RESEARCH ARTICLE

State estimator for multisensor systems with irregular sampling and time-varying delays

I. Peñarrocha, R. Sanchis, J.A. Romero

ipenarro@esid.uji.es, rsanchis@esid.uji.es, romeroj@esid.uji.es

Departament d'Enginyeria de Sistemes Industrials i Disseny,

Universitat Jaume I,

Campus de Riu Sec,

12071 Castelló, Spain.

This paper addresses the state estimation in linear time-varying systems with several sensors with different availability, randomly sampled in time, and whose measurements have a time-varying delay. The approach is based on a modification of the Kalman filter with the negative-time measurement update strategy, avoiding running back the full standard Kalman filter, the use of full augmented order models or the use of reorganization techniques, leading to a lower implementation cost algorithm. The update equations are run every time a new measurement is available, independently of the time when it was taken. The approach is useful for networked control systems, systems with long delays and scarce measurements, and for out-of-sequence measurements.

Keywords: State estimation; scarce measurements; time varying delays

1. Introduction

In many industrial applications the control signal is updated at a fixed rate T, but the output cannot be measured at that control period due to the use of slow measuring procedures such as image processing or chemical or biochemical analysis, that limit the sampling rate and introduce a (possibly varying) delay. Sometimes, the output variable is measured by various sensors, each one having a different sampling rate, time delay and reliability. The use of communication networks between sensors and controllers may also introduce time varying delays in the received measurements and random data dropouts. As a practical example, consider an industrial ceramic atomizer whose purpose is to produce ceramic powder with a predefined humidity content. The humidity can be measured by infrared online sensors, but with a high noise and error. As a consequence, the humidity is also measured precisely with an analytical procedure that is performed scarcely in time by a human operator. The result is that there are two sets of available measurements: a noisy one obtained at regular intervals with no delay, and another one, more precise, obtained scarcely and irregularly on time with a varying delay (the time the operator spends in the analytical procedure). A similar problem occurs in biological or chemical reactors, where some concentrations can only be measured by analytical procedures that imply scarce and delayed measurements availability.

It is well known that the Kalman filter Kalman (1960) is the tool that minimizes the state prediction error variance when dealing with systems affected by gaussian disturbances and noises.

Some modifications of the original Kalman filter have been developed in order to take into account the existence of several measurable signals (states or outputs) that could be measured with redundant sensors of different precision, sampling rates and delay, leading to a multisensor system with possibly time disordered measurements. The goal of this work is to define a state estimator that takes into account the random availability of samples (non-uniform sampling and out-of-sequence measurements) and time-varying delays.

Several works have been published in order to adapt the original Kalman filter to any sampling scenario when there are no delays in the system or measurement device. For example, in Sanchis and Albertos (2002), the Kalman filter is applied to the sensorial fusion for output prediction under any sampling scenario without delay (synchronous, asynchronous, periodic and random). In Zhang et al. (2004) an optimal estimator is developed for systems that receive measurements of different nature: continuous, multirate and random, but also without delay. In Li et al. (2005) the Kalman filter equations are developed for the case of multirate control, where the measurements and input updates follow a given pattern. In Gupta et al. (2005) a sensor scheduling is addressed to minimize the state error covariance, assuming that only one sensor is active in every sampling instant. This scheme it is not applicable if random availability and time delays are present on the sensors or in the communication channel.

Also, different works have been developed trying to make use of the delayed measurements in the Kalman filter. In that sense, Hsiao and Pan (1996) proposes an extended order model when a reduced constant delay is present. In Larsen et al. (1998) the introduction of delayed measurements is dealt with by extrapolating the delayed measurement to the actual instant using the previous estimations of the Kalman filter, and calculating the optimum gain for that extrapolated measurement. For that purpose, the covariance matrix is updated at the delayed instant, i.e., at the instant where the sample was taken, and before the measurement is available. For this reason the Kalman filter is suboptimal until the measurement arrives, and the authors propose the computation of two parallel Kalman filters to avoid this fact. For this strategy to be applied, the device that computes the Kalman filter needs to know when the samples of the output occur, before the measurement is available, that in many cases is not a practical situation. The methodology presented there do not address the existence of several sensors with different delays and availability.

Reorganization technique of delayed measurements has been used to avoid the use of an extended order model leading in Lu et al. (2005) to $\bar{d}$ Kalman filters (being $\bar{d}$ the maximum discrete delay) to be computed in parallel, and leading in Wang et al. (2005) to an algorithm that is only updated when a set of $\bar{d}$ measurements of every sensor is gathered, delaying the use of the available information in the Kalman filter, and leading to a complex and high computational cost algorithm.

In Basin and Martinez-Zuñiga (2004), a continuous time optimum filter is derived for continuous time systems with multiple, but constant measurement delays. In Basin et al. (2005), a Kalman filter is derived also for continuous time systems with constant state delay, while in Zhang et al. (2007) an optimal state estimator for continuous systems with both continuous and sampled measurements with delays is proposed.

In Julier and Uhlmann (2005), the fusion of delayed measurements is studied, but the assumption is that the time delays are uncertain.

More recently, in Gopalakrishnan et al. (2011), a comparative review of different strategies to incorporate delayed and infrequent measurements in Extended Kalman Filter estimators is developed. The first alternative described is the full filter recalculation every time a delayed measurement arrives. This implies to store all past states, measurement updates, gains and covariance matrices till the maximum possible delay. This strategy is optimum, but impractical for long delays due to the high computational cost. The second approach is called Alexander's method, and consists of updating the gain and covariance matrices from the instant when the measurement was taken, till the measurement is received, and then to add a correction term

to the state estimate. The drawback of this strategy is that requires the algorithm to know the instant when the measurement is taken before the measurement is available. The third alternative is the same as the previous one, but adding a second Kalman filter that is computed from the instant when the measurement is taken till it arrives, and has the same drawback as the previous, the measuring instant must be known before the measurement is received. The other algorithms studied in that work are based on the state augmentation. The first one is the fixed-lag smoothing, that consists of augmenting the state with the past $\bar{d}$ states (where $\bar{d}$ is the maximum delay). A conventional Kalman filter is applied to the resulting $\bar{d}n$ order model. This filter is optimum, but it implies a high computational cost for long delays. If the measurements dimension is lower than the state dimension, a lower cost alternative is to augment the state with the estimates of the measurements. The resulting model order is lower, but it may still be large for long delays. The lowest cost state augmentation approach is the sample-sate augmentation, where the state is augmented only with the value of the state at the measuring instant, but the time when the measurement was taken is needed before the measurement is received.

The most recent published work is Moayedi et al. (2011). In this work, a simplified state augmentation approach is proposed. The state is augmented with the past $\bar{d}$ states. The simplification consists of only updating the current state when a measurement is available, leaving unchanged the previous state estimates. The updating gain is then of order n, and not $\bar{d}n$. The size of the variance covariance matrix is $\bar{d}n \times \bar{d}n$, but the computational cost of the updating equation of this matrix is not so high due to the special structure of the updating gain, that has only n nonzero rows. Of course, the approach leads to a suboptimal estimation, compared to the full state augmentation schemes. This approach can be applied if the delay is only known when the measurement is received, and admits out of sequence measurements.

In the present work, the authors propose a different approach for the state estimation in systems with scarce measurements with time varying delays, that can be applied when the delay is only known at the instant when the measurement is received, and also admits out of sequence measurements (as can occur in networked control systems). As in Moayedi et al. (2011), the idea is to reduce the computational cost of the state augmentation schemes. In this case, a small (order n) variance covariance matrix is calculated and stored, but an augmented state of size $\bar{d}n$ is used. The updating gain is also of order n, but the full augmented state vector is updated every time a measurement is received. The approach is based on the use of negative-time measurement update, and is first developed for general time variant systems, and then particularized for time invariant systems. The approach is suboptimal, but the loss in performance is low compared to the optimal filter, and the computational cost is much lower. The proposed algorithm can be applied in any sampling situation: multisensor sets with different sampling rate, time-varying sampling, random availability of measurements, different time-varying delays in each sensor and out-of-sequence measurements. Compared with the approach in Moayedi et al. (2011), the computational cost is similar, but the performance is better, especially if the system is unstable, as it is shown in the comparative example developed.

The layout of this paper is as follows: In section 2 the plant and the sampling scenario are described. In section 3 the proposed estimation algorithm is described. The particularization of the algorithm for linear time invariant systems is detailed in in section 4. Some numerical examples are analyzed on section 5, showing the validity of the proposal. Finally, on section 6 the main conclusions are summarized.

2. Problem statement

The process is assumed to be a discrete-time linear time-variant system modeled by the equations

$$\boldsymbol{x}[t+1] = \boldsymbol{A}[t]\,\boldsymbol{x}[t] + \boldsymbol{B}[t]\,\boldsymbol{u}[t] + \boldsymbol{w}[t], \tag{1a}$$

$$\boldsymbol{z}[t] = \boldsymbol{C}_z[t]\,\boldsymbol{x}[t], \tag{1b}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the state, $\boldsymbol{u} \in \mathbb{R}^{n_u}$ is the known input vector, $\boldsymbol{w} \in \mathbb{R}^n$ is the unmeasurable state disturbance vector and $\boldsymbol{z} \in \mathbb{R}^{n_z}$ is the targeted variables output vector.

As previously mentioned, the sensor measurements are assumed to be only available at discrete instants of time $t = t_k$, being a noisy and delayed function of the measurable outputs:

$$m_{i,k} = \boldsymbol{c}_{i,k}\,\boldsymbol{x}[t_k - d_{i,k}] + v_{i,k}, \quad i = 1, \ldots, n_m \tag{2}$$

where $d_{i,k}$ is the discrete delay (measured in number of control periods) assigned to sensor i in the k-th sampling (not all sensor measurements are available at each sampling time). The elapsed time between the last two consecutive sampling instants is denoted by

$$N_k = t_k - t_{k-1},$$

and $\bar{d}$ is an upper bound of sensor delay. The availability factor $\alpha_{i,k}$ defined as

$$\alpha_{i,k} = \begin{cases} 1, \text{ if } m_{i,k} \text{ is available,} \\ 0, \text{ if } m_{i,k} \text{ is not available,} \end{cases} \tag{3}$$

indicates which are the available measurements at the k-th sampling instant, and the number of available measurements is denoted by μ_k.

3. State estimation algorithm

The problem to be solved is how to use the available measurements of the plant that are assumed to be scarce, with different delays, availability and with the possibility of time-disordered information. Time-tagged measurements are assumed in order to solve this problem, but the time when the measurement was taken is only known after the measurement is received. This assumption is realistic in networked control systems and with slow sensors that need some time to give a measurement value from a taken sample.

The novelty of the proposed approach lies in the direct utilization of the delayed available measurements in the correction of the state estimation, without the necessity of running the Kalman filter backwards in order to introduce the measurements time ordered, and without the necessity of waiting until a complete set of sensors measurements are available. This allows to deal with all the available information and even dealing with measurements that arrive disordered in time. This problem has been solved by other authors with a state augmentation strategy, but the resulting computational cost is very high for long delays. In the presented approach, a standard size variance covariance matrix is used, leading to a much lower computational cost, at the expense of a slightly suboptimal estimation.

The Kalman filter is based on the computation of both an estimate of the system state $\hat{\boldsymbol{x}}[t]$ and the state estimation error covariance matrix defined as

$$\boldsymbol{P}[t] = \mathcal{E}\{(\boldsymbol{x}[t] - \hat{\boldsymbol{x}}[t])(\boldsymbol{x}[t] - \hat{\boldsymbol{x}}[t])^{\mathsf{T}}\}. \tag{4}$$

The computation is done by the propagation equation, where an initial estimation of state ($\hat{\boldsymbol{x}}[t|t-1]$) and covariance ($\boldsymbol{P}[t|t-1]$) is calculated using the process model, and by the updating equation, where the available measurements information is used to update the previous estimations (leading to $\hat{\boldsymbol{x}}[t] = \hat{\boldsymbol{x}}[t|t]$ and $\hat{\boldsymbol{P}}[t] = \hat{\boldsymbol{P}}[t|t]$).

3.1. Propagation

The propagation equations must be calculated at every sampling instant T. The state estimation propagation equation is

$$\hat{\boldsymbol{x}}[t|t-1] = \boldsymbol{A}[t-1]\,\hat{\boldsymbol{x}}[t-1] + \boldsymbol{B}[t-1]\,\boldsymbol{u}[t-1], \tag{5a}$$

and the state estimation error covariance matrix propagation equation is

$$\boldsymbol{P}[t|t-1] = \boldsymbol{A}[t-1]\boldsymbol{P}[t-1]\boldsymbol{A}[t-1]^{\mathsf{T}} + \boldsymbol{W},$$

where $\boldsymbol{P}[t|t-1]$ is defined as

$$\boldsymbol{P}[t|t-1] = \mathcal{E}\{(\boldsymbol{x}[t] - \hat{\boldsymbol{x}}[t|t-1])(\boldsymbol{x}[t] - \hat{\boldsymbol{x}}[t|t-1])^{\mathsf{T}}\}.$$

When there is no measurement available, there is no possible update, and the best state and covariance estimations are the previous (open loop) ones, leading to

$$\hat{\boldsymbol{x}}[t] = \hat{\boldsymbol{x}}[t|t-1],$$

and

$$\boldsymbol{P}[t] = \boldsymbol{P}[t|t-1].$$

3.2. Update

The updating equations are only applicable when there is some measurement available at $t = t_k$. The proposed state estimation updating equation is

$$\hat{\boldsymbol{x}}[t_k] = \hat{\boldsymbol{x}}[t_k|t_k-1] + \sum_{i=1}^{n_m} \boldsymbol{\ell}_{i,k}\left(m_{i,k} - \boldsymbol{c}_{i,k}\hat{\boldsymbol{x}}[t_k - d_{i,k}|t_k-1]\right)\alpha_{i,k} \tag{5b}$$

where vector $\hat{\boldsymbol{x}}[t_k - d_{i,k}|t_k-1]$ is the best estimation of the state at instant $t_k - d_{i,k}$ with all the information gathered until instant $t_k - 1$ and assuming null disturbances from $t_k - d_{i,k}$ to t_k, fulfilling

$$\hat{\boldsymbol{x}}[t_k|t_k-1] = \left(\prod_{j=1}^{d_{i,k}} \boldsymbol{A}[t_k - j]\right)\hat{\boldsymbol{x}}[t_k - d_{i,k}|t_k-1] + \sum_{l=1}^{d_{i,k}}\left(\prod_{j=1}^{l-1} \boldsymbol{A}[t_k - j]\right)\boldsymbol{B}[t_k - l]\boldsymbol{u}[t_k - l],$$

and then, can be computed as

$$\hat{\boldsymbol{x}}[t_k - d_{i,k}|t_k-1] = \left(\prod_{j=1}^{d_{i,k}} \boldsymbol{A}[t_k - j]\right)^{-1}\hat{\boldsymbol{x}}[t_k|t_k-1] - \sum_{l=1}^{d_{i,k}}\left(\prod_{j=l}^{d_{i,k}} \boldsymbol{A}[t_k - j]\right)^{-1}\boldsymbol{B}[t_k - l]\boldsymbol{u}[t_k - l]. \tag{5c}$$

Vector $\boldsymbol{\ell}_{i,k}$ is the updating gain for the i-th sensor (there only exists vector $\boldsymbol{\ell}_{i,k}$ at t_k if $\alpha_{i,k} = 1$). As a result of the analysis developed in theorem 3.1, the state estimation error covariance matrix is proposed to be updated as

$$\boldsymbol{P}[t_k] = (\boldsymbol{I} - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k}) \, \boldsymbol{P}[t_k|t_{k-1}] + \boldsymbol{L}_k \boldsymbol{M}_k \bar{\boldsymbol{C}}_{d,k} \bar{\boldsymbol{W}} \bar{\boldsymbol{A}}_k^\mathsf{T}. \tag{5d}$$

where $\boldsymbol{L}_k$ is the matrix formed with gains corresponding to available sensors, and $\boldsymbol{M}_k$ is the $\mu_k \times n_m$ matrix formed with the rows of an identity matrix that correspond to the position of the elements $m_{i,k}$ available at instant t_k. Therefore, vector $\boldsymbol{\ell}_{i,k}$ corresponds to the i-th row of matrix $\boldsymbol{L}_k \boldsymbol{M}_k$. Note that, if there is no information available from sensor j, then $\boldsymbol{\ell}_{j,k} = \boldsymbol{0}$. $\boldsymbol{C}_{d,k}$ is the $n_m \times n$ matrix whose i-th row is given by

$$[\boldsymbol{C}_{d,k}]_{\{i,:\}} = \boldsymbol{c}_{i,k} \left(\prod_{j=1}^{d_{i,k}} \boldsymbol{A}[t_k - j] \right)^{-1},$$

and $\bar{\boldsymbol{C}}_{d,k}$ the $n_m \times n\bar{d}$ matrix whose i-th row is given by

$$[\bar{\boldsymbol{C}}_{d,k}]_{\{i,:\}} = \boldsymbol{c}_{i,k} \left[\left(\prod_{j=1}^{d_{i,k}} \boldsymbol{A}[t_k - j] \right)^{-1} \left(\prod_{j=2}^{d_{i,k}} \boldsymbol{A}[t_k - j] \right)^{-1} \cdots \boldsymbol{A}[t_k - d_{i,k}]^{-1} \boldsymbol{0} \cdots \boldsymbol{0} \right].$$

If there is no measurement of sensor i, or if there is no delay associated to the i-th sensor in the k-th measurement, then, the i-th row of both $\boldsymbol{C}_{d,k}$ and $\bar{\boldsymbol{C}}_{d,k}$ are null. Matrix $\bar{\boldsymbol{A}}_k$ contains the product of delayed state matrices

$$\bar{\boldsymbol{A}}_k = \left[\boldsymbol{I} \; \boldsymbol{A}[t_k - 1] \; \boldsymbol{A}[t_k - 1]\boldsymbol{A}[t_k - 2] \; \cdots \; \prod_{j=1}^{\bar{d}-1} \boldsymbol{A}[t_k - j], \right] \tag{5e}$$

and matrix $\bar{\boldsymbol{W}}$ is the matrix of the state disturbance for all possible delays:

$$\bar{\boldsymbol{W}} = \mathrm{diag}\{\boldsymbol{W}, \ldots, \boldsymbol{W}\}_{n\bar{d} \times n\bar{d}}.$$

Finally, the process output is estimated at every sampling time as

$$\hat{\boldsymbol{z}}[t] = \boldsymbol{C}_z[t] \, \hat{\boldsymbol{x}}[t]. \tag{5f}$$

The next theorem defines the matrix gain that leads to the optimal estimation of the state with the state updating equation (5b).

Theorem 3.1 : *Assume that the disturbance $\boldsymbol{w}[t]$ in system (1) is a white noise with covariance matrix $\boldsymbol{W}$, and that the sensors noises in (2) are white noises with variances σ_i ($i = 1, \ldots, n_m$). Define $\boldsymbol{V} = \mathrm{diag}\{\sigma_1, \ldots, \sigma_{n_m}\}$. Assume also that the measurements are received time ordered. Under these assumptions, the gain matrix $\boldsymbol{L}_k$ that minimizes the error variance of the algorithm (5) is defined by*

$$\boldsymbol{L}_k = \left(\boldsymbol{P}[t_k|t_k - 1]\boldsymbol{C}_{d,k}^\mathsf{T} - \boldsymbol{S}_k \right) \boldsymbol{M}_k^\mathsf{T} \cdot$$
$$\cdot \left(\boldsymbol{M}_k \left(\boldsymbol{C}_{d,k} \boldsymbol{P}[t_k|t_k - 1]\boldsymbol{C}_{d,k}^\mathsf{T} + \boldsymbol{V} + \bar{\boldsymbol{C}}_{d,k} \bar{\boldsymbol{W}} \bar{\boldsymbol{C}}_{d,k}^\mathsf{T} - \boldsymbol{C}_{d,k} \boldsymbol{S}_k - \boldsymbol{S}_k^\mathsf{T} \boldsymbol{C}_{d,k}^\mathsf{T} \right) \boldsymbol{M}_k^\mathsf{T} \right)^{-1}, \tag{6}$$

with

$$\boldsymbol{S}_k = \bar{\boldsymbol{A}}_k \bar{\boldsymbol{W}} \bar{\boldsymbol{C}}_{d,k}^\mathsf{T}.$$

Proof: The state at instant t_k can be related with the state at instant $t_k - d_{i,k}$ as

$$\boldsymbol{x}[t_k] = \left(\prod_{j=1}^{d_{i,k}} \boldsymbol{A}[t_k - j]\right) \boldsymbol{x}[t_k - d_{i,k}] + \sum_{l=1}^{d_{i,k}} \left(\prod_{j=1}^{l-1} \boldsymbol{A}[t_k - j]\right) (\boldsymbol{B}[t_k - l]\boldsymbol{u}[t_k - l] + \boldsymbol{w}[t_k - l]),$$

and then, measurement $m_{i,k}$ can be written as

$$m_{i,k} = \boldsymbol{c}_{i,k} \left(\prod_{j=1}^{d_{i,k}} \boldsymbol{A}[t_k - j]\right)^{-1} \boldsymbol{x}[t_k] - \boldsymbol{c}_{i,k} \sum_{l=1}^{d_{i,k}} \left(\prod_{j=l}^{d_{i,k}} \boldsymbol{A}[t_k - j]\right)^{-1} (\boldsymbol{B}[t_k - l]\boldsymbol{u}[t_k - l] + \boldsymbol{w}[t_k - l]) + v_{i,k}.$$

Introducing this expression in update equation (5b) and subtracting $x[t]$ in both sides of the resulting expression, it leads to

$$\tilde{\boldsymbol{x}}[t_k] = (\boldsymbol{I} - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k}) \tilde{\boldsymbol{x}}[t_k|t_k - 1] - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{v}_k + \boldsymbol{L}_k \boldsymbol{M}_k \bar{\boldsymbol{C}}_{d,k} \bar{\boldsymbol{w}}_{d,k}, \tag{7}$$

being $\tilde{\boldsymbol{x}}[t] = \boldsymbol{x}[t] - \hat{\boldsymbol{x}}[t]$ the state estimation error, and where $\boldsymbol{M}_k \boldsymbol{v}_k$ is the vector containing the noises of available sensors (size μ_k), and $\bar{\boldsymbol{w}}_{d,k}$ is extended disturbance vector containing the history of signal $\boldsymbol{w}[t]$ from $t_k - \bar{d}$ till $t_k - 1$:

$$\bar{\boldsymbol{w}}_{d,k} = \begin{bmatrix} \boldsymbol{w}[t_k - 1] \\ \vdots \\ \boldsymbol{w}[t_k - \bar{d}] \end{bmatrix}$$

Using expression (7) in the definition of covariance matrix (4) at sampling instants $t = t_k$, it can be written

$$\begin{aligned}
\boldsymbol{P}[t_k] = {}& (\boldsymbol{I} - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k}) \, \mathcal{E}\{\tilde{\boldsymbol{x}}[t_k|t_k - 1] \, \tilde{\boldsymbol{x}}[t_k|t_k - 1]^{\mathsf{T}}\} \, (\boldsymbol{I} - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k})^{\mathsf{T}} \\
& + \boldsymbol{L}_k \boldsymbol{M}_k \mathcal{E}\{\boldsymbol{v}_k \boldsymbol{v}_k^{\mathsf{T}}\} \boldsymbol{M}_k^{\mathsf{T}} \boldsymbol{L}_k^{\mathsf{T}} + \boldsymbol{L}_k \boldsymbol{M}_k \bar{\boldsymbol{C}}_{d,k} \mathcal{E}\{\bar{\boldsymbol{w}}_{d,k} \, \bar{\boldsymbol{w}}_{d,k}^{\mathsf{T}}\} \bar{\boldsymbol{C}}_{d,k}^{\mathsf{T}} \boldsymbol{M}_k^{\mathsf{T}} \boldsymbol{L}_k^{\mathsf{T}} \\
& + (\boldsymbol{I} - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k}) \, \mathcal{E}\{\tilde{\boldsymbol{x}}[t_k|t_k - 1]\bar{\boldsymbol{w}}_{d,k}^{\mathsf{T}}\} \bar{\boldsymbol{C}}_{d,k}^{\mathsf{T}} \boldsymbol{M}_k^{\mathsf{T}} \boldsymbol{L}_k^{\mathsf{T}} \\
& + \boldsymbol{L}_k \boldsymbol{M}_k \bar{\boldsymbol{C}}_{d,k} \mathcal{E}\{\bar{\boldsymbol{w}}_{d,k}\tilde{\boldsymbol{x}}[t_k|t_k - 1]^{\mathsf{T}}\} \, (\boldsymbol{I} - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k})^{\mathsf{T}} \tag{8}
\end{aligned}$$

where it has been considered that $\mathcal{E}\{\boldsymbol{w}[t_k - a] \, \boldsymbol{w}[t_k - b]^{\mathsf{T}}\} = \boldsymbol{0}$, $a \neq b$. As estimation error $\tilde{\boldsymbol{x}}[t_k|t_k - 1]$ is affected by the disturbances $\boldsymbol{w}[t_k - j]$ for $j = 1, \ldots, \bar{d}$, the expectation $\mathcal{E}\{\tilde{\boldsymbol{x}}[t_k|t_k-1]\bar{\boldsymbol{w}}_{d,k}^{\mathsf{T}}\}$ and its transposed must be calculated. In order to do this, the estimation error $\tilde{\boldsymbol{x}}[t_k|t_k - 1]$ is written as a function of $\tilde{\boldsymbol{x}}[t_k - \bar{d}|t_k - 1]$ and all the disturbances from $t_k - \bar{d}$ to $t_k - 1$, taking into account the assumption of time ordered measurements, leading to

$$\tilde{\boldsymbol{x}}[t_k|t_k - 1] = \left(\prod_{j=1}^{\bar{d}} \boldsymbol{A}[t_k - j]\right) \tilde{\boldsymbol{x}}[t_k - \bar{d}|t_k - 1] + \sum_{l=1}^{\bar{d}} \left(\prod_{j=1}^{l-1} \boldsymbol{A}[t_k - j]\right) \boldsymbol{w}[t_k - l], \tag{9}$$

and, therefore, the expectation $\mathcal{E}\{\tilde{\boldsymbol{x}}[t_k|t_k-1]\bar{\boldsymbol{w}}_{d,k}^{\mathsf{T}}\}$ is given by

$$\mathcal{E}\{\tilde{\boldsymbol{x}}[t_k|t_k-1]\bar{\boldsymbol{w}}_{d,k}^{\mathsf{T}}\} = \bar{\boldsymbol{A}}_k \bar{\boldsymbol{W}}, \tag{10}$$

as $\tilde{\boldsymbol{x}}[t_k - \bar{d}|t_k - 1]$ is not affected by $\bar{\boldsymbol{w}}_{d,k}$. With this expectation, the covariance matrix reduces to

$$
\begin{aligned}
\boldsymbol{P}[t_k] =&\, \boldsymbol{P}[t_k|t_k-1] - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k} \boldsymbol{P}[t_k|t_k-1] - \boldsymbol{P}[t_k|t_k-1] \left(\boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k} \right)^{\mathsf{T}} \\
&+ \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k} \boldsymbol{P}[t_k|t_k-1] \boldsymbol{C}_{d,k}^{\mathsf{T}} \boldsymbol{M}_k^{\mathsf{T}} \boldsymbol{L}_k^{\mathsf{T}} + \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{V} \boldsymbol{M}_k^{\mathsf{T}} \boldsymbol{L}_k^{\mathsf{T}} + \boldsymbol{L}_k \boldsymbol{M}_k \bar{\boldsymbol{C}}_{d,k} \bar{\boldsymbol{W}} \bar{\boldsymbol{C}}_{d,k}^{\mathsf{T}} \boldsymbol{M}_k^{\mathsf{T}} \boldsymbol{L}_k^{\mathsf{T}} \\
&+ \boldsymbol{S}_k \boldsymbol{M}_k^{\mathsf{T}} \boldsymbol{L}_k^{\mathsf{T}} + \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{S}_k^{\mathsf{T}} - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{C}_{d,k} \boldsymbol{S}_k \boldsymbol{M}_k^{\mathsf{T}} \boldsymbol{L}_k^{\mathsf{T}} - \boldsymbol{L}_k \boldsymbol{M}_k \boldsymbol{S}_k^{\mathsf{T}} \boldsymbol{C}_{d,k}^{\mathsf{T}} \boldsymbol{M}_k^{\mathsf{T}} \boldsymbol{L}_k^{\mathsf{T}}.
\end{aligned}
\tag{11}
$$

To minimize the estate estimation error variance, the trace of matrix $\boldsymbol{P}[t_k]$ must be minimized. Calculating the matrix derivative of the trace of (11) with respect to $\boldsymbol{L}_k$, it is obtained that

$$
\frac{\partial \mathrm{tr}(\boldsymbol{P}[t_k])}{\partial \boldsymbol{L}_k} = -2 \left(\boldsymbol{P}[t_k|t_k - 1] \boldsymbol{C}_{d,k}^{\mathsf{T}} - \boldsymbol{A}_k \bar{\boldsymbol{W}} \bar{\boldsymbol{C}}_{d,k} \right) \boldsymbol{M}_k^{\mathsf{T}}
\tag{12}
$$

$$
+ 2 \boldsymbol{L}_k \left(\boldsymbol{M}_k \left(\boldsymbol{C}_{d,k} \boldsymbol{P}[t_k|t_k - 1] \boldsymbol{C}_{d,k}^{\mathsf{T}} + \boldsymbol{V} + \bar{\boldsymbol{C}}_{d,k} \bar{\boldsymbol{W}} \bar{\boldsymbol{C}}_{d,k}^{\mathsf{T}} - \boldsymbol{C}_{d,k} \boldsymbol{S}_k - \boldsymbol{S}_k^{\mathsf{T}} \boldsymbol{C}_{d,k}^{\mathsf{T}} \right) \boldsymbol{M}_k^{\mathsf{T}} \right).
\tag{13}
$$

The matrix gain that makes zero this expression is the updating gain (6) defined on theorem. If this gain is substituted in equation (11), it leads to the covariance updating equation (5d) exposed in the proposed algorithm. $\qquad\square$

Remark 1: The previous theorem is valid (and the estimate is then optimum) if the measurements are assumed to be received time ordered. Nevertheless, the resulting algorithm can also be applied if this is not the case (i.e. if an older measurement is received after a newer one). In that case, the gain defined by the theorem is suboptimal. However, the low performance loss due to this suboptimality, together with the high reduction in computational load with respect full filter recalculation or full state augmentation approaches, makes the proposed algorithm very useful in practice.

Remark 2: The proposed algorithm requires the inversion of matrices $\boldsymbol{A}[t]$. This is only possible if the matrix is non singular, i.e., if there are no eigenvalues on the origin for any $\boldsymbol{A}[t]$. The system will have as many delays from input to state as eigenvalues on the origin so, in order to avoid singularity, the delay must be transmitted to the inputs, that are assumed to be known and, therefore, can be stored in a new delayed variable $u'[t] = u[t - \sigma]$, being σ the input to state delay. With this, matrix $\boldsymbol{A}[t]$ will be non-singular and the previous algorithm will be applicable.

Remark 3: If the process is state delayed with model equations

$$
\boldsymbol{x}[t + 1] = \boldsymbol{A}[t]\boldsymbol{x}[t] + \boldsymbol{A}_\sigma[t]\boldsymbol{x}[t - \sigma] + \boldsymbol{B}[t]\boldsymbol{u}[t],
\tag{14}
$$

an extended order notation must be used in order to apply the approaches in this work. The model equations must be written as

$$
\underbrace{\begin{bmatrix} \boldsymbol{x}[t + 1] \\ \boldsymbol{x}[t] \\ \vdots \\ \boldsymbol{x}[t - \sigma + 1] \end{bmatrix}}_{} = \underbrace{\begin{bmatrix} \boldsymbol{A}[t]\ 0\ \ldots\ \boldsymbol{A}_\sigma[t] \\ \phantom{\boldsymbol{A}[t]}\ \ \ \ \ 0 \\ \boldsymbol{I} \quad \vdots \\ \phantom{\boldsymbol{I}}\quad\ \ 0 \end{bmatrix}}_{\boldsymbol{A}'[t]} \underbrace{\begin{bmatrix} \boldsymbol{x}[t] \\ \boldsymbol{x}[t - 1] \\ \vdots \\ \boldsymbol{x}[t - \sigma] \end{bmatrix}}_{\boldsymbol{x}'[t]} + \underbrace{\begin{bmatrix} \boldsymbol{B}[t] \\ 0 \\ \vdots \end{bmatrix}}_{\boldsymbol{B}'[t]} \boldsymbol{u}[t].
\tag{15}
$$

This equation can be rewritten as

$$
\boldsymbol{x}'[t + 1] = \boldsymbol{A}'[t]\boldsymbol{x}'[t] + \boldsymbol{B}'[t]\boldsymbol{u}[t]
\tag{16}
$$

where $\boldsymbol{x}'[t]$, $\boldsymbol{A}'[t]$ and $\boldsymbol{B}'[t]$ are the matrices and vectors that would apply in the algorithm explained in this work.

Remark 4: If there is a non-zero mean disturbance modeled by a random walk (a constant disturbance would be a particular case), an extended order model including the disturbance as a new state can be written. Let w_1 the gaussian disturbance with zero mean, and w_2 a random walk. Then, the system equations can be written as

$$\boldsymbol{x}[t+1] = \boldsymbol{A}[t]\boldsymbol{x}[t] + \boldsymbol{B}[t]\boldsymbol{u}[t] + \boldsymbol{w}_1[t] + \boldsymbol{w}_2[t],$$
$$\boldsymbol{z}[t] = \boldsymbol{C}_z\boldsymbol{x}[t].$$

As $\boldsymbol{w}_2$ is a random walk, it can written that

$$\boldsymbol{w}_2[t] = \boldsymbol{w}_2[t-1] + \Delta\boldsymbol{w}_2[t-1],$$

where $\Delta\boldsymbol{w}_2[t-1]$ is a Gaussian zero-mean variable. The system equations can be written as

$$\begin{bmatrix} \boldsymbol{x}[t+1] \\ \boldsymbol{w}_2[t+1] \end{bmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{A}[t] & \boldsymbol{I} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}}_{\boldsymbol{A}'[t]} \underbrace{\begin{bmatrix} \boldsymbol{x}[t] \\ \boldsymbol{w}_2[t] \end{bmatrix}}_{\boldsymbol{x}'[t]} + \underbrace{\begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{0} \end{bmatrix}}_{\boldsymbol{B}'[t]} \boldsymbol{u}[t] + \underbrace{\begin{bmatrix} \boldsymbol{w}_1[t] \\ \Delta\boldsymbol{w}_2[t] \end{bmatrix}}_{\boldsymbol{w}'[t]},$$

$$\boldsymbol{z}[t] = \underbrace{\begin{bmatrix} \boldsymbol{C}_z & \boldsymbol{0} \end{bmatrix}}_{\boldsymbol{C}_z'[t]} \begin{bmatrix} \boldsymbol{x}[t] \\ \boldsymbol{w}_2[t] \end{bmatrix}.$$

Using the new state vector $\boldsymbol{x}'[t] = \begin{bmatrix} \boldsymbol{x}[t] \\ \boldsymbol{w}_2[t] \end{bmatrix}$, and the new state disturbance vector $\boldsymbol{w}'[t] = \begin{bmatrix} \boldsymbol{w}_1[t] \\ \Delta\boldsymbol{w}_2[t] \end{bmatrix}$, the model is reduced to a system affected by Gaussian disturbances, where the proposed approach can be directly applied.

Remark 5: The previous representation is also valid when some of the sensors have a non negligible dynamic. To demonstrate it, take the process difference equations

$$\boldsymbol{x}_p[t+1] = \boldsymbol{A}_p[t]\boldsymbol{x}_p[t] + \boldsymbol{B}_p[t]\boldsymbol{u}[t] + \boldsymbol{w}_p[t], \tag{17}$$
$$\boldsymbol{y}[t] = \boldsymbol{C}_p[t]\boldsymbol{x}_p[t], \tag{18}$$

where $\boldsymbol{y}$ are the signals connected to the different sensors, and the difference equations associated to the sensors

$$\boldsymbol{x}_s[t+1] = \boldsymbol{A}_s[t]\boldsymbol{x}_s[t] + \boldsymbol{B}_s[t]\boldsymbol{y}[t] + \boldsymbol{w}_s[t] \tag{19}$$
$$m_{i,k} = \boldsymbol{c}_{s,i,k}\,\boldsymbol{x}_s[t_k - d_{i,k}] + \boldsymbol{d}_{y,i,k}\,\boldsymbol{y}[t_k - d_{i,k}] + v_{i,k}, \quad i = 1, \ldots, n_m. \tag{20}$$

Grouping both equations it can be expressed

$$\underbrace{\begin{bmatrix} \boldsymbol{x}_p[t+1] \\ \boldsymbol{x}_s[t+1] \end{bmatrix}}_{} = \underbrace{\begin{bmatrix} \boldsymbol{A}_p[t] & \boldsymbol{0} \\ \boldsymbol{B}_s[t]\boldsymbol{C}_p[t] & \boldsymbol{A}_s[t] \end{bmatrix}}_{=\boldsymbol{A}[t]} \begin{bmatrix} \boldsymbol{x}_p[t] \\ \boldsymbol{x}_s[t] \end{bmatrix} + \underbrace{\begin{bmatrix} \boldsymbol{B}_p[t] \\ \boldsymbol{0} \end{bmatrix}}_{=\boldsymbol{B}[t]} \boldsymbol{u}[t] + \underbrace{\begin{bmatrix} \boldsymbol{w}_p[t] \\ \boldsymbol{w}_s[t] \end{bmatrix}}_{=\boldsymbol{w}[t]} \tag{21}$$

$$\boldsymbol{y}[t] = \underbrace{\begin{bmatrix} \boldsymbol{C}_p[t] & \boldsymbol{0} \end{bmatrix}}_{=\boldsymbol{C}_y[t]} \begin{bmatrix} \boldsymbol{x}_p[t] \\ \boldsymbol{x}_s[t] \end{bmatrix} \tag{22}$$

$$m_{i,k} = \underbrace{\begin{bmatrix} \boldsymbol{d}_{y,i,k}\boldsymbol{C}_p[t_k - d_{i,k}] & \boldsymbol{c}_{s,i,k} \end{bmatrix}}_{=\boldsymbol{c}_{i,k}} \begin{bmatrix} \boldsymbol{x}_p[t_k - d_{i,k}] \\ \boldsymbol{x}_s[t_k - d_{i,k}] \end{bmatrix} + v_{i,k}. \tag{23}$$

Taking the state of the whole plant (process plus sensors) as the one that joints the process state plus the sensors state $\boldsymbol{x}[t] = \begin{bmatrix} \boldsymbol{x}_p[t] \\ \boldsymbol{x}_s[t] \end{bmatrix}$, the representation (1) is reached.

3.3. *Implementation of the estimator*

In this section an alternative way of implementing the proposed estimator is addressed. The idea is to use an extended state in order to avoid the multiple matrix inversions that are needed in the algorithm exposed in the previous section.

Let us define the extended state vector and system matrices as

$$\hat{\mathbb{X}}[t|t-1] = \begin{bmatrix} \hat{\boldsymbol{x}}[t|t-1] \\ \hat{\boldsymbol{x}}[t-1|t-1] \\ \vdots \\ \hat{\boldsymbol{x}}[t-\bar{d}|t-1] \end{bmatrix} ; \quad \mathbb{A}[t] = \left[\begin{array}{c|c} \boldsymbol{A}[t]\ \boldsymbol{0}\cdots\ \boldsymbol{0} & \\ \hline & \vdots \\ \boldsymbol{I}_{\bar{d}n} & \\ & \boldsymbol{0} \end{array} \right] ; \quad \mathbb{B}[t] = \begin{bmatrix} \boldsymbol{B}[t] \\ \boldsymbol{0} \\ \vdots \\ \boldsymbol{0} \end{bmatrix} ;$$

and a matrix $\boldsymbol{D}[t]$ containing all the inverse matrices needed in the implementation

$$\boldsymbol{D}[t] = \begin{bmatrix} \boldsymbol{A}[t-1]^{-1} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{A}[t-2]^{-1}\boldsymbol{A}[t-1]^{-1} & \boldsymbol{A}[t-2]^{-1} & & \vdots \\ \vdots & & \ddots & \\ (\prod_{j=1}^{\bar{d}} \boldsymbol{A}[t-j])^{-1} & (\prod_{j=2}^{\bar{d}} \boldsymbol{A}[t-j])^{-1} & \cdots & \boldsymbol{A}[t-\bar{d}]^{-1} \end{bmatrix}.$$

Making use of these matrices, the resulting algorithm can be implemented with the propagation equations

$$\hat{\mathbb{X}}[t|t-1] = \mathbb{A}[t-1]\,\hat{\mathbb{X}}[t-1] + \mathbb{B}[t-1]\,\boldsymbol{u}[t-1], \tag{24a}$$

$$\boldsymbol{P}[t|t-1] = \boldsymbol{A}[t-1]\boldsymbol{P}[t-1]\boldsymbol{A}[t-1]^{\mathsf{T}} + \boldsymbol{W}, \tag{24b}$$

$$\bar{\boldsymbol{A}}[t] = \begin{bmatrix} \boldsymbol{I}_n & \boldsymbol{0} \end{bmatrix} + \boldsymbol{A}[t-1]\bar{\boldsymbol{A}}[t-1] \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I}_{\bar{d}n} \\ \boldsymbol{0}_n & \boldsymbol{0} \end{bmatrix} \tag{24c}$$

$$\boldsymbol{D}[t] = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0}_n \\ \boldsymbol{I}_{\bar{d}n} & \boldsymbol{0} \end{bmatrix} \boldsymbol{D}[t-1] \left(\begin{bmatrix} \boldsymbol{A}[t-1]^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0}_{\bar{d}n} \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I}_{\bar{d}n} \\ \boldsymbol{0}_n & \boldsymbol{0} \end{bmatrix} \right) + \begin{bmatrix} \boldsymbol{A}[t-1]^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0}_{\bar{d}n} \end{bmatrix} \tag{24d}$$

plus the updating equations

$$\boldsymbol{L}_k = \left(\boldsymbol{P}[t_k|t_k - 1]\boldsymbol{C}_{d,k}^\mathsf{T} - \boldsymbol{S}_k\right)\boldsymbol{M}_k^\mathsf{T} \cdot$$

$$\cdot \left(\boldsymbol{M}_k \left(\boldsymbol{C}_{d,k}\boldsymbol{P}[t_k|t_k - 1]\boldsymbol{C}_{d,k}^\mathsf{T} + \boldsymbol{V} + \bar{\boldsymbol{C}}_{d,k}\bar{\boldsymbol{W}}\bar{\boldsymbol{C}}_{d,k}^\mathsf{T} - \boldsymbol{C}_{d,k}\boldsymbol{S}_k - \boldsymbol{S}_k^\mathsf{T}\boldsymbol{C}_{d,k}^\mathsf{T}\right)\boldsymbol{M}_k^\mathsf{T}\right)^{-1} \quad (24\text{e})$$

$$\hat{\mathbb{X}}[t_k] = \hat{\mathbb{X}}[t_k|t_k - 1] + \Gamma_k \cdot \sum_{i=1}^{n_m} \boldsymbol{\ell}_{i,k}\left(m_{i,k} - \boldsymbol{c}_i\,\hat{\boldsymbol{x}}[t_k - d_{i,k}|t_k - 1]\right)\alpha_{i,k} \quad (24\text{f})$$

$$\boldsymbol{P}[t_k] = \left(\boldsymbol{I} - \boldsymbol{L}_k\boldsymbol{M}_k\boldsymbol{C}_{d,k}\right)\boldsymbol{P}[t_k|t_{k-1}] + \boldsymbol{L}_k\boldsymbol{M}_k\boldsymbol{S}_k^\mathsf{T}. \quad (24\text{g})$$

Matrices $\boldsymbol{C}_{d,k}$ and $\bar{\boldsymbol{C}}_{d,k}$ are obtained from $\boldsymbol{D}[t]$ as

$$[\bar{\boldsymbol{C}}_{d,k}]_{\{i,:\}} = [\,\overbrace{\boldsymbol{0}\ \cdots\ \boldsymbol{0}}^{n\cdot(d_{i,k}-1)}\ \boldsymbol{I}_n\ \boldsymbol{0}\ \cdots\ \boldsymbol{0}\,] \cdot \boldsymbol{D}[t_k]$$

$$[\boldsymbol{C}_{d,k}]_{\{i,:\}} = [\bar{\boldsymbol{C}}_{d,k}]_{\{i,:\}} \cdot \begin{bmatrix} \boldsymbol{I}_n \\ \boldsymbol{0} \end{bmatrix},$$

and $\boldsymbol{S}_k$ is computed by

$$\boldsymbol{S}_k = \bar{\boldsymbol{A}}[t_k]\bar{\boldsymbol{W}}\bar{\boldsymbol{C}}_{d,k}^\mathsf{T}.$$

Matrix Γ_k is defined as

$$\Gamma_k = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{A}[t_k - 1]^{-1} \\ \vdots \\ (\prod_{j=1}^{\bar{d}}\boldsymbol{A}[t_k - j])^{-1} \end{bmatrix}$$

that can be computed, to avoid the matrix inversions, as

$$\Gamma_k = \begin{bmatrix} \boldsymbol{I}_n \\ \boldsymbol{D}[t_k]\begin{bmatrix}\boldsymbol{I}_n \\ \boldsymbol{0}\end{bmatrix} \end{bmatrix}.$$

The delayed state estimation needed in the update equation can be directly extracted from the extended state as

$$\hat{\boldsymbol{x}}[t_k - d_{i,k}|t_k - 1] = [\,\overbrace{\boldsymbol{0}\ \cdots\ \boldsymbol{0}}^{n\cdot d_{i,k}}\ \boldsymbol{I}\ \boldsymbol{0}\ \cdots\ \boldsymbol{0}\,] \cdot \hat{\mathbb{X}}[t_k|t_k - 1].$$

Note that with this algorithm implementation, the computational cost due to matrix inversions is drastically reduced, but at the expense of an increment of the memory needed in the computing system.

4. State estimator for LTI systems

In the case of dealing with LTI systems, the matrices that define the behavior of the system (1) reduce to the constant ones

$$A[t] = A, \quad , B[t] = B, \quad , C_z[t] = C_z, \quad c_{i,k} = c_i \ (i = 1, \ldots, n_m),$$

Using this equations, the previous algorithm is reduced to the propagation equations

$$\hat{x}[t|t-1] = A\,\hat{x}[t-1] + B\,u[t-1], \tag{25a}$$

$$P[t|t-1] = AP[t-1]A^\mathsf{T} + W, \tag{25b}$$

plus the update equations

$$L_k = \left(P[t_k|t_k - 1]C_{d,k}^\mathsf{T} - S_k\right) M_k^\mathsf{T} \cdot$$
$$\cdot \left(M_k \left(C_{d,k} P[t_k|t_k - 1]C_{d,k}^\mathsf{T} + V + \bar{C}_{d,k}\bar{W}\bar{C}_{d,k}^\mathsf{T} - C_{d,k}S_k - S_k^\mathsf{T}C_{d,k}^\mathsf{T}\right) M_k^\mathsf{T}\right)^{-1}, \tag{25c}$$

$$\hat{x}[t_k] = \hat{x}[t_k|t_k - 1] + \sum_{i=1}^{n_m} \ell_{i,k} \left(m_{i,k} - c_i\hat{x}[t_k - d_{i,k}|t_k - 1]\right) \alpha_{i,k}, \tag{25d}$$

$$P[t_k] = \left(I - L_k M_k C_{d,k}\right) P[t_k|t_{k-1}] + L_k M_k S_k^\mathsf{T}, \tag{25e}$$

that are applied when any sensor is available at $t = t_k$, being $\ell_{i,k}$ the i-th column of matrix $L_k M_k$ and the matrices $C_{d,k}$ and $\bar{C}_{d,k}$ (of size $n_m \times n\bar{d}$) defined by its rows

$$[C_{d,k}]_{\{i,:\}} = c_i A^{-d_{i,k}},$$
$$[\bar{C}_{d,k}]_{\{i,:\}} = c_i \left[A^{-d_{i,k}} \ A^{-d_{i,k}+1} \ \cdots \ A^{-1} \ 0 \ \cdots \ 0\right].$$

Note that these rows are null if there is no delay in the sensor measurement. The delayed state estimation is computed as

$$\hat{x}[t_k - d_{i,k}|t_k - 1] = A^{-d_{i,k}}\hat{x}[t_k|t_k - 1] - \sum_{l=1}^{d_{i,k}} A^{-d_{i,k}+l-1}Bu[t_k - l]. \tag{26}$$

S_k is the matrix given by

$$S_k = \bar{A}\bar{W}\bar{C}_{d,k}^\mathsf{T},$$

with $\bar{A}$

$$\bar{A} = \left[I \ A \ A^2 \ \cdots \ A^{\bar{d}-1}\right].$$

4.1. Extended state implementation

In order to avoid matrix inversion in the implementation of the state estimator, an extended state based implementation is now addressed. The idea is to store the delayed state estimations and to updated them when a new set of measurements arrives, i.e., to avoid calculating $\hat{x}[t_k - d_{i,k}|t_k - 1]$ with (26). On the other hand, there should be enough memory to allocate A^{-j} $(j = 1, \ldots, \bar{d})$.

With this modification the resulting algorithm is implemented with the propagation equations

$$\hat{\mathbb{X}}[t|t-1] = \mathbb{A}\,\hat{\mathbb{X}}[t-1] + \mathbb{B}\,\boldsymbol{u}[t-1], \tag{27a}$$

$$\boldsymbol{P}[t|t-1] = \boldsymbol{A}\boldsymbol{P}[t-1]\boldsymbol{A}^{\mathsf{T}} + \boldsymbol{W}, \tag{27b}$$

plus the updating equations

$$\boldsymbol{L}_k = \left(\boldsymbol{P}[t_k|t_k-1]\boldsymbol{C}_{d,k}^{\mathsf{T}} - \boldsymbol{S}_k\right)\boldsymbol{M}_k^{\mathsf{T}}\cdot$$
$$\cdot\left(\boldsymbol{M}_k\left(\boldsymbol{C}_{d,k}\boldsymbol{P}[t_k|t_k-1]\boldsymbol{C}_{d,k}^{\mathsf{T}} + \boldsymbol{V} + \bar{\boldsymbol{C}}_{d,k}\bar{\boldsymbol{W}}\bar{\boldsymbol{C}}_{d,k}^{\mathsf{T}} - \boldsymbol{C}_{d,k}\boldsymbol{S}_k - \boldsymbol{S}_k^{\mathsf{T}}\boldsymbol{C}_{d,k}^{\mathsf{T}}\right)\boldsymbol{M}_k^{\mathsf{T}}\right)^{-1}, \tag{27c}$$

$$\hat{\mathbb{X}}[t_k] = \hat{\mathbb{X}}[t_k|t_k-1] + \Gamma\cdot\sum_{i=1}^{n_m}\boldsymbol{\ell}_{i,k}\left(m_{i,k} - \boldsymbol{c}_i\,\hat{\boldsymbol{x}}[t_k - d_{i,k}|t_k-1]\right)\alpha_{i,k} \tag{27d}$$

$$\boldsymbol{P}[t_k] = \left(\boldsymbol{I} - \boldsymbol{L}_k\boldsymbol{M}_k\boldsymbol{C}_{d,k}\right)\boldsymbol{P}[t_k|t_{k-1}] + \boldsymbol{L}_k\boldsymbol{M}_k\boldsymbol{S}_k^{\mathsf{T}} \tag{27e}$$

where $\hat{\mathbb{X}}[t-1] = \hat{\mathbb{X}}[t-1|t-1]$, and

$$\hat{\mathbb{X}}[t|t-1] = \begin{bmatrix} \hat{\boldsymbol{x}}[t|t-1] \\ \hat{\boldsymbol{x}}[t-1|t-1] \\ \vdots \\ \hat{\boldsymbol{x}}[t-\bar{d}|t-1] \end{bmatrix}; \; \mathbb{A} = \begin{bmatrix} \boldsymbol{A}\,\boldsymbol{0}\cdots\boldsymbol{0} \\ \boldsymbol{I} \quad\begin{matrix}\vdots\\\boldsymbol{0}\end{matrix} \end{bmatrix}; \; \mathbb{B} = \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{0} \\ \vdots \\ \boldsymbol{0} \end{bmatrix}; \; \Gamma = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{A}^{-1} \\ \vdots \\ \boldsymbol{A}^{-\bar{d}} \end{bmatrix}$$

being $\mathbb{A}$, $\mathbb{B}$ and Γ constant matrices of sizes $n(\bar{d}+1)\times n(\bar{d}+1)$, $n(\bar{d}+1)\times n_u$ and $n(\bar{d}+1)\times n$, respectively. With this notation, the delayed state estimation is obtained as

$$\hat{\boldsymbol{x}}[t_k - d_{i,k}|t_k-1] = [\,\overbrace{\boldsymbol{0}\,\cdots\,\boldsymbol{0}}^{n\cdot d_{i,k}}\,\boldsymbol{I}\,\boldsymbol{0}\,\cdots\,\boldsymbol{0}\,]\cdot\hat{\mathbb{X}}[t_k|t_k-1],$$

where no matrix inversion is needed to compute (26).

The propagation equation of the state is equivalent to the standard propagation equation of order n plus storing the old past state estimates (as in Moayedi et al. (2011)). As a difference with the approach in Moayedi et al. (2011), the size of the variance-covariance matrix is $n\times n$, instead of $\bar{d}n\times\bar{d}n$, therefore, the computing cost associated with the covariance matrix update is much lower. On the other hand, the augmented state update equation has a higher computational cost than the one in Moayedi et al. (2011), because here, all the past states are updated every time a measurement is received, while in Moayedi et al. (2011), the past states are simply stored, but only the current state is updated. Taking into account the sizes of all the matrices in the algorithm, it is easy to calculate the approximate number of products that must be computed every step. The resulting number of products is $\bar{d}(n^2+n^2n_m+n_m^2n)+3n^3+2n^2n_m+4n_m^2n+n^2+n_m^2+2n_mn$, that represents the same computer cost order $(o(\bar{d}))$ than the algorithm in Moayedi et al. (2011). The behaviour of both algorithms is very similar if the system is stable. However, for unstable systems, the performance of the proposed algorithm is much better than the algorithm described in Moayedi et al. (2011). The next examples show the comparative behaviour of both approaches.

5. Examples

5.1. *Example 1*

To evaluate the performance of the proposed algorithm, and to compare it with the proposal of Moayedi et al. (2011), the state estimation of three different SISO systems is addressed. Systems G_1 and G_2 are both stable, but with very different damping rate: 0.7 for G_1 and 0.05 for G_2. On the other hand, system G_3 has an integrator and thus, it is unstable.

$$G_1(s) = \frac{1}{(s+2)^2 + 2^2}$$

$$G_2(s) = \frac{1}{(s+0.1)^2 + 2^2}$$

$$G_3(s) = \frac{1}{s(s+1)}$$

For the simulation study the ZOH discrete-time models are obtained with sampling-time of 0.1 second, resulting in the following state-space models with $C = [0\ 1]$ in all cases.

$$\boldsymbol{A}_1 = \begin{bmatrix} 0.640 & -0.651 \\ 0.081 & 0.965 \end{bmatrix} \quad \boldsymbol{A}_2 = \begin{bmatrix} 0.960 & -0.394 \\ 0.098 & 0.980 \end{bmatrix} \quad \boldsymbol{A}_3 = \begin{bmatrix} 0.905 & 0 \\ 0.095 & 1.000 \end{bmatrix}$$

$$\boldsymbol{B}_1 = \begin{bmatrix} 0.0813 \\ 0.0044 \end{bmatrix} \quad \boldsymbol{B}_2 = \begin{bmatrix} 0.0983 \\ 0.0050 \end{bmatrix} \quad \boldsymbol{B}_3 = \begin{bmatrix} 0.0952 \\ 0.0048 \end{bmatrix}$$

A measurement noise with variance of $\boldsymbol{V} = 0.08$ and a process noise with covariance matrix

$$\boldsymbol{W} = \begin{bmatrix} 0.08 & 0.01 \\ 0.01 & 0.08 \end{bmatrix}$$

are assumed. With respect the sampling scenario, an average of 50% of packet dropout is considered, while a random delay between 0 and 5 samples is assumed for the available measurements. The previous measurement pattern, used in the simulations, is depicted in Figure 1. It must be pointed out that this sampling scenario includes the reception of time disordered measurements, and, therefore, the proposed approach is suboptimal, as explained in remark 1.

The state estimation error variances are presented in Table 1. These values are first compared with the full augmented state algorithm described in Gopalakrishnan et al. (2011), and secondly with the algorithm proposed in Moayedi et al. (2011). As can be noted, for system G_1 the results are very similar. For system G_2 the behaviour of the proposed algorithm is also similar to the full augmented state one, however the performance of the algorithm described in Moayedi et al. (2011) is clearly worse. For system G_3 the performance of the proposed algorithm is worse than the augmented state optimal, but it is still reasonable, while the algorithm in Moayedi et al. (2011) results in a very poor performance, as it is shown in Figure 2. This wrong behaviour for unstable systems is due to the fact that the older states of the extended state vector are
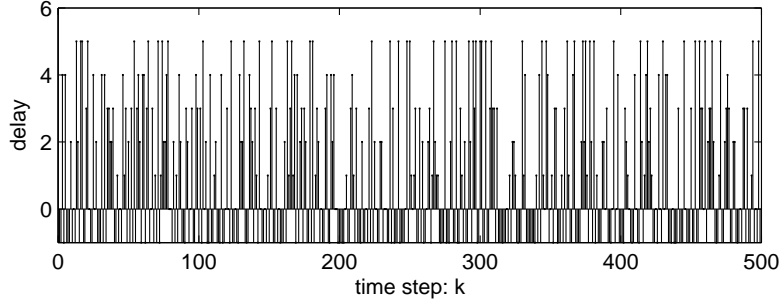
Figure 1. Measurement pattern used in the simulation study. Values below zero represent time instants when packets dropout occur

Algorithm	$\boldsymbol{G_1}$		$\boldsymbol{G_2}$		$\boldsymbol{G_3}$	
new proposal	[0.1000	0.0848]	[0.7571	0.2042]	[0.4205	1.2428]
augmented state	[0.1001	0.0848]	[0.7543	0.2036]	[0.4110	0.4977]
proposal in Moayedi et al. (2011)	[0.1013	0.0771]	[1.5566	0.3875]	[6.9800	229.1900]

Table 1. State estimation error variances ($[var(x_1 - \hat{x}_1) \quad var(x_2 - \hat{x}_2)]$)

not updated when a measurement is received. If the system is unstable, the estimation error of those older states increases rapidly, and hence, the corrections made with those wrong state estimates do not improve the estimation. Only when a measurement without delay is received, the estimation is improved significantly, as can be seen in Figure 2, where the estimation error (especially in state 2) only decreases in the instants with zero measurement delay.
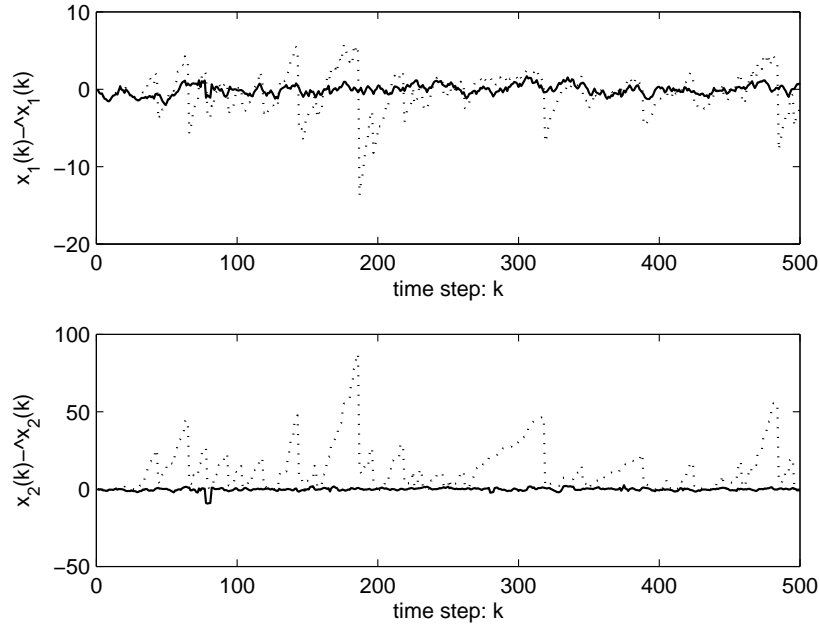

Figure 2. State estimation errors for system G_3. Solid line: new proposal; dashed line: proposal in Moayedi et al. (2011)

5.2. *Example 2*

The aim of this example is to illustrate the application of the proposed algorithm to systems with the features addressed in Remarks 2 to 5. Consider the following system:

$$\boldsymbol{x}_p[t+1] = \boldsymbol{A}_{p1}\boldsymbol{x}_p[t] + \boldsymbol{A}_{p2}\boldsymbol{x}_p[t-1] + \boldsymbol{B}_p(\boldsymbol{u}[t-2] + \boldsymbol{w}_1[t]) + \boldsymbol{w}_2[t],$$

$$\boldsymbol{y}_p[t] = \boldsymbol{C}_p\boldsymbol{x}_p[t].$$

where A_{p1}, A_{p2}, B_p and,C_p are known matrices, $\boldsymbol{w}_2[t]$ is a white noise signal with known covariance matrix $\boldsymbol{W}_2$. It is assumed that $\boldsymbol{w}_1[t]$ is a slow disturbance affecting the input that fulfils the equation

$$\boldsymbol{w}_1[t+1] = \boldsymbol{w}_1[t] + \Delta\boldsymbol{w}_1[t], \tag{28}$$

being $\Delta\boldsymbol{w}_1[t]$ a zero mean white noise signal with known covariance matrix $\boldsymbol{W}_1$. Furthermore, the input signal $\boldsymbol{u}$ is stored such that a new variable can be defined as

$$\boldsymbol{u}'[t] = \boldsymbol{u}[t-2] \tag{29}$$

in order to avoid extending the state to include input delays.

The output $\boldsymbol{y}$ is measured through a sensor modeled by the dynamic equation

$$\boldsymbol{x}_s[t+1] = \boldsymbol{A}_s[t]\boldsymbol{x}_s[t] + \boldsymbol{B}_s[t]\boldsymbol{y}_p[t] + \boldsymbol{w}_s[t] \tag{30}$$

$$m_{1,k} = \boldsymbol{c}_{s,1,k}\,\boldsymbol{x}_s[t_k - d_k] + \boldsymbol{d}_{y,1,k}\,\boldsymbol{y}[t_k - d_k] + v_{1,k}, \quad i = 1,\ldots,n_m. \tag{31}$$

where A_s, B_s, $\boldsymbol{c}_{s,1,k}$ and $\boldsymbol{d}_{y,1,k}$ are known matrices, and $\boldsymbol{w}_s[t]$ and $v_{1,k}$ are white noise signals with known covariance matrices $\boldsymbol{W}_s$ and $\mathbf{V}$.

Taking into account the Remarks 2, 3 and 4, an extended order model can be defined as

$$\begin{bmatrix} \boldsymbol{x}_p[t+1] \\ \boldsymbol{x}_p[t] \\ \boldsymbol{w}_1[t+1] \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_{p1} & \boldsymbol{A}_{p2} & \boldsymbol{B}_p \\ \boldsymbol{I} & 0 & 0 \\ 0 & 0 & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_p[t] \\ \boldsymbol{x}_p[t-1] \\ \boldsymbol{w}_1[t] \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_p \\ 0 \\ 0 \end{bmatrix} \boldsymbol{u}'[t] + \begin{bmatrix} \boldsymbol{I} & 0 \\ 0 & 0 \\ 0 & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_2[t] \\ \Delta\boldsymbol{w}_1[t] \end{bmatrix} \tag{32a}$$

$$\boldsymbol{y}[t] = \begin{bmatrix} \boldsymbol{C}_p & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_p[t] \\ \boldsymbol{x}_p[t-1] \\ \boldsymbol{w}_1[t] \end{bmatrix} \tag{32b}$$

Furthermore, according Remark 5, equations (32) and (30) can be gathered and expressed as

$$\underbrace{\begin{bmatrix} \boldsymbol{x}_p[t+1] \\ \boldsymbol{x}_p[t] \\ \boldsymbol{w}_1[t+1] \\ \boldsymbol{x}_s[t+1] \end{bmatrix}}_{\bar{\boldsymbol{x}}[t+1]} = \underbrace{\begin{bmatrix} \boldsymbol{A}_{p1} & \boldsymbol{A}_{p2} & \boldsymbol{B}_p & 0 \\ \boldsymbol{I} & 0 & 0 & 0 \\ 0 & 0 & \boldsymbol{I} & 0 \\ \boldsymbol{B}_s\boldsymbol{C}_p & 0 & 0 & \boldsymbol{A}_s \end{bmatrix}}_{\bar{\boldsymbol{A}}} \underbrace{\begin{bmatrix} \boldsymbol{x}_p[t] \\ \boldsymbol{x}_p[t-1] \\ \boldsymbol{w}_1[t] \\ \boldsymbol{x}_s[t] \end{bmatrix}}_{\bar{\boldsymbol{x}}[t]} + \underbrace{\begin{bmatrix} \boldsymbol{B}_p \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\bar{\boldsymbol{B}}} \boldsymbol{u}'[t] + \underbrace{\begin{bmatrix} \boldsymbol{I} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \boldsymbol{I} & 0 \\ 0 & 0 & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_2[t] \\ \Delta\boldsymbol{w}_1[t] \\ \boldsymbol{w}_s[t] \end{bmatrix}}_{\bar{\boldsymbol{w}}[t]} \tag{33a}$$

The proposed algorithm can now be applied replacing $\boldsymbol{A}$ and $\boldsymbol{x}$ by $\bar{\boldsymbol{A}}$ and $\bar{\boldsymbol{x}}$, respectively. The

gathered disturbance vector $\bar{\boldsymbol{w}}$ has the following covariance matrix

$$\bar{\boldsymbol{W}} = \begin{bmatrix} \boldsymbol{W}_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \boldsymbol{W}_1 & 0 \\ 0 & 0 & 0 & \boldsymbol{W}_s \end{bmatrix} \tag{34}$$

Figure 3 shows the results of an implementation of the proposed algorithm with the previous assumptions compared with the full augmented state algorithm described in Gopalakrishnan et al. (2011), that assumes white noise disturbances. The system is defined by the following matrices

$$A_{p1} = \begin{bmatrix} 0.3 & 0.5 \\ 0.4 & 0.1 \end{bmatrix}, \; A_{p2} = \begin{bmatrix} 0.1 & 0.3 \\ 0.4 & 0.2 \end{bmatrix}, \; B_p = \begin{bmatrix} 0.4 \\ 0.3 \end{bmatrix}, \; C_p = \begin{bmatrix} 0.3 & 0.2 \end{bmatrix}$$

$$A_s = \begin{bmatrix} 0.1 & 0.3 \\ 0.3 & 0.2 \end{bmatrix}, \; B_s = \begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix}, \boldsymbol{c}_{s,1,k} = \begin{bmatrix} 0.4 & 0.3 \end{bmatrix}, \; \boldsymbol{d}_{y,1,k} = 0.1$$

$$\boldsymbol{W}_1 = 0.08, \; \boldsymbol{W}_2 = \begin{bmatrix} 0.08 & 0.0 \\ 0.0 & 0.08 \end{bmatrix}, \; \boldsymbol{W}_s = \begin{bmatrix} 0.8 & 0.0 \\ 0.0 & 0.8 \end{bmatrix}, \; \mathbf{V} = 0.08$$

The sampling scenario is the same as in the example 1, depicted in Figure 1. It can be noted that the proposed approach leads to an unbiased estimation, while a considerable estimation error is observed for the augmented state algorithm due to the presence of the slow disturbance.
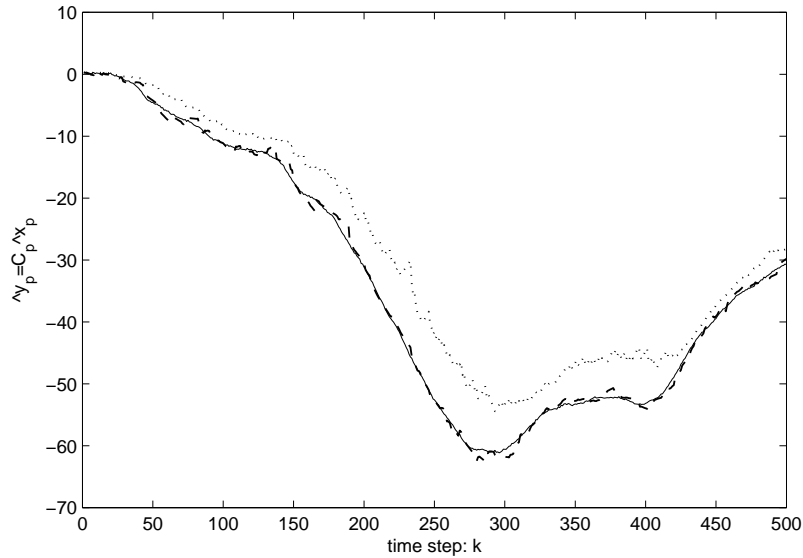

Figure 3. Output estimation: Solid line: actual system output; dashed line: new proposal; doted line: full augmented state algorithm

6.   Conclusions

The paper presents a state estimation algorithm for systems with scarce, irregular measurements, with time varying delays, that can be applied when the delay is only known at the instant when the measurement is received, and also admits out of sequence measurements (as can occur in networked control systems). The algorithm is based on a modification of the discrete Kalman filter, using negative-time measurement update, and is first developed for general time variant systems, and then particularized for time invariant systems.

The computational cost of the proposed estimator is lower than the full state augmentation Kalman filter schemes. This is mainly because in this case, a small (order n) covariance matrix is calculated and stored, but an augmented state of size $\bar{d}n$ is used. The updating gain is also of order n, but the full augmented state vector is updated every time a measurement is received. The approach is suboptimal, but the loss of performance is low compared to the optimal filter, and the computational cost is much lower. The proposed algorithm can be applied in any sampling situation: multisensor sets with different sampling rate, time-varying sampling, random availability of measurements, different time-varying delays in each sensor and out-of-sequence measurements. Compared with the recently proposed approach in Moayedi et al. (2011), the computational cost is similar, but the performance is better, especially if the system is unstable, as it has been shown in the comparative examples developed.

An interesting future research line could be to solve the inverse problem, i.e. to decide when to measure the different outputs to reach a given performance, minimizing the resources (number of measurements or network bandwidth, for example). This could be formulated as an off line scheduling problem, or as an on line algorithm that decides which are the measurements that should be taken at each instant. The extension of the algorithm to non linear systems can also be a future research line.

References

Basin, M., and Martinez-Zuñiga, R. (2004), "Optimal linear filtering over observations with multiple delays," *International Journal of Robust and Nonlinear Control*, 14, 685–696.

Basin, M., Rodriguez-Gonzalez, J., and Martinez-Zuñiga, R. (2005), "Optimal Filtering for Linear State Delay Systems," *IEEE Transactions on Automatic Control*, 50(5), 684–690.

Gopalakrishnan, A., Kaisare, N.S., and Narasimhan, S. (2011), "Incorporating delayed and infrequent measurements in Extended Kalman Filter based nonlinear state estimation," *Journal of Process Control*, 21, 119–129.

Gupta, V., Chung, T., Hassibi, B., and Murray, R. (2005), "On a stochastic algorithm for sensor scheduling," *16th IFAC world congress*.

Hsiao, F.H., and Pan, S.T. (1996), "Robust Kalman filter synthesis for uncertain multiple time-delay stochastic systems," *Journal of Dynamic Systems, Measurement, and Control*.

Julier, S., and Uhlmann, J. (2005), "Fusion of Time Delayed Measurements With Uncertain Time Delays," *Proceedings of the American Control Conference*.

Kalman, R. (1960), "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME. Journal of Basic Engineering*, 82, 35–45.

Larsen, T.D., Andersen, N.A., Ravn, O., and Poulsen, N.K. (1998), "Incorporation of time delayed measurements in a discrete-time Kalman filter," *In Proceeding ofIEEE 37th Conference of Decision and Control*, pp. 3972–3977.

Li, W., Shah, S., and Xiao, D. (2005), "Kalman filters for non-uniformly sampled multirate systems," *16th IFAC world congress*.

Lu, X., Zhang, H., Wang, W., and Teo, K.L. (2005), "Kalman filtering for multiple time-delay systems," *Automatica*, 41, 1455–1461.

Moayedi, M., Foo, Y., and Soh, Y. (2011), "Filtering for networked control systems with single/multiple measurement packets subject to multiple-step measurement delays and multiple packet dropouts," *International Journal of Systems Science*, 42(3), 335–348.

Sanchis, R., and Albertos, P. (2002), "Virtual sensors: model based output prediction," *First EurAsia Conference on Advances in Information and Communication Technology*.

Wang, H., Zhang, H., and Duan, G. (2005), "State estimation schemes for descriptor systems with multi-time delayed measurements," *16th IFAC world congress*.

Zhang, H., Basin, M., and Skliar, M. (2007), "Ito-Volterra optimal state estimation with continuous, multirate, randomly sampled, and delayed measurements," *IEEE Transactions on Automatic Control*, (3), 401–416.

Zhang, H., Basin, M., and Skliar, M. (2004), "Optimal state estimation with continuous, multirate and randomly sampled measurements," *Proceedings of the American Control Conference*.