Fusion Performance in Long-Haul Sensor Networks with Message Retransmission and Retrodiction

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

Katharine Brigham and B. V. K. Vijaya Kumar
Carnegie Mellon University
Pittsburgh, PA 15213
Email: {kbrigham,kumar}@ece.cmu.edu

***Abstract*—In a long-haul sensor network, sensors are remotely deployed over a large geographical area to perform certain tasks. We consider 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 satellite links. The severe loss and delay inherent over the satellite channels render insufficient the number of estimates successfully arriving at the fusion center, thereby limiting the potential fusion gain and resulting in suboptimal accuracy performance of the fused estimates. The system can adopt certain retransmission-based transport protocols so that lost messages can be recovered over time. However, excess delay may be incurred that can potentially violate the deadline for reporting the estimate. For many applications, though, retrodiction/smoothing techniques can be applied so that the chances of incurring such excess delay are greatly reduced. In this work, we analyze the extent to which retrodiction, along with message retransmission, can improve the performance of delay-sensitive state estimation tasks. Results of numerical and simulation studies of an illustrative example and a ballistic target tracking application are shown in the end to demonstrate the validity of our analysis.**

***Index Terms*—Long-haul sensor networks, state estimate fusion, message retransmission, prediction and retrodiction, mean-square-error (MSE) performance, reporting latency.**

I. Introduction

In a long-haul sensor network, sensors are deployed to cover a vast geographical area, which could be a continent or even the entire globe depending on the specific application. We consider a class of such networks in which state estimates (e.g., position and velocity) of certain dynamic targets – such as aircrafts or ballistic missiles [3] – are sent from the remote sensors to a fusion center so that a global estimate can be obtained by fusing the individual estimates. In some situations, satellite links might be the only type of cost-effective medium for such long-range communications because of the prohibitive cost of extending submarine and terrestrial fiber connections extensively to rough terrains and sparsely populated areas. This work in particular focuses on such satellite link-based monitoring and tracking applications.

The motivation for this work is the many challenges arising from the imperfect communications over the long-haul satellite links. Because of the long distance (tens of thousands of miles), the propagation time of signals is significant. For example, the round-trip time (RTT) for signal propagation with a geostationary earth orbit (GEO) satellite is more than a half second [10]. More importantly, communication over the satellite links is characterized by sporadic high bit-error rates (BERs) and burst losses. Losses either incurred during transmission or resulting from the high BERs could further effectively reduce the number of messages available at the fusion center. It is well known that fusion of estimates from different sensors is a viable means of reducing the estimation error; with high loss rates, however, only a portion of the potential fusion gain could be achieved and the quality of the fused estimate output obtained may be deemed unacceptable by the system operator. Apparently, all the above-mentioned drawbacks of the satellite links could work against the very purpose of the underlying task – to promptly and accurately report state estimates – and may result in failure to comply with the requirement on the worst-case estimation error.

State estimation under imperfect communications has been studied in the literature. State augmentation [11] can handle fixed delay up to several sampling periods. In [4], an upper bound of the packet loss rate is derived above which the estimation error becomes unbounded. A dynamic selective fusion method based on information gain is proposed in [9] so that fusion is deferred till enough information has arrived at the fusion center. One way to counteract the effect of the lossy transmission link is to adopt certain transport protocols in which message retransmission is implemented and some lost messages can be recovered after one or multiple rounds of retransmission. Not to be overlooked, however, is another aspect of the system requirement – the delay performance. Owing to often near real-time requirements of the monitoring/tracking tasks, the system often allows for only a small time gap between the time of interest and the time when the estimate should be finally obtained and reported. This often comes as a predefined reporting deadline before which an estimate must be reported by the fusion center. Message retransmission may exacerbate the reporting delay performance by incurring extra time on top of the already relatively large propagation and transmission latency. The fusion center may have to increase its reporting time significantly in order to recover the lost messages, even at the risk of violating the stipulated reporting deadline.

The transmission control protocol (TCP) implemented in wired Internet and wireless local area networks (WLANs) is still garnering research efforts that are too numerous to list. Analysis of TCP-like transport protocols over satellite links can be found in studies such as [1] and [5]. Commonly acknowledged are the difficulties in applying “conventional” TCP protocols to transmission over satellite links, mainly because of the very large propagation delay not encountered in other networks. The specificity of our application also somewhat distinguishes our analysis from the ones geared toward the voice- and video-based broadcasting and data-based Internet access, both of which have continuous data in flight. Also of note is that in our settings, state estimates from the remote sensors are generally intermittently sent over a wide-band satellite channel – with the interval possibly ranging from a few times within a second to once every few minutes – and thus congestion is not as much a concern as in conventional TCP applications. Hence, we assume a simplified transport protocol in which retransmission is performed on the message-level basis.

In many state estimation applications, retrodiction, also known as smoothing, serves as the "backward prediction" of an earlier estimate. Depending on the relationship between the length of data used and the time of interest, we can categorize retrodiction roughly into fixed-point, fixed-lag, and fixed-interval retrodiction [11]. Whereas the conventional retrodiction techniques are used mainly for improving estimates that have been obtained, we are primarily interested in how missing estimates can be interpolated from retrodiction[1] and how much the excess latency can be potentially reduced for recovering the lost messages.

In this work, we provide analytical models to systematically study the impact from retransmission and retrodiction on fusion performance under variable loss and delay conditions in a long-haul sensor network. In particular, we study two types of retrodiction mechanisms: non-cooperative and cooperative retrodiction. In the former case, sensors do not participate in retroactive estimation themselves and the fusion center has to extrapolate available information and perform retrodiction on its own; whereas in the latter scenario, the sensors send out their own retrodicted values upon request, which are to be utilized directly by the fusion center upon successful reception. To the best of our knowledge, this work is among the first to link both communication (message retransmission) and computation (prediction and retrodiction) with state estimation performance. Accounting for both estimation errors and reporting latency, we explore the effect of applying retransmission and retrodiction on system performance improvement and their limitations. Simulations of a numerical example and a coasting ballistic target tracking application are conducted and the results under various conditions are shown to validate our analysis.

The paper is organized as follows: After briefly introducing retransmission and retrodiction mechanisms in Sec. II, we provide analysis of the delivery rate of a message after retransmission (without retrodiction) in Sec. III. Then we focus on the effect of retrodiction on the estimation performance improvement in Secs. IV and V, considering both non-cooperative and coooperative types of retrodiction. Numerical and simulation results are presented in Sec. VI before we conclude the paper in Sec. VII.

II. Background and System Model

A. Estimate Fusion

Fig. 1 illustrates the architecture of estimate fusion. The measurement data are collected at the remote sensors and state estimates are individually generated. These estimates are then sent via long-haul links to the fusion center, which, upon successful reception of a subset of these estimates, applies a certain fusion rule and obtains the final estimate to be reported. The raw measurement data usually come in larger volumes and thus are not directly sent to the fusion center. Many types of filtering (generating the estimates from the noisy measurements) and fusion (fusing the estimates to generate a better one) algorithms for tracking applications have been studied in the literature. However, our focus in this work is not on performance comparison among different algorithms; rather, of interest to us is the performance improvement, for given filtering and fusion algorithms, from retransmission and retrodiction to be described below.

[1]In the meantime, an available estimate is retrodicted by subsequent estimate(s) as well whenever applicable as in conventional retrodiction.

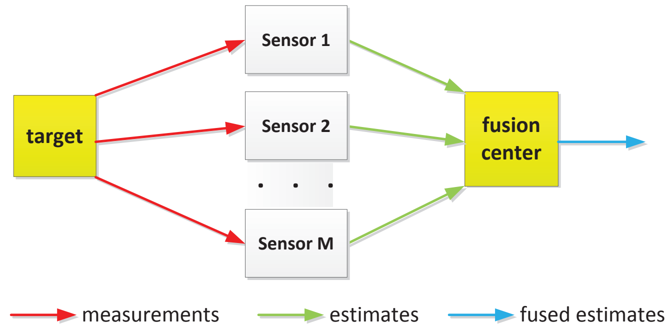

Fig. 1: Fusion of state estimates generated by a total of M sensors

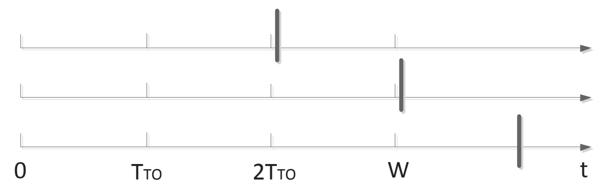

Fig. 2: Timing of message retransmission: the timeout is T_{TO}, retransmission window is W, and different choices of the cutoff time T_{CO} are marked by bold lines

B. Message Retransmission

In a long-haul sensor network, a remote sensor sends out a message containing the state estimate; upon successful reception of this message, the fusion center sends back an acknowledgment (ACK) message to the sensor. A failed arrival of the ACK message before the expiration of the timeout T_{TO} – due to loss and/or long delay of the message itself or the ACK – will prompt the sensor to retransmit the message. Typically, T_{TO} could be several times the RTT of the connection, and over long-haul connections it could be of the order of seconds. Setting T_{TO} too long could reduce the maximum number of retransmissions, thereby limiting the potential to recover the lost message; on the other hand, a short T_{TO} may incur many rounds of retransmission (often unnecessarily) when the sensor could have waited a bit longer to receive the ACK. Such retransmission continues till the acknowledgment is received by the sensor, or the retransmission window W expires. This window should ideally contain multiple T_{TO} periods so that under adverse link conditions, it's likely that the message can eventually be recovered after multiple tries.

In a real system, the reporting deadline may limit the potential gain from retransmission as the overall time before reporting

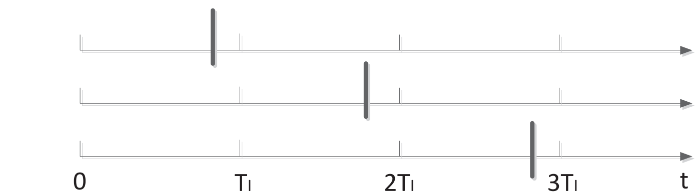

Fig. 3: Timing of estimate retrodiction: the estimation interval is T_I and different choices of the cutoff time T_{CO} are marked by bold lines

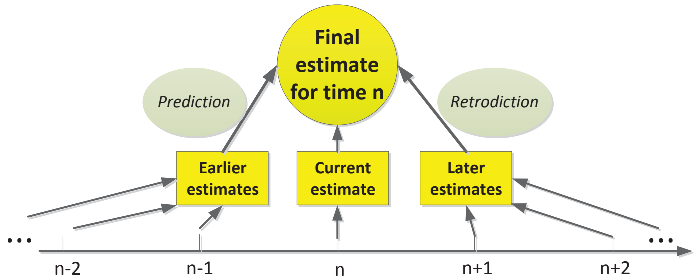

Fig. 4: Prediction and retrodiction: the estimate at time n is to be obtained

can be very short. A cutoff time T_{CO} is defined to mark the end of the waiting at the fusion center. This cutoff on the one hand limits the total number of retransmissions, and on the other limits certain messages from being eventually delivered due to the randomness of the delay. In Fig. 2, the effect of this time cutoff is shown. The window W is set to be $3T_{TO}$ and hence there are a total of two rounds of retransmision (at T_{TO} and $2T_{TO}$). In the first case, T_{CO} is small so that the last round of retransmitted message cannot arrive in time for fusion. While in the last case, setting T_{CO} way beyond the end of the retransmission window is not likely to significantly increase the chance of receiving the message. Therefore, the system should guarantee that the retransmission window – at the sensors side – is commensurate with the cutoff time – at the fusion center, as in the second case in the figure, so that the fusion center could benefit from all rounds of retransmission while not wasting time attempting to recover the pending message after the retransmission has ended.

C. Estimate Retrodiction

Estimation of a target state at a particular time based on measurements collected beyond that time is generally called retrodiction or smoothing. Retrodiction improves the accuracy of the estimates, thanks to the use of more information, at the cost of extra delay. Nevertheless, the inherent link delay in a long-haul network occurring before the final reporting entails that the fusion center can exploit the opportunities for potential retrodiction to improve the accuracy of the fused estimate. Moreover, the randomness of the arrival delay of different messages also facilitates the fusion center to opportunistically interpolate the missing messages from the available ones for subsequent time periods.

Consider a discrete-time system in which estimates are generated regularly at a certain fixed period T_I. We emphasize that retrodiction cannot be universally applied. The prerequisite is that an estimate at least cannot be reported before a retrodicted estimate can possibly be obtained. In other words, the retransmission window W and the cutoff time T_{CO} must contain at least one estimation interval T_I[2]. This is shown in Fig. 3. In the first case, no retrodiction can be possibly performed because the cutoff time comes before the end of the same estimation interval; while in the second and third case, a maximum of one and two, respectively, rounds of retrodiction can be possibly carried out. In many actual systems, the near real-time reporting requirement dictates that the estimation intervals are chosen fairly small. As a result, the next estimate can be generated and sent to the fusion center in time for retrodiction before the reporting deadline.

Let $\mathbf{x}_n$ denote the true target state at time $t = nT_I$ and $\hat{\mathbf{x}}_n$ its estimate. One note is that since we have both continuous time – when addressing latency – and the discrete time indices for retrodiction analysis, all time subsequently labeled as "n" – the time of interest to us – should be understood as the continuous time nT_I and the retransmission parameters T_{TO} and W and the cutoff time T_{CO} are all defined relative to this time instant.

Fig. 4 demonstrates the effect of prediction and retrodiction on the quality of the state estimate. For each sensor, for example, we have the following types of estimates when retrodiction of *up to one step* is performed:

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

In the cases a) and d), the estimate $\hat{\mathbf{x}}_n$ is received by the fusion center; whereas in both other cases, this estimate is missing and hence prediction of one or multiple steps is at first necessary. As is well-known in filtering theory, estimates derived from prediction alone generally have higher errors when system uncertainty exists; and the errors will increase with the number of prediction steps that have accrued. For example, $\hat{\mathbf{x}}_{n|n-2}$ is a worse estimate than $\hat{\mathbf{x}}_{n|n-1}$ in terms of accuracy. On the other hand, the presence of the subsequent estimate $\hat{\mathbf{x}}_{n+1}$ (in the last two cases) helps improve the quality of the estimate of time n. Of course, the improvement is on top of the predicted estimate in case c) but on the already received $\hat{\mathbf{x}}_n$ in case d). Of concern here is whether an interpolated estimate from retrodiction – such as that in case c) – can adequately substitute the default estimate in case a); and what is the probabilistic performance of obtaining these different types of estimates so that the system requirement on estimation errors can be met.

The quantitative performance of retrodiction also depends on the specific algorithm. In this work, we apply the Rauch-Tung-Striebel (RTS) retrodiction [11] because not only the algorithm is easy to implement, with relatively low computational cost, but the algorithm involves only the state estimates and their covariances – rather than the raw measurement data – which is well suited in our settings when the fusion center needs to run the algorithm. When the retrodiction task is also carried out by the sensors (as will be studied in Sec. V), they have greater latitude in selecting which retrodiction algorithm to use since they have measurements immediately available. Nevertheless, for fair comparison, we have the sensors apply the RTS retrodiction algorithm as well when needed.

[2] And also the initial latency of the latest estimate to arrive at the fusion center.

III. Message Delivery with Retransmission

The message-level loss and delay characteristics are determined by the long-haul link conditions. We assume that each message sent by a sensor is lost during transmission with probability p independently of other messages. Normally, the latency that a message experiences before arriving at the fusion center consists of the initial detection and measurement delay, data processing delay by both the sensor and the fusion center, propagation delay, and transmission delay, among others[3]. These are collectively considered as the minimum delay that a message must undergo to reach the fusion center, which is bound mostly by factors such as the distance of the satellite link, the transmission data rate, and length of the message. The extra random delay is often due to link conditions such as weather and terrain. We suppose a pdf $f(t)$ can model the overall delay t that a message experiences to be successfully delivered to the fusion center. One typical example is that of the shifted exponential distribution:

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

in which T serves as the common link and processing delay, and μ is the mean of the random delay beyond T. In a real system, the empirical values of the message delay can be measured over time and thus an approximate function $\tilde{f}$ can be estimated. In the following analysis, however, we still use the generic function $f(t)$ to model the arrival delay.

A. Message Delivery Rate

We are interested in the average probability of a message being successfully delivered by a certain time, that is, by the cutoff time T_{CO}. An estimate is only counted once even if it arrives multiple times due to retransmission. The duplicate messages received by the FC can simply be ignored as they will not contribute further to the fusion performance.

With the time of interest being regarded as time zero in this section, the maximum number of retransmissions before the cutoff time T_{CO} is

$$K_{retx} = \left\lceil \frac{\min\{T_{CO}, W\}}{T_{TO}} \right\rceil - 1. \tag{2}$$

From the definition, $K_{retx}+1$ is the total rounds of transmission, including the original and subsequent retransmissions.

We define $p^k_{del,t}$ as the probability that a message is delivered by time t after k rounds of retransmissions, and

$$T_{retx,k} = T_{CO} - kT_{TO}, \text{ for } k = 0, 1, ..., K_{retx} \tag{3}$$

as the duration of the period $[kT_{TO}, T_{CO}]$ in which the k-th retransmitted message is in flight and could be potentially delivered to the fusion center.

When there is no retransmission within $[0, t]$, the probability of a message being delivered by time t is

$$p^0_{del,t} = (1-p)F(t), \tag{4}$$

in which $F(t) = \int_0^t f(u)\, \mathrm{d}u$ is the cdf of the arrival delay. Its complement, the probability that the original message is unavailable at time t, is denoted as

$$p^0_{loss,t} = 1 - p^0_{del,t} = p + (1-p)\overline{F}(t), \tag{5}$$

in which $\overline{F}(t) = 1 - F(t)$ is the tail distribution. With these two probabilities, we can derive the message delivery rate $p^{K_{retx}}_{del,T_{CO}}$.

The original message is delivered by T_{CO} with probability

$$p^0_{del,T_{CO}} = p^0_{del,T_{retx,0}} = (1-p)F(T_{retx,0}). \tag{6}$$

And with the first round of retransmission, the delivery probability totals

$$p^1_{del,T_{CO}} = p^0_{del,T_{CO}} + p^0_{loss,T_{retx,0}} p^0_{del,T_{retx,1}}. \tag{7}$$

In general, for the k-th ($0 < k \leq K_{retx}$) round of message retransmission, we have

$$p^k_{del,T_{CO}} = p^{k-1}_{del,T_{CO}} + p^0_{del,T_{retx,k}} \left(\prod_{i=0}^{k-1} p^0_{loss,T_{retx,i}} \right). \tag{8}$$

In other words, the extra delivery rate from the k-th round is realized when all the previous $k-1$ retransmissions and the original message are not available by T_{CO}. Subsequently, we can obtain the overall message delivery probability within time $[0, T_{CO}]$ by summing up all such probabilities:

$$\begin{aligned} p^{K_{retx}}_{del,T_{CO}} &= \sum_{k=0}^{K_{retx}} p^0_{del,T_{retx,k}} \left(\prod_{i=0}^{k-1} p^0_{loss,T_{retx,i}} \right) \\ &= (1-p) \sum_{k=0}^{K_{retx}} F(T_{retx,k}) \left\{ \prod_{i=0}^{k-1} [1 - (1-p)F(T_{retx,i})] \right\}. \end{aligned} \tag{9}$$

B. MSE Performance

The estimation mean-square-error (MSE) performance can be linked to the message delivery probability. As a general rule of thumb, as an unbiased estimate becomes more likely to be delivered to the fusion center, the quality of the final estimate benefits in terms of improved accuracy. To illustrate the effect, we consider a numerical example, in which the scalar dynamic state evolves as follows:

$$x_n = -0.95 x_{n-1} + w_n, \tag{10}$$

where w_n is zero-mean white Gaussian process noise with variance 1. The system is observed by a sensor of the form

$$\hat{x}_n = x_n + v_n, \tag{11}$$

where the i.i.d. estimation error term v_n also satisfies $v_n \sim \mathcal{N}(0, 1)$ and is independent of w_n.

The MSE is simply the square of the estimation error for this scalar system, which is $MSE = 1$ when all the messages are delivered. Let $\hat{x}^F$ denote the final state estimate at the fusion center. Now suppose in a lossy environment, the fusion center uses the estimate provided by the sensor $\hat{x}^F_n = \hat{x}_n$, if the estimate is successfully received; and its own predicted state from the previous estimate $\hat{x}^F_n = -0.95\hat{x}^F_{n-1}$ otherwise. As the system imposes its maximum tolerable MSE as $MSE_{max} = 1.5$, there exists an associated minimum message delivery rate $p_{del,min}$ – which is found to be 0.63 from simulations – below which the MSE requirement will be violated. Consequently, the effect

[3]The queueing delay is minimal for the same reason with little/no congestion.

of network loss and delay characteristics and the retransmission parameters (including timeout T_{TO}, window W, and cutoff T_{CO}) should guarantee that the delivery rate in Eq. (9) is at least $p_{del,min}$.

TABLE I: MSE performance with different TCP timeout and final cutoff: loss $p = 0.5$, delay $t \sim U(0.5, 2.5)$

cases	T_{TO}	W	T_{CO}	No. retx	del prob	satisfy MSE?
1	1.5	3	3	1	.625	N
2	1.2	3	3	2	.671	Y
3	1.5	3	3.5	1	.688	Y

In Table I, we show how parameters can be tuned to meet the MSE requirement. The message-level loss rate is measured to be $p = 0.5$, and the arrival delay falls uniformly in the interval [0.5,2.5]. Case 1 – considered as the default setting – cannot satisfy the maximum tolerable MSE set at 1.5. From Eq. (9), the actual delivery rate is 0.625, just shy of the minimum delivery rate 0.63. To improve the performance, the system can either schedule more frequent retransmission (case 2) or delay its final reporting if allowed (case 3). In case 2, the timeout is set as 1.2 instead of 1.5; subsequently, a new round of retransmission can be scheduled before the cutoff time, thereby increasing the message delivery rate to 0.671, now above the minimum requirement. Alternatively, in case 3, with the same retransmission schedule, the fusion center simply waits an extra half second, and the resulting delivery rate is even better, at 0.688.

IV. State Estimates with Retransmission and Non-Cooperative Retrodiction

From the last section, retransmission can effectively increase the message delivery rate, which in turn improves the estimation MSE performance. However, at times, we wish to expedite this recovery process so that the final estimate can be reported promptly; besides, the system may impose rather stringent requirements on the estimation MSE so that given the same allocated time for retransmission, we want more accurate estimates from the output of the fusion center. This section, along with Sec. V, addresses these concerns by means of utilizing estimate retrodiction.

Based on the criteria that whether the sensors actively participate in retrodiction during message retransmission, we categorize our schemes into non-cooperative and cooperative retrodiction. In the former case, message retransmission is carried out in exactly the same way as before; it is up to the fusion center to opportunistically apply retrodiction whenever applicable. In contrast, cooperative retrodiction means that the sensors themselves, upon request, send out the retrodicted estimates during retransmission so that the fusion center can directly fuse such retrodicted values if successfully delivered. In what follows, we derive the delivery rates of different types of estimates during retransmission for both types of retrodiction and consider their impact on the final MSE performance. The remainder of this section focuses on the non-cooperative retrodiction.

A. Analysis of One-Step Non-Cooperative Retrodiction

We aim to derive the delivery probabilities of different types of estimates with retrodiction of up to one step being performed and the resulting MSE performance. Analysis for retrodiction of two or more steps can be similarly obtained, albeit in a more exhaustive manner, as the number of possible scenarios grows exponentially with the number of steps[4].

Using similar notations as those introduced in Sec. II, we consider the following probabilities at the cutoff time $nT_I + T_{CO}$:

- p_{n^-}, the probability that $\hat{\mathbf{x}}_{n^-}$ is reported (neither $\hat{\mathbf{x}}_n$ or $\hat{\mathbf{x}}_{n+1}$ is delivered by the cutoff);
- $p_{n|n}$, the probability that $\hat{\mathbf{x}}_{n|n}$ is reported ($\hat{\mathbf{x}}_{n+1}$ is not delivered yet);
- $p_{n^-|n+1}$, the probability that $\hat{\mathbf{x}}_{n^-|n+1}$ is reported ($\hat{\mathbf{x}}_{n+1}$ has been delivered but not $\hat{\mathbf{x}}_n$);
- $p_{n^+|n+1}$, the probability that $\hat{\mathbf{x}}_{n^+|n+1}$ is reported (both $\hat{\mathbf{x}}_n$ and $\hat{\mathbf{x}}_{n+1}$ have been delivered).

The analysis in the last section can be readily applied here, thanks to the independence of the transmission from different time intervals. Similar to Eq. (2), we have the maximum number of retransmissions during $t \in [nT_I + T_I, nT_I + T_{CO}]$ given as

$$K_{retx,retr_1} = \left\lceil \frac{\min\{T_{CO} - T_I, W\}}{T_{TO}} \right\rceil - 1, \tag{12}$$

in which the subscript $retr_1$ denotes that there is a maximum of one-step retrodiction. And the duration in Eq. (3) can also be defined likewise for the above time interval:

$$T_{retx,retr_1,k} = T_{CO} - T_I - kT_{TO}, \text{ for } k = 0, 1, ..., K_{retx,retr_1}. \tag{13}$$

To calculate the above probabilities, we need to consider the probability that $\hat{\mathbf{x}}_{n+1}$ is delivered at the cutoff time. This probability, denoted as $p_{del,T_{CO}-T_I}^{K_{retx,retr_1}}$, follows the very same form as in Eq. (9), but with newly defined Eqs. (12) and (13) substituting the corresponding terms. Then we have

$$p_{n^-} = (1 - p_{del,T_{CO}}^{K_{retx}})(1 - p_{del,T_{CO}-T_I}^{K_{retx,retr_1}}) \tag{14}$$

$$p_{n|n} = p_{del,T_{CO}}^{K_{retx}}(1 - p_{del,T_{CO}-T_I}^{K_{retx,retr_1}}) \tag{15}$$

$$p_{n^-|n+1} = (1 - p_{del,T_{CO}}^{K_{retx}})p_{del,T_{CO}-T_I}^{K_{retx,retr_1}} \tag{16}$$

$$p_{n^+|n+1} = p_{del,T_{CO}}^{K_{retx}}p_{del,T_{CO}-T_I}^{K_{retx,retr_1}} \tag{17}$$

It's generally difficult to derive the MSE analytically. Note that in Eqs. (14) and (16), the time at which the last received estimate was generated is not specified; that is, the number of prediction steps leading up to $\hat{\mathbf{x}}_{n^-}$ is unknown. Even with known MSEs for $\hat{\mathbf{x}}_{n|n}$ and $\hat{\mathbf{x}}_{n^+|n+1}$, an exact evaluation of MSEs for $\hat{\mathbf{x}}_{n^-}$ and $\hat{\mathbf{x}}_{n^-|n+1}$ requires knowledge of the MSEs for *any* number of prediction steps, which is of course unrealistic. However, approximation from finite-step predicted values can be used to somewhat reflect the actual MSE performance. Numerical results will be provided in Sec. VI.

V. State Estimates with Retransmission and Cooperative Retrodiction

In this section, we provide similar analysis as above – that is, with a maximum of one step of retrodiction being performed

[4]Another caveat is that message-level loss and delay may worsen as significantly more data are sent simultaneously with increasing retrodiction steps.

– and focus on comparisons between cooperative and non-cooperative retrodiction techniques. In particular, two possible implementations of the cooperative retrodiction are studied, one in which we only consider one-way communications – as we have done so far – and the other requiring two-way analysis that addresses the delivery of the ACK messages as well.

A. Condition for One-Step Cooperative Retrodiction

In non-cooperative retrodiction, message retransmission is scheduled in the manner as described in Sec. II and a sensor is oblivious to the retrodiction process happening at the fusion center. In contrast, cooperative retrodiction requires the retrodicted estimates to be sent out during retransmission. In order for a sensor to actually send out its one-step retrodicted estimates, the retransmission window W should not have expired at the end of the estimation interval T_I; in fact, there should be at least one round of retransmission initiated by the sensor after T_I when the retrodicted estimate can be sent out by the sensor. Hence, compared to non-cooperative retrodiction, tighter conditions are in place for cooperative retrodiction.

B. Cooperative Retrodiction: One-way Communications without ACK

During the time period $[nT_I + T_I, nT_I + T_{CO}]$, instead of the original estimate, the sensor sends out the retrodicted estimate $\hat{\mathbf{x}}_{n^+|n+1}$ directly. The total number of retransmission rounds for the original message during $[nT_I, nT_I + T_I]$ is reduced to

$$K_{retx,coop} = \left\lceil \frac{T_I}{T_{TO}} \right\rceil - 1, \qquad (18)$$

while both the new state estimate $\hat{\mathbf{x}}_{n+1|n+1}$ and retrodicted estimate $\hat{\mathbf{x}}_{n^+|n+1}$ are sent after T_I. If $T_I = lT_{TO}$, $l = 1, 2, 3, ...$, both estimates will undergo

$$K_{retx,coop,retr_1} = K_{retx,retr_1} = \left\lceil \frac{T_{CO} - T_I}{T_{TO}} \right\rceil - 1, \qquad (19)$$

rounds of retransmission, in which the subscripts "$coop, retr_1$" and "$retr_1$" denote cooperative and non-cooperative retrodiction of up to one step respectively. Similar to our earlier analysis, we can obtain the delivery probabilities of the original estimate $\hat{\mathbf{x}}_{n|n}$ as $p_{del,T_{CO}}^{K_{retx,coop}}$, of the subsequent estimate $\hat{\mathbf{x}}_{n+1|n+1}$ as $p_{del,T_{CO}-T_I}^{K_{retx,retr_1}}$, and of the retrodicted estimate by the sensor $\hat{\mathbf{x}}_{n^+|n+1}$ as $p_{del,T_{CO}-T_I}^{K_{retx,coop,retr_1}}$. With an increased size of the state space, the probabilities of obtaining different types of estimates at the cutoff time can now be computed as

$$p_{n^-} = (1 - p_{del,T_{CO}-T_I}^{K_{retx,coop,retr_1}})(1 - p_{del,T_{CO}}^{K_{retx,coop}})(1 - p_{del,T_{CO}-T_I}^{K_{retx,retr_1}}) \qquad (20)$$

$$p_{n|n} = (1 - p_{del,T_{CO}-T_I}^{K_{retx,coop,retr_1}}) p_{del,T_{CO}}^{K_{retx,coop}} (1 - p_{del,T_{CO}-T_I}^{K_{retx,retr_1}}) \qquad (21)$$

$$p_{n^-|n+1} = (1 - p_{del,T_{CO}-T_I}^{K_{retx,coop,retr_1}})(1 - p_{del,T_{CO}}^{K_{retx,coop}}) p_{del,T_{CO}-T_I}^{K_{retx,retr_1}} \qquad (22)$$

$$p_{n^+|n+1} = p_{del,T_{CO}-T_I}^{K_{retx,coop,retr_1}} + (1 - p_{del,T_{CO}-T_I}^{K_{retx,coop,retr_1}}) p_{del,T_{CO}}^{K_{retx,coop}} p_{del,T_{CO}-T_I}^{K_{retx,retr_1}} \qquad (23)$$

Note in Eq. (23) that the estimate $\hat{\mathbf{x}}_{n^+|n+1}$ can be obtained either directly from the sensor, or indirectly in the manner we discussed in the non-cooperative retrodiction case.

The above analysis is along the same line as that in the last section, where only one-way communication is considered. This can also been seen as the extreme case where no ACK is ever sent back by the fusion center, since the sensors always have their one-step retrodicted estimates sent out after one estimation interval T_I. In reality, though, the ACK might have been successfully received by the sensor within T_I, thereby obviating the need for further retransmission. Next, we carry out two-way communication analysis to account for such scenarios.

C. Cooperative Retrodiction: Two-way Communications with ACK

For satellite systems with the conventional bent-pipe type of transponders [10], one uplink (sensor $\rightarrow$ satellite) and downlink (satellite $\rightarrow$ FC) pair is used for the forward link, and the reverse link similarly consists of the uplink (FC $\rightarrow$ satellite) and downlink (satellite $\rightarrow$ sensor) pair. Depending on specific channel allocation schemes (e.g., TDMA- or FDMA-based), that is, whether the forward and reverse channels are assigned the same frequency band, the delay distribution of the ACK could vary from that of the messages[5]. Regardless, we have the pdf of the sum of two random delay values being expressed as the convolution of their respective pdfs:

$$h(t) = f(t) \star g(t) = \int_{t=0}^{\infty} f(u) g(t-u) \, \mathrm{d}u, \qquad (24)$$

in which f and g are the distributions of the forward and reverse links, respectively. Meanwhile, if the ACK message is lost over the reverse link with a probability p_{ACK}, the overall probability that the ACK message can be eventually delivered is $(1-p)(1-p_{ACK})$, and its complement

$$p_T = 1 - (1-p)(1-p_{ACK}) \qquad (25)$$

is the loss rate of the "super-message" that includes both the estimate message and ACK. With this loss rate and $h(t)$ function, we have the probability that the ACK is delivered by time t and hence no more retransmission occurs afterward:

$$\overline{p_{retx}}(t) = (1 - p_T) H(t), \text{ for } t \in [0, \min\{T_{CO}, W\}], \qquad (26)$$

in which $H(t) = \int_0^t h(u) f(u) \, \mathrm{d}u$ is the cdf of the two-way communications delay.

After the ACK has been successfully received within one estimation interval, the retrodicted estimate is no longer to be sent out, thereby reducing the chance that the best estimate $\hat{\mathbf{x}}_{n^+|n+1}$ is available at the fusion center. Subsequently, with probability $1 - \overline{p_{retx}}(T_I)$, Eqs. (20)–(23) hold true; on the other hand, with probability $\overline{p_{retx}}(T_I)$, only two types of estimates are possible to be used by the cutoff time – since $\hat{\mathbf{x}}_{n|n}$ has been received successfully – with probabilities $1 - p_{del,T_{CO}-T_I}^{K_{retx,retr_1}}$ for $\hat{\mathbf{x}}_{n|n}$ and $p_{del,T_{CO}-T_I}^{K_{retx,retr_1}}$ for $\hat{\mathbf{x}}_{n^+|n+1}$. Using the law of total probability, we can easily incorporate them to calculate the overall probabilities of obtaining each type of estimate.

[5] Also the initial delay could be quite different too, owing to the usually much smaller size of the ACK messages.

TABLE II: Probabilities of using different types of estimates with default link statistics

estimate	$\hat{x}_{n^-}$	$\hat{x}_n$	$\hat{x}_{n^-\|n+1}$	$\hat{x}_{n^+\|n+1}$
no retx	.500	.500	-	-
retx, no retr	.259	.741	-	-
non-coop retr	.134	.384	.125	.357
coop retr (w/o ACK)	.134	.134	.125	.607
coop retr (w/ ACK)	.080	.288	.075	.557

VI. Performance Studies of Estimate Fusion with Retransmission and Retrodiction

In this section, we first revisit the numerical example introduced in Sec. III and explore the effects of estimate fusion, message retransmission, and state retrodiction on estimation performance. Both the one-sensor and two-sensor scenarios are considered, the latter of which can be generalized to multi-sensor fusion. Then, we simulate the tracking of a coasting ballistic target and demonstrate the benefits and limitations of applying retransmission and retrodiction.

A. Communication Link Statistics

The following link statistics are used in our simulations. The default forward link loss rate is $p = 0.5$, compared to that of the reverse link $p_{ACK} = 0.1$. The ACK message is often much shorter than the estimate and thus is less likely to be lost. The arrival delay of both directions satisfies the shifted exponential distribution defined in Eq. (1), with $\mu_F = 0.3$ s and $\mu_R = 0.2$ s for the forward and reverse links respectively and common initial latency $T = 0.5$ s. The default $T_{TO} = T_I$ and $W = T_{CO}$ are set to be 1.5 s and 3 s respectively in the numerical example, both multiples of the measured RTT at 0.75 s; whereas in the ballistic target tracking simulations, $W = T_{CO} = 5$ s and $T_{TO} = T_I$ is set as 3 s.

B. Performance of The Numerical Example

Again, the system state evolves according to Eq. (10). With the statistics from the last subsection, we list the probabilities, as derived in Secs. IV and V, of using different types of estimates at the time cutoff, in Table II. Comparing the first two rows, we observe that message retransmission can effectively increase the delivery rate of the original message $\hat{x}_n$ as expected. Comparing the last three rows with the second, we observe that although the probability of using the original estimate $\hat{x}_n$ becomes smaller when retrodiction is applied, the fusion center has a good chance to obtain the estimate with the smallest possible error – $\hat{x}_{n^+|n+1}$ in this case as shown in the last column – and result in further reduction of the estimation error.

1) Estimation MSE with One or Two Sensors: Consider the following two estimates

$$\hat{x}^1_n = x_n + v^1_n \tag{27}$$

$$\hat{x}^2_n = x_n + v^2_n \tag{28}$$

generated by Sensor 1 and 2 respectively, where $v^1_n \sim \mathcal{N}(0, 1)$ and $v^2_n \sim \mathcal{N}(0, 2)$, independent of each other, and both are independent of the process noise w_n. The two sensors are heterogeneous as they have uneven estimation error performance.

TABLE III: Estimation error variances from different pairs of sensor estimates

		$\hat{x}^1_{n\|n-1}$	$\hat{x}^1_n$	$\hat{x}^1_{n^-\|n\pm1}$	$\hat{x}^1_{n^+\|n+1}$
	EV	1.903	1	1.143	0.775
$\hat{x}^2_{n\|n-1}$	2.805	1.602	0.737	0.812	0.607
$\hat{x}^2_n$	2	0.975	0.667	0.727	0.559
$\hat{x}^2_{n^-\|n\pm1}$	1.933	0.959	0.659	0.990	0.553
$\hat{x}^2_{n^+\|n+1}$	1.631	0.878	0.620	0.672	0.663

EV: error variance

If both sensors send their state estimates to the fusion center, the following fusion rule is applied:

$$P^F = P^1 - \frac{P^1 - P^{12}}{P^1 + P^2 - 2P^{12}}, \tag{29}$$

$$\hat{x}^F = \hat{x}^1 + \frac{P^1 - P^{12}}{P^1 + P^2 - 2P^{12}}(\hat{x}^2 - \hat{x}^1), \tag{30}$$

which is the scalar form of the track-to-track fusion (T2TF) [2]. In these equations, $\hat{x}^1$ and $\hat{x}^2$ are respectively the estimates for Sensor 1 and 2 eventually served as the inputs to the fusion algorithm; P^1 and P^2 are the associated error variances. Because of the common process noise w_n, the correlation between the two estimates cannot be ignored, whose effect is reflected in the cross-covariance term P^{12}. Calculation of P^{12} is not always easy, and we have obtained the results from trial-and-error for different pairs of estimates from both sensors.

In Table III, the error variances of different types of estimates are listed in the second row for Sensor 1 and the second column for Sensor 2. The symbol "$\hat{x}_{n^-|n\pm1}$" denotes the estimate is obtained with both $\hat{x}_{n+1}$ and $\hat{x}_{n-1}$ available, but not $\hat{x}_n$. We note that although the accuracy performance of Sensor 2 is generally worse than that of Sensor 1, retrodiction can generate a better estimate than the original when the latter is not available; that is, $\hat{x}_{n^-|n\pm1}$ has a smaller error variance than the original $\hat{x}_n$ for Sensor 2, but not the case for Sensor 1. The remaining entries in the table are the error variances of the fused state estimate for different pairs of estimates of both sensors. Some interesting results can be observed from the table. For example, the best estimate is not obtained when both the retrodicted estimates are used, but rather $\hat{x}_{n^+|n+1}$ from Sensor 1 and $\hat{x}_{n^-|n\pm1}$ from Sensor 2. In fact, the last column shows that with $\hat{x}_{n^+|n+1}$ from Sensor 1 available, it is a better choice to use the one-step predicted estimate from Sensor 2 (which has a higher error variance) instead of the retrodicted $\hat{x}^2_{n^+|n+1}$.

Figs. 5 and 6 show the estimation MSEs of various schemes when the fusion center uses only Sensor 1 or 2, respectively. As can be observed in both figures, the performance difference before versus after using retransmission and retrodiction becomes more significant as the message-level loss rate increases. Even with moderate loss rates, for both sensors, message retransmission can effectively reduce the MSEs by more than 20% over one-time transmission. When retrodiction is also applied, the MSEs are reduced by an overall of more than 50% with Sensor 1 versus that of over 40% with Sensor 2. Further performance improvement from non-cooperative to cooperative retrodiction, both with or without ACK, though discernible from the plots, is not as significant.

In Fig. 7, the MSEs of the fused estimate for different schemes under variable loss rates are plotted. Comparing with

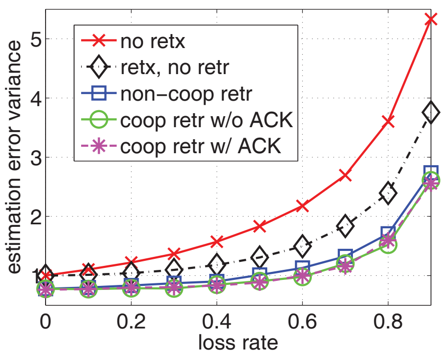

Fig. 5: Estimation MSE using Sensor 1 with different loss rates

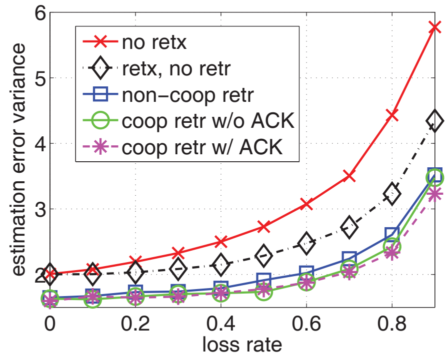

Fig. 6: Estimation MSE using Sensor 2 with different loss rates

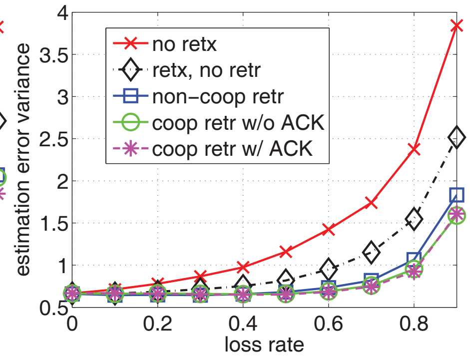

Fig. 7: Estimation MSE using both sensors with different loss rates

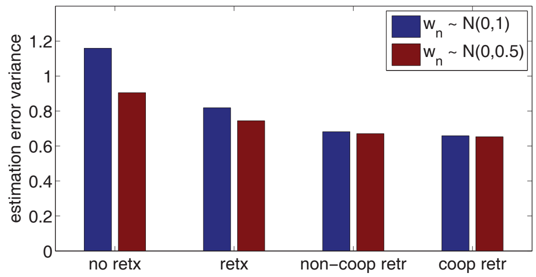

Fig. 8: Estimation MSEs under different process noise variances

the two figures to the left, we can clearly see the improved accuracy performance after fusion for every case. In addition, the distinction of the error reduction performance between low- and high-loss scenarios becomes more obvious. All schemes share nearly identical MSEs when the loss rate is low; however, the percentage of error reduction after applying retransmission and retrodiction is even higher, under moderate to severe loss rates, than that with only either sensor, at more than 60%.

2) Process Noise: In Eq. (10), the process noise has a variance of 1. We repeat the above studies for a system whose process noise follows $w_n \sim \mathcal{N}(0, 0.5)$. The comparisons are shown in Fig. 8 for the two-sensor fused estimates with the loss rate 0.5 (the cooperative retrodiction is shown here as one case as the performance difference with or without ACK's is small). The differences in the MSEs among the schemes are smaller when the process noise variance is reduced by a half. Generally speaking, when the system evolves with less uncertainty – corresponding to a smaller noise variance – the effect of loss is not as significant as the system becomes more "predictable". In the same light, retrodiction is not likely to bring as much improvement since the newest measurement does not provide significant information to reduce the estimation error.

3) Cutoff Time: As the default parameters, the cutoff time T_{CO} has been set to be 3 s, twice the duration of both the estimation interval T_I and retransmission timeout T_{TO}. The effect of reducing the cutoff time on the fused estimate is shown in Fig. 9. For one-time transmission without retrodiction, the error hardly changes because a message, if not lost, would have been delivered within 2 s with a probability of over 99%. However, since the loss rate is 0.5, retransmission can recover, with increasing probabilities when the cutoff time is increased, the message lost in the first round of transmission. Also the chances of receiving the subsequent and/or the retrodicted estimate go up with longer cutoff deadlines. Such effect is most noticeable as the cutoff increases from 2 s to 2.4 s during which the new round of (re)transmission is mostly likely to arrive.

C. Performance of Tracking of a Coasting Ballistic Target

We implemented estimation of a coasting ballistic target whose motion is governed mostly by gravity. By considering an example with minimal process noise, we demonstrate that although the use of retrodiction is fairly limited with perfect communications, retransmission along with retrodiction can still provide system improvements in a lossy transmission medium.

1) Target Model: The states of a coasting ballistic target are generated using the following state-space model [8]:

$$\dot{\mathbf{x}} \triangleq \begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{v}} \end{bmatrix} = \mathbf{f}\left(\begin{bmatrix} \mathbf{p} \\ \mathbf{v} \end{bmatrix}\right) \triangleq \begin{bmatrix} \mathbf{v} \\ \mathbf{a}_G(\mathbf{p}) \end{bmatrix}. \tag{31}$$

The target state vector $\mathbf{x} = \begin{bmatrix} \mathbf{p}^T & \mathbf{v}^T \end{bmatrix}^T$, where $\mathbf{p} = \begin{bmatrix} x & y & z \end{bmatrix}^T$ and $\mathbf{v} \triangleq \dot{\mathbf{p}} = \begin{bmatrix} \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T$ are the target position and velocity vectors, respectively. $\mathbf{a}_G(\mathbf{p})$ is the gravitational acceleration under the spherical Earth model [8]:

$$\mathbf{a}_G(\mathbf{p}) = -\frac{\mu}{p^2}\mathbf{u}_p = -\frac{\mu}{p^3}\mathbf{p}, \tag{32}$$

where $\mathbf{p}$ is the vector from the Earth's center to the target, $p \triangleq \|\mathbf{p}\|$ is its length, $\mathbf{u}_p \triangleq \mathbf{p}/p$ is the unit vector in the direction of $\mathbf{p}$, and $\mu = 3.986012 \times 10^5$ km^3/s^2 is the Earth's gravitational constant. The algorithm for state propagation can be found in [12]. The initial target state is [6]: $[113.75 \quad 3950 \quad 5150 \quad 0.94 \quad 3.33 \quad -6.0125]^T$, in which the position and velocity values are in the units of km and km/s respectively.

2) Sensor Profiles: A total of $M = 3$ sensors are deployed for reporting their state estimates. The measurements ($\mathbf{z}$) of the range (r), elevation (E), and azimuth (A) of the target are generated using the following measurement model [6]:

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{v}, \tag{33}$$

where the target state $\mathbf{x}$ is in Cartesian coordinates, but the measurement $\mathbf{z}$ and additive noise $\mathbf{v}$ are in the sensor spherical coordinates. If $[x \quad y \quad z]^T$ is the true position of the target, then

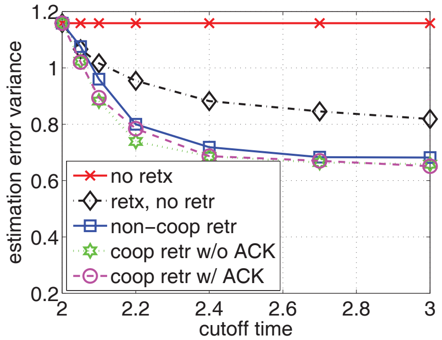

Fig. 9: Estimation MSE of the fused estimate versus varying cutoff time T_{CO} (in seconds)

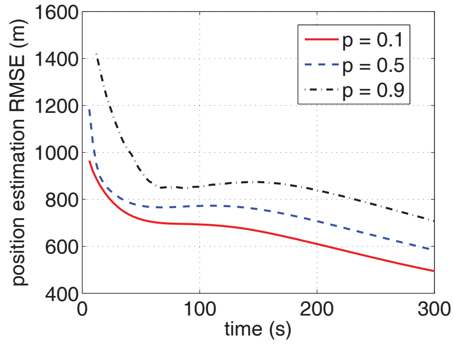

Fig. 10: RMSE of the fused position estimate over time with different loss rates

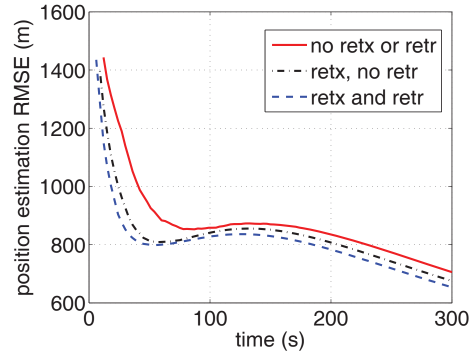

Fig. 11: RMSE of the fused position estimate over time with different schemes: loss rate = 0.9

the measurement is given as

$$\mathbf{z} = \begin{bmatrix} r \\ E \\ A \end{bmatrix} + \mathbf{v} = \begin{bmatrix} \sqrt{x^2+y^2+z^2} \\ \tan^{-1}\left(\frac{z}{\sqrt{x^2+y^2}}\right) \\ \tan^{-1}\left(\frac{x}{y}\right) \end{bmatrix} + \mathbf{v}, \quad (34)$$

$$\mathbf{v} \sim \mathcal{N}(0, \mathbf{R}), \quad \mathbf{R} = \begin{bmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_E^2 & 0 \\ 0 & 0 & \frac{\sigma_E^2}{\cos^2(E)} \end{bmatrix}. \quad (35)$$

The sensors apply the recursive best linear unbiased estimator (BLUE) proposed in [13] improving upon the measurement-conversion approach [7]; that is, the output has the minimum MSE among all linear unbiased filters in Cartesian coordinates.

3) Fusion Rule: We apply the linear fuser defined as follows:

$$\mathbf{P}^F = \left(\sum_{i=1}^{M} (\mathbf{P}^i)^{-1}\right)^{-1}, \text{ and } \hat{\mathbf{x}}^F = \mathbf{P}^F \sum_{i=1}^{M} (\mathbf{P}^i)^{-1} \hat{\mathbf{x}}^i, \quad (36)$$

where $\hat{\mathbf{x}}^F$ is the fused estimate and $\mathbf{P}^F$ is its error covariance matrix. $\mathbf{P}^i$ and $\hat{\mathbf{x}}^i$ are similarly defined for the estimates used by the fusion center corresponding to Sensor $i \in \{1, 2, 3\}$. The cross-covariances do not appear in the fuser because the process noise is zero in this example; that is, the trajectory of the ballistic target is deterministic.

4) Position Root-Mean-Square-Error (RMSE) Performance: Because of the zero process noise, the effect of prediction and retrodiction on estimation errors is minimal. Our test results indicate that under perfect communications, no discernible difference exists among the actual, predicted, or retrodicted estimates (one-step for both of latter cases). With lossy transmission links, however, the accuracy performance is degraded by various degrees compared to the full-communication case. In Fig. 10, the RMSE performance of the position estimate during the first 5 minutes of the tracking task is plotted for various loss rates. The estimation errors become noticeably higher as the lost rate is increased. The performance difference in the first 30 seconds is especially striking since the average waiting time for the fusion center to receive the initial estimates of all the sensors is significantly longer with higher loss rates.

To reduce the estimation error, we introduce retransmission and retrodiction, and the results are plotted in Fig. 11 for a severe loss rate 0.9. As the initial messages are more likely to be delivered with retransmission, the chances of applying retrodiction are also increased; the combined effects are two-fold: first, the fusion center can report its first estimate earlier; second, the estimation error can be reduced by 20-30% in the first half minute when both techniques are applied. These can be significant improvements in time-critical tracking tasks.

VII. Conclusion

In this paper, focusing on the state estimate fusion in lossy long-haul sensor networks, we analyzed the probabilities of obtaining different types of estimates by the fusion center when retransmission and retrodiction techniques are applied. Simulation results of a numerical example and one coasting ballistic target tracking example demonstrate the effectiveness of the retransmission and retrodiction mechanisms and the extent to which they can be applied so that the system requirements on estimation errors can be satisfied.

References

[1] M. Allman, C. Hayes, H. Kruse, and S. Ostermann. Tcp performance over satellite links. In *Proceedings of Fifth International Conference on Telecommunications Systems*, Nashville, TN, Mar. 1997.

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

[3] W. Boord and J. B. Hoffman. *Air and Missile Defense Systems Engineering*. CRC Press, 2012.

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

[5] T. R. Henderson and R. H. Katz. Transport protocols for internet-compatible satellite networks. *IEEE Journal on Selected Areas in Communications*, 17:326–344, 1999.

[6] T. H. Kerr. Streamlining measurement iteration for ekf target tracking. *IEEE Transactions on Aerospace and Electronic Systems,*, 27(2):408–421, Mar. 1991.

[7] X. R. Li and V. P. Jilkov. A survey of maneuvering target tracking, part iii: Measurement models. In *Proceedings of the 2001 SPIE Conference on Signal and Data Processing of Small Targets*, volume 4473, pages 423–446, San Diego, CA, Jul–Aug. 2001.

[8] X. R. Li and V. P. Jilkov. Survey of maneuvering target tracking. part ii: Motion models of ballistic and space targets. *Aerospace and Electronic Systems, IEEE Transactions on*, 46(1):96–119, Jan. 2010.

[9] 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.

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

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

[12] M. Yeddanapudi, Y. Bar-Shalom, K. R. Pattipati, and S. Deb. Ballistic missile track initiation from satellite observations. *IEEE Transactions on Aerospace and Electronic Systems*, 31(3):1054–1071, Jul. 1995.

[13] Z. Zhao, X. R. Li, and V. P. Jilkov. Best linear unbiased filtering with nonlinear measurements for target tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 40(4):1324–1336, Oct. 2004.