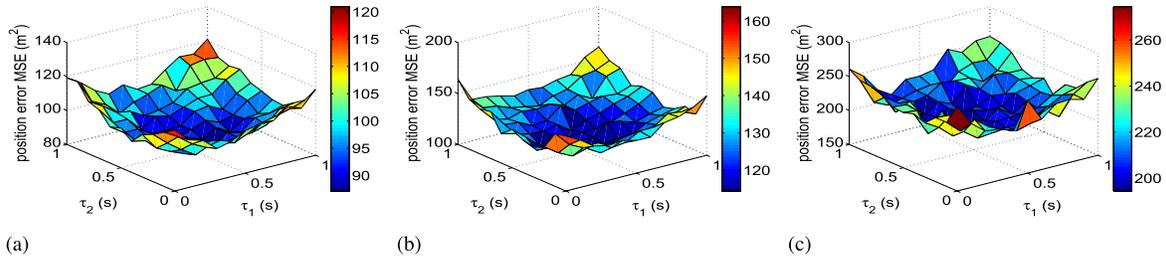


Fig. 7. Actual position estimate MSE performance with two-sensor T2TF fusion and variable staggered interval τ and loss rate p values: (a) $p = 0$; (b) $p = 0.25$; and (c) $p = 0.5$.

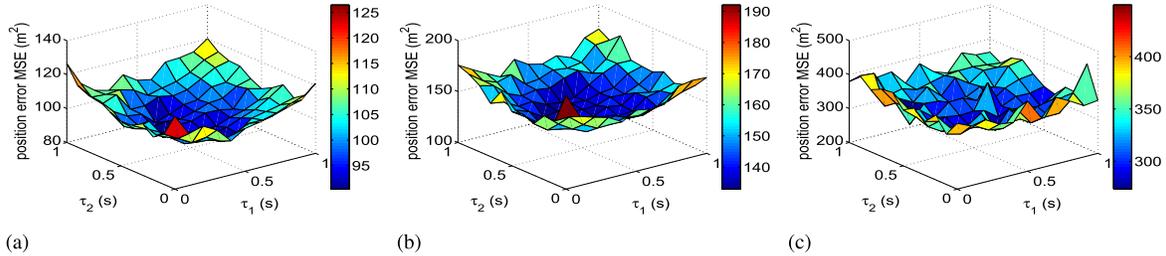


Fig. 8. Actual position estimate MSE performance with two-sensor fast-CI fusion and variable staggered interval τ and loss rate p values: (a) $p = 0$; (b) $p = 0.25$; and (c) $p = 0.5$.

C. Estimation and Fusion Performance With Two Sensors

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 previous 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 the position estimate MSEs for the linear T2TF and the fast-CI fuser are plotted respectively.

From the figures, all generated three-dimensional surfaces resemble a sheet with downward-curved center regions, the extension of the earlier one-sensor estimation performance. We observe that for almost 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 over half of those values with one sensor only, reflecting the effect of the common process noise and cross-covariance.

Although not as easily seen in the figures here, a more “microscopic” examination of the numerical results reveals the effect of cross-covariance in staggered scheduling.

⁵Note 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. (11) but with an inflated error covariance matrix (increased by a factor of 2).

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 between increases, so does the reduction of correlation.

VI. PERFORMANCE WITH TARGET MANEUVERS

In this section we evaluate the impact of variable staggered schedules on estimation and fusion performance in tracking a target undergoing a more complex set of motions as described in Section II.

A. Sensor Behaviors

Whereas Sensor 1 uses an IMM estimator consisting of KF and EKF components whose parameters match the noise profiles of the CWNA and CT models respectively, Sensor 2 uses a KF throughout with a much higher noise density level to account for the uncertainty in the target trajectory. In Fig. 9(a), we observe the following behaviors of the individual sensors: The IMM estimator at Sensor 1 yields more accurate estimates – compared to the KF estimator at Sensor 2 – initially during the straight-line motion; but after the turn begins at $t = 60$ s, its error gradually increases and at around $t = 80$ s, shoots up rapidly till after the maneuver ends at $t = 90$ s. Afterward, the error decreases, less precipitately compared to the previous phase, and doesn’t fall back to the pre-maneuver level until around $t = 140$ s.

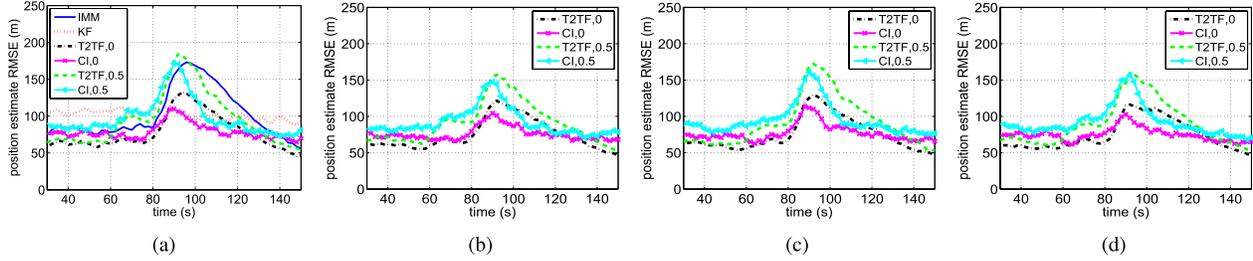


Fig. 9. Position estimate RMSE performance with variable staggered intervals. Deadline is 2 s: (a) $\tau_1 = \tau_2 = 0$ s; (b) $\tau_1 = 0$ s, $\tau_2 = 1$ s; (c) $\tau_1 = 1$ s, $\tau_2 = 0$ s; and (d) $\tau_1 = 1$ s, $\tau_2 = 1$ s.

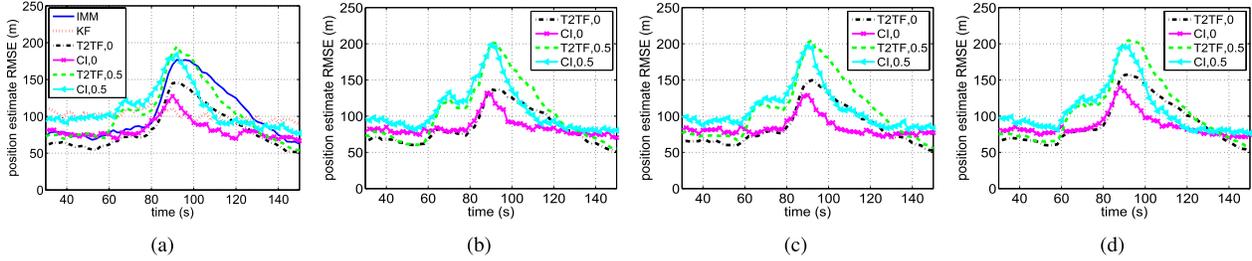


Fig. 10. Position estimate RMSE performance with variable staggered intervals. Deadline is 1 s. (a) $\tau_1 = \tau_2 = 0$ s; (b) $\tau_1 = 0$ s, $\tau_2 = 1$ s; (c) $\tau_1 = 1$ s, $\tau_2 = 0$ s; and (d) $\tau_1 = 1$ s, $\tau_2 = 1$ s.

The relatively poor performance of the IMM estimator during and after maneuver in this case is due mainly to the selection of parameters for component modes. On the other hand, the error of the KF filter output at Sensor 2 remains stable throughout the process, higher than that of the IMM during the initial straight-line motion, but much lower as the other experiences inflated errors during and after the turn. Such sensor heterogeneity will be reflected in the effect of sensing scheduling on fusion as demonstrated below.

B. Staggered Sensing Scheduling Without Sensor Bias

We now shift our focus to the fusion performance. In Fig. 9(a), the performance of the T2TF and CI fusers under conventional scheduling is shown for both $p = 0$ and $p = 0.5$ cases. Note the selection of fusers or different link profiles does not change the sensor estimation performance. As has been observed in the case study in Section V, the CI fuser yields somewhat higher errors during the initial straight-line motion; however, during and shortly after the turn, the T2TF experiences higher errors than its CI counterpart. Also, when loss is severe enough, say, when $p = 0.5$, for a short period of time after the turn is initiated, between $t = 80$ s and 90 s, both fusers yield even worse performance than the individual sensors. In this case, the CI fuser is able to pull back from its elevated errors much faster than the T2TF, as can be observed by the somewhat narrower error peak around $t = 90$ s.

Next we consider staggered scheduling that spans different choices of τ_1 and τ_2 values. In Fig. 9, the standard scheduling is shown in (a), where $\tau_1 = \tau_2 = 0$; alternately, one or both of the τ 's are set to 1 s in (b)-(d). The deadline is chosen to be $D = 2$ s. From the plots, while the improvements during the straight-line segments are not very obvious, we can easily

observe that when τ_1 is 1 s, regardless of the τ_2 choices, the peak levels of both T2TF and CI drop significantly, especially for the $p = 0.5$ case. For example, the position RMSE is about 20% lower when $\tau_1 = 1$ s compared to that following the default scheduling. This demonstrates by staggering out the estimation times of the more error-prone Sensor 1, the fusion center can reduce the uncertainty in target motion by applying intra-state prediction and retrodiction. In contrast, different choices of τ_2 don't seem to affect the fusion performance as much. Besides, the improvement at the beginning of the maneuver, if any, is rather limited, since even using retrodiction doesn't help much during the phase of inflating errors at Sensor 1.

In Fig. 10, the same set of plots are shown, with the exception that the deadline is decreased to one half of the previous value, now set at 1 s. The reduced deadline apparently reduces the probability of receiving a sensor estimate at the final reporting time; but more importantly, it also changes the dynamics of the staggered estimation process by reducing the retrodiction opportunities and increasing the chances of using prediction only. From the plots, it's easily observed that the previous $\tau_1 = 1$ s cases now experience even higher errors than the standard scheduling during and after the maneuver. Under these settings, the prediction steps are long, and due to the short deadline, there is not enough time for the subsequent sensor estimate to arrive, thereby greatly reducing the chances for applying retrodiction that can potentially decrease the estimation error during and after the maneuver.

C. Staggered Sensing Scheduling With Sensor Bias

Next we study the case where the IMM estimator at Sensor 1 experiences measurement bias. In Fig. 11, the plots are shown for all above staggered schedules where there is a consistent positive bias term with the range and azimuth angle

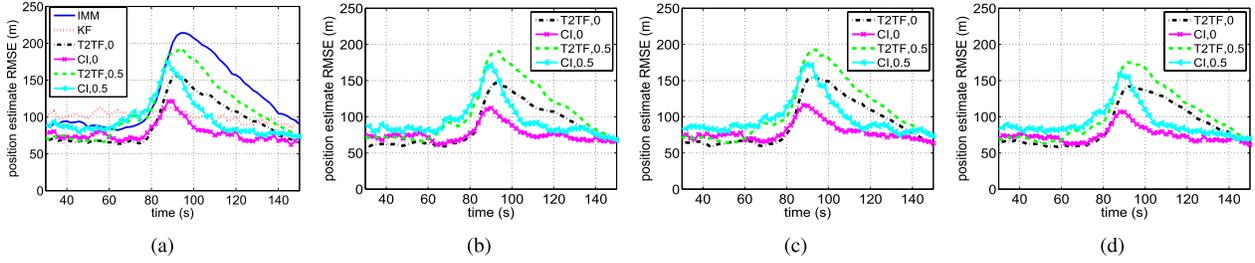


Fig. 11. Position estimate RMSE performance with variable staggered intervals and steady measurement bias at Sensor 1 (IMM). Deadline is 2 s: (a) $\tau_1 = \tau_2 = 0$ s; (b) $\tau_1 = 0$ s, $\tau_2 = 1$ s; (c) $\tau_1 = 1$ s, $\tau_2 = 0$ s; and (d) $\tau_1 = 1$ s, $\tau_2 = 1$ s.

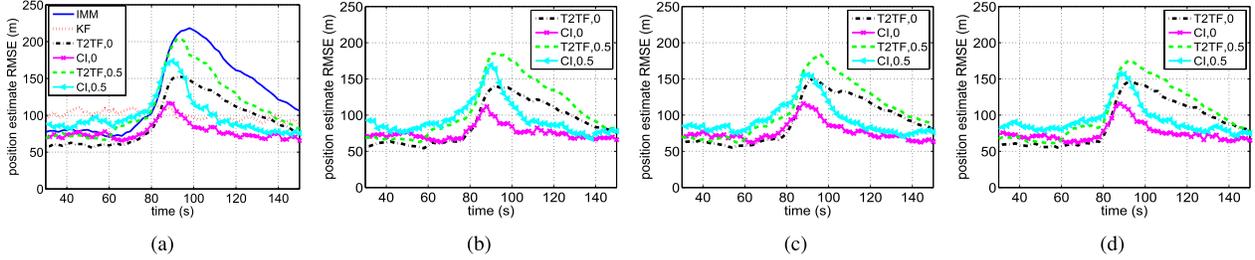


Fig. 12. Position estimate RMSE performance with variable staggered intervals and increasing measurement bias at Sensor 1 (IMM). Deadline is 2 s: (a) $\tau_1 = \tau_2 = 0$ s; (b) $\tau_1 = 0$ s, $\tau_2 = 1$ s; (c) $\tau_1 = 1$ s, $\tau_2 = 0$ s; and (d) $\tau_1 = 1$ s, $\tau_2 = 1$ s.

measurement bias values as $\sigma_r/\sqrt{5}$ and $\sigma_a/\sqrt{20}$ respectively. From the plots, the performances with staggered scheduling largely follow the trends discovered earlier in Fig. 9 and cases with $\tau_1 = 1$ s yield improved tracking accuracy levels compared to others. In addition, the performance gap between the T2TF and CI fusers after the maneuver is over becomes more pronounced, especially for the case where $p = 0.5$. This demonstrates that to some extent, the CI fuser is more tolerant of bias than its T2TF counterpart as the distances between the estimates (which reflects the bias) also appear in the weights which can somewhat mitigate the effect of more biased estimates.

Finally, we consider the case in which an increasing measurement bias occurs at Sensor 1, and the range and azimuth biases rise steadily from 0 to $3\sigma_r/\sqrt{5}$ and $3\sigma_a/\sqrt{20}$ respectively toward the end of the 150-second trajectory. From the plots in Fig. 12, the estimates from Sensor 1 are now even worse than before, as the errors never return to the pre-maneuver level toward the end of the trajectory. However, both fusers are able to retain similar tracking performance as in the previous steady-bias case, even when only half of the original sensor data are available.

VII. CONCLUSIONS

We investigated ways the fusion center can exploit staggered scheduling and opportunistically apply intra-state prediction and retrodiction to improve the fusion performance under variable link-level loss and delay conditions. Tracking performances of a maneuvering target under variable network and sensor profiles as well as sensing schedules demonstrate the major potentials of such staggered estimation in reducing the tracking errors in the presence of communication and computation constraints. Future extensions of this work

include sensing scheduling with more sensors and in a more dynamic environment where the fusion center keeps track of the evolving system dynamics and makes adaptive changes to its existing sensing schedules.

REFERENCES

- [1] *Earth Systems Research Laboratory, National Oceanic and Atmospheric Administration*, accessed on Apr. 25, 2016. [Online]. Available: <http://www.esrl.noaa.gov/gmd/ccgg/index.html>
- [2] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. Storrs, CT, USA: YBS Publishing, 2011.
- [3] W. J. Boord and J. B. Hoffman, *Air and Missile Defense Systems Engineering*. Boca Raton, FL, USA: CRC Press, 2016.
- [4] A. Chiuso and L. Schenato, "Information fusion strategies and performance bounds in packet-drop networks," *Automatica*, vol. 47, pp. 1304–1316, Jul. 2011.
- [5] T. Hanselmann, M. Morelande, B. Moran, and P. Sarunic, "Sensor scheduling for multiple target tracking and detection using passive measurements," in *Proc. 11th Int. Conf. Inf. Fusion*, Jun./Jul. 2008, pp. 1–8.
- [6] A. O. Hero and D. Cochran, "Sensor management: Past, present, and future," *IEEE Sensors J.*, vol. 11, no. 12, pp. 3064–3075, Dec. 2011.
- [7] S. J. Julier and J. K. Uhlmann, *General Decentralized Data Fusion With Covariance Intersection* (Handbook of Data Fusion). Boca Raton, FL, USA: CRC Press, 2001.
- [8] C. Kreucher, A. O. Hero, K. Kastella, and D. Chang, "Efficient methods of non-myopic sensor management for multitarget tracking," in *Proc. 43rd IEEE Conf. Decision Control*, vol. 1, Dec. 2004, pp. 722–727.
- [9] Q. Liu, X. Wang, and N. S. V. Rao, "Information feedback for estimation and fusion in long-haul sensor networks," in *Proc. 17th Int. Conf. Inf. Fusion (FUSION)*, Salamanca, Spain, Jul. 2014, pp. 1–8.
- [10] Q. Liu, X. Wang, and N. S. V. Rao, "Fusion of state estimates over long-haul sensor networks with random loss and delay," *IEEE/ACM Trans. Netw.*, vol. 23, no. 2, pp. 644–656, Apr. 2015.
- [11] Q. Liu, X. Wang, N. S. V. Rao, K. Brigham, and B. V. K. V. Kumar, "Effect of retransmission and retrodiction on estimation and fusion in long-haul sensor networks," *IEEE/ACM Trans. Netw.*, vol. 24, no. 1, pp. 449–461, Feb. 2016.
- [12] S. Liu, M. Fardad, E. Masazade, and P. K. Varshney, "Optimal periodic sensor scheduling in networks of dynamical systems," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3055–3068, Jun. 2014.

- [13] M. Mallick and K. Zhang, "Optimal multiple-lag out-of-sequence measurement algorithm based on generalized smoothing framework," in *Proc. SPIE, Signal Data Process. Small Targets*, San Diego, CA, USA, Apr. 2005, p. 5913-08.
- [14] M. Longbin, S. Xiaoquan, Z. Yiyu, and Y. Bar-Shalom, "Unbiased converted measurements for tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 3, pp. 1023–1027, Jul. 1998.
- [15] R. N. Murty and M. Welsh, "Towards a dependable architecture for Internet-scale sensing," in *Proc. 2nd Workshop Hot Topics Syst. Dependability*, vol. 2. Seattle, WA, USA, Nov. 2006, p. 8.
- [16] N. Naba and R. H. Bishop, "Validation and comparison of coordinated turn aircraft maneuver models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 36, no. 1, pp. 250–259, Jan. 2000.
- [17] R. Niu, P. K. Varshney, K. Mehrotra, and C. Mohan, "Temporally staggered sensors in multi-sensor target tracking systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 3, pp. 794–808, Jul. 2005.
- [18] University of Southampton, "Global network of new-generation telescopes will track astrophysical events as they happen," *Asian J. Sci. Res.*, vol. 4, no. 2, p. 185, Jan. 2011.
- [19] N. S. V. Rao, K. Brigham, B. V. K. V. Kumar, Q. Liu, and X. Wang, "Effects of computing and communications on state fusion over long-haul sensor networks," in *Proc. 15th Int. Conf. Inf. Fusion (FUSION)*, Jul. 2012, pp. 1570–1577.
- [20] D. Roddy, *Satellite Communications*. New York, NY, USA: McGraw-Hill, 2006.
- [21] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. New York, NY, USA: Wiley, 2006.
- [22] J. K. Uhlmann, "Covariance consistency methods for fault-tolerant distributed data fusion," *Inf. Fusion*, vol. 4, no. 3, pp. 201–215, Sep. 2003.
- [23] Y. Wang and X. R. Li, "Distributed estimation fusion with unavailable cross-correlation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 1, pp. 259–278, Jan. 2012.
- [24] T. Yuan, Y. Bar-Shalom, and X. Tian, "Heterogeneous track-to-track fusion," in *Proc. 14th Int. Conf. Inf. Fusion (FUSION)*, Jul. 2011, pp. 1–8.
- [25] K. Zhang, X. R. Li, and Y. Zhu, "Optimal update with out-of-sequence measurements," *IEEE Trans. Signal Process.*, vol. 53, no. 6, pp. 1992–2004, Jun. 2005.
- [26] S. Zhang and Y. Bar-Shalom, "Optimal update with multiple out-of-sequence measurements with arbitrary arriving order," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 4, pp. 3116–3132, Oct. 2012.
- [27] S. Zhang, Y. Bar-Shalom, and G. Watson, "Tracking with multisensor out-of-sequence measurements with residual biases," *J. Adv. Inf. Fusion*, vol. 6, no. 1, pp. 3–23, Jun. 2011.

Qiang Liu (S'07–M'14) received the Ph.D. degree in electrical and computer engineering from Stony Brook University, Stony Brook, NY, in 2014. He is currently a Research Associate with the Oak Ridge National Laboratory, Computer Science and Mathematics Division, Oak Ridge, TN. His research areas include networked data and information fusion, signal/target detection and estimation, statistical signal processing, wireless sensor networks, cognitive radios, cyberphysical systems, transport protocols, high-performance computing, and software defined networking. He has published over a dozen peer-reviewed papers in prestigious conference proceedings and journals, and has reviewed dozens of works by others. He has been invited to present his work at various venues and is a recipient of several travel grant awards.

Nageswara S. V. Rao (M'90–SM'97–F'08) received the B.Tech. degree in electronics and communications engineering from the National Institute of Technology, Warangal, India, in 1982, the M.E. degree in computer science and automation from the Indian Institute of Science, Bangalore, India, in 1984, and the Ph.D. degree in computer science from Louisiana State University in 1988. In 1993, he joined the Oak Ridge National Laboratory, Computer Science and Mathematics Division, where he is currently a Corporate Fellow. He was a Technical Director of the C2BMC Knowledge Center at the Missile Defense Agency from 2008 to 2010. He published over 350 technical conference and journal papers in the areas of sensor networks, information fusion, and high-performance networking. He received the 2005 IEEE Technical Achievement Award and the 2014 Research and Development 100 Award.

Xin Wang (S'98–M'01) received the Ph.D. degree in electrical and computer engineering from Columbia University, New York, NY. She is currently an Associate Professor with the Department of Electrical and Computer Engineering, State University of New York at Stony Brook, Stony Brook, NY. She was a member of the Technical Staff in the area of mobile and wireless networking with Bell Laboratories Research, Lucent Technologies, Holmdel, NJ, and an Assistant Professor with the Department of Computer Science and Engineering, State University of New York at Buffalo, Buffalo, NY. Her research interests include algorithm and protocol design in wireless networks and communications, mobile and distributed computing, and networked sensing and detection. She has served on the executive committee and technical committee of numerous conferences and funding review panels, and served as an Associate Editor of the IEEE TRANSACTIONS ON MOBILE COMPUTING. She received the NSF Career Award in 2005 and the ONR Challenge Award in 2010.