

Multisensor Out-Of-Sequence Data Fusion for Estimating the State of Dynamic Systems

E. Besada-Portas, J.A. Lopez-Orozco, J.M de la Cruz

*Departamento de Arquitectura de Computadores y Automática, Universidad Complutense de Madrid
Av. Complutense s/n. Ciudad Universitaria. 28040 Madrid. Spain, {evabes, jalo, jmcruz}@dacya.ucm.es*

Abstract

Physically distributed sensors, communication networks, pre-processing algorithms and other reasons can delay or disorder the arrival of the information, provided by a set of sensors, to the fusion centre which is in charge of estimating the state of a system. In real-time control systems, the out-of-sequence data cannot delay the process of estimating the state of the system with the already received data and so the algorithms of the fusion centre need to handle this type of data properly.

In this paper we present two new algorithms for solving the out-of-sequence data problem for the case of linear and nonlinear dynamic control systems and compare them with other algorithms which exist in the literature. The algorithm for the linear case is equivalent to others but more general, while the nonlinear one is a new solution of the problem.

1. INTRODUCTION

The state of a complex control system is estimated by its fusion subsystem with the information provided by one or several sensors. The time and order of arrival of the data to the fusion centre depends on many factors including the positions of the sensors, the communication network used to send the information to the fusion centre, the time used to pre-process the measurements and extract its information useful for the fusion centre, and the priority of the different tasks of the complete system. The most difficult scenario occurs when the delays and the sequence of arrival of all the information are not fixed, constituting the named Out-Of-Sequence Problem (OOSP) [1].

In the case of sequential fusion algorithms, such as the Kalman filter (KF) and the Information filter (IF), there are three naïve solutions to solve the OOSP. The first consists on rejecting the delayed information. This is only valid when spurious measurements arrived delayed and the lost of information implies an increment of the uncertainty of the system state [2]. The second is waiting for all the information relevant to the current state before estimating it [3]. For real control systems this approach is not optimal when the delays are significant, because the control loop should be closed with the system state estimated with all the information available so far. And the third consists in storing the estimates of the state, the control signals, and the sensor data for all the time instants and whenever a new measurement arrives

restarts the sequential fusion process from the timestamp associated with it. This solution lets the fusion centre obtain the same estimate as when the same sensorial data is received without delays at the expenses of incrementing the computational cost and memory needs of the fusion algorithms. To minimize these problems, several algorithms have been developed in the last years for different types of linear ([1], [4]-[9]) and nonlinear ([10], [11]) dynamic systems.

In this paper two new algorithms are presented. The first one, IFAsyn (IF for Asynchronous data), can be used to estimate the state of a linear dynamic control system with additive gaussian noise when the data arrive delayed or out-of-sequence to the fusion centre, and it is equivalent to the IF (and so, to the KF) when the same data is provided to those algorithms without any delays. It is also equivalent to some of the algorithms presented in the literature but it can be used in more general situations. The second, EIFAsyn (Extended IF for Asynchronous data) is a novelty solution inside the Extended Information Filter (EIF) and Extended Kalman Filter (EKF) framework, and it is working successfully in a complex control system for autonomous mobile robot [12].

The paper is organized as follows. Section 2 presents the OOSP for linear control system, the algorithm developed to solve the problem, and a comparison of it with other existing algorithms. Section 3 shows the OOSP for a general control system with linear and non linear equations and the algorithm proposed to solve it. Finally, some of the conclusions are drawn in section 4.

2. ESTIMATING THE STATE OF LINEAR CONTROL SYSTEM WITH OUT-OF-SEQUENCE MEASUREMENTS

A. Problem Statement

The behaviour of a linear control system which has n sensors which are providing data related linearly with the state of the system is modelled by the following equations:

$$\mathbf{x}_t = \mathbf{F}_{t-1} \mathbf{x}_{t-1} + \mathbf{u}_{t-1} + \mathbf{v}_{t-1}$$

$$\mathbf{z}_{s,t} = \mathbf{H}_{s,t} \mathbf{x}_t + \mathbf{v}_{s,t} \quad \text{with } s = 1:n$$

where \mathbf{x}_t and $\mathbf{z}_{s,t}$ represent respectively the state of the system and the measurement of the s sensor at time t , \mathbf{u}_{t-1} the control signal applied during the time period from $t-1$ to t ,

\mathbf{F}_{t-1} and $\mathbf{H}_{s,t}$ the transition and measurement matrixes, and \mathbf{v}_{t-1} and $\mathbf{v}_{s,t}$ additive white noise with zero mean and covariances \mathbf{Q}_{t-1} and $\mathbf{R}_{s,t}$ respectively.

The objective of the fusion centre is to estimate the current system state and covariance ($\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}$) given the original state and covariance ($\mathbf{x}_0, \mathbf{P}_0$), the series of model parameters and control signals $\{\mathbf{F}_{k-1}, \mathbf{Q}_{k-1}, \mathbf{H}_{s,k}, \mathbf{R}_{s,k}, \mathbf{u}_{k-1} \mid s=1:n, k=1:t\}$, and all the data $\{\xi_{s,k,a} \mid s \in 1:n, k \in 1:t, a \leq t\}$ measured by sensor s at time k which has arrived at the fusion centre at time a , and which is already available ($a \leq t$).

When all the data arrive to the fusion centre without any delay ($a=k$), the current state estimate and covariance ($\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}$) can be obtained optimally using sequentially the prediction and assimilation steps of the KF, which only make use of the previous state estimate and covariance ($\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{P}_{t-1|t-1}$), the current time model parameters and control signals $\{\mathbf{F}_{t-1}, \mathbf{Q}_{t-1}, \mathbf{H}_{s,t}, \mathbf{R}_{s,t}, \mathbf{u}_{t-1} \mid s=1:n\}$, and all the current data $\{\xi_{s,t,t} \mid s \in 1:n\}$. An equivalent approach is the IF which operates in the information space instead of the state space used in KF, and so it estimates the information state and its covariance ($\hat{\mathbf{y}}_{t|t}, \mathbf{I}_{t|t}$). Each working space simplifies the operation of one of the two steps, what makes advantageous to make each step in the correct space (with the expression shown in Table 1), after doing the needed projections ($\hat{\mathbf{y}}_{j|t} = \mathbf{P}_{j|t}^{-1} \hat{\mathbf{x}}_{j|t}, \mathbf{I}_{j|t} = \mathbf{P}_{j|t}^{-1}$) to change of working space.

TABLE 1: KF PREDICTION (P) STEP AND IF ASSIMILATION (A) STEP

P	$\hat{\mathbf{x}}_{t t-1} = \mathbf{F}_{t-1} \hat{\mathbf{x}}_{t-1 t-1} + \mathbf{u}_{t-1}, \mathbf{P}_{t t-1} = \mathbf{F}_{t-1} \mathbf{P}_{t-1 t-1} \mathbf{F}_{t-1}^T + \mathbf{Q}_{t-1}$
A	$\mathbf{i}_{s,t} = \mathbf{H}_{s,t}^T \mathbf{R}_{s,t}^{-1} \zeta_{s,t,t}, \mathbf{I}_{s,t} = \mathbf{H}_{s,t}^T \mathbf{R}_{s,t}^{-1} \mathbf{H}_{s,t}$
	$\hat{\mathbf{y}}_{t t} = \hat{\mathbf{y}}_{t t-1} + \sum_{s=1}^S \mathbf{i}_{s,t}, \mathbf{Y}_{t t} = \mathbf{Y}_{t t-1} + \sum_{s=1}^S \mathbf{I}_{s,t}$

When some data arrive delayed to the fusion centre ($a > k$) the KF and IF can not incorporate it into the state of the system and so new algorithms need to be developed. Those algorithms will be optimal in the same sense as the KF and IF are, when the results of the new algorithms with out-of-sequence data will be the same as the result of the KF and IF with the same data without delays.

B. IFAsyn: An algorithm for the OOSP based on the IF

Our first algorithm is based on a junction tree (JT) algorithm [13], which is an exact inference algorithm for Bayesian Networks (BNs). We select it because it has two properties that make it a direct solution for our OOSP. On one hand, its behaviour is equivalent to the IF when only the operations associated with the prediction and assimilations steps are applied to the JT that models our system behaviour ([14], [15]). On the other, as an exact inference algorithm, the order in which the data is introduced in the nodes of the JT doesn't

change the final results. After a deep analysis of the operations performed by the selected algorithm in our system JT we determine the minimal subset of operations needed for solving our OOSP and how they are related with the IF [12].

The result of that analysis is IFAsyn, which is based on the separation (highlighted in Figure 1) that exists in the IF assimilation step between the information state that stores all the sensorial information previous to t ($\hat{\mathbf{y}}_{t|t-1}, \mathbf{Y}_{t|t-1}$) and the sensorial information (already projected onto the information space) of instant t ($\sum_{s=1}^S \mathbf{i}_{s,t}, \sum_{s=1}^S \mathbf{I}_{s,t}$).

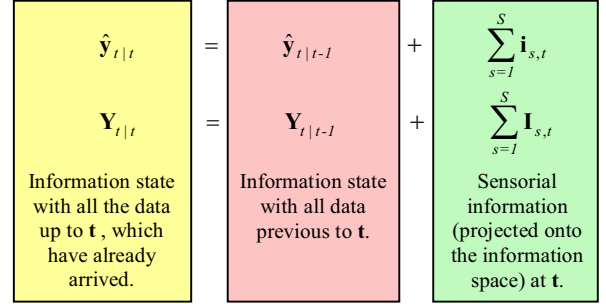


Fig. 1: Separation that exists in the assimilation step of the IF between the current sensorial information at t and all the previous one.

To exploit this separation, IFAsyn stores for each step time j five variables, two for the information state with all the data previous to j ($\hat{\mathbf{y}}_{j|j-1}, \mathbf{Y}_{j|j-1}$), two for accumulating the sensorial information of instant j already received by the fusion centre and already projected to the information space ($\mathbf{i}_j, \mathbf{I}_j$), and one for storing the control signal applied at that step time (\mathbf{u}_{j-1}). Additionally, it also stores for the current state t its estimated values ($\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}$). To minimize the memory requirements we could only store the variables that belong to a time window with the maximum expected delay.

With those five variables per time step, IFAsyn performs two steps as IF, with the expression presented in Table 2. In this table, we have included the projection operation needed to change the working space, and identified them between brackets after the symbol \perp . Additionally, inside the loop of the assimilation step (A), three other subgroups of operations (projecting/accumulating measurements (I), prediction (P) and assimilation (A)) are also identifiable.

The prediction step (P) is performed cyclically when the time from $t-1$ to t has elapsed. It has the same operations as the KF prediction step plus the initialization of variables $\mathbf{i}_j, \mathbf{I}_j, \hat{\mathbf{y}}_{j|j-1}$ and $\mathbf{Y}_{j|j-1}$.

The assimilation step (A) is carried out asynchronously, as soon as a new piece of data ($\zeta_{s,k,t}$) which has been measured by sensor s at time k is received by the fusion centre (whose current time is t). This step is more complex and consists on two different parts. In the first, the measurement is projected

into the information space $(\mathbf{i}_{s,k}, \mathbf{I}_{s,k})$, added to the variables which accumulate all the sensorial information already projected with the same timestamp k $(\mathbf{i}_k, \mathbf{I}_k)$, and assimilated to the information state up to time k $(\hat{\mathbf{y}}_{k|k}, \mathbf{Y}_{k|k})$. The second is a loop of: predictions (P) that re-propagate all the previous information (including the new piece of data) to the next time step j , and assimilations (A) of the just predicted information with the sensorial information with a bigger timestamp than k which was already projected onto the information space and accumulated. Finally, the current estimated system state is obtained via a final projection.

TABLE 2: IFASYN PREDICTION (P) AND ASSIMILATION (A) STEPS, WITH THE PROJECTIONS (\perp) BETWEEN THE STATE AND INFORMATION SPACES

P	$\hat{\mathbf{x}}_{t t-1} = \mathbf{F}_{t-1} \hat{\mathbf{x}}_{t-1 t-1} + \mathbf{u}_{t-1}, \quad \mathbf{P}_{t t-1} = \mathbf{F}_{t-1} \mathbf{P}_{t-1 t-1} \mathbf{F}_{t-1}^T + \mathbf{Q}_{t-1}$ $\perp (\hat{\mathbf{y}}_{t t-1} = \mathbf{P}_{t t-1}^{-1} \hat{\mathbf{x}}_{t t-1}, \quad \mathbf{Y}_{t t-1} = \mathbf{P}_{t t-1}^{-1})$ $\mathbf{i}_t = \mathbf{0}, \quad \mathbf{I}_t = \mathbf{0}$
A	$I [\mathbf{i}_{s,k} = \mathbf{H}_{s,k}^T \mathbf{R}_{s,k}^{-1} \zeta_{s,k,t}, \quad \mathbf{I}_{s,k} = \mathbf{H}_{s,k}^T \mathbf{R}_{s,k}^{-1} \mathbf{H}_{s,k}$ $\mathbf{i}_k = \mathbf{i}_k + \mathbf{i}_{s,k}, \quad \mathbf{I}_k = \mathbf{I}_k + \mathbf{I}_{s,k}]$ $A [\hat{\mathbf{y}}_{k k} = \hat{\mathbf{y}}_{k k-1} + \mathbf{i}_k, \quad \mathbf{Y}_{k k} = \mathbf{Y}_{k k-1} + \mathbf{I}_k]$ <p>for $j=k+1:t$</p> $\perp [\hat{\mathbf{x}}_{j-1 j-1} = \mathbf{Y}_{j-1 j-1}^{-1} \hat{\mathbf{y}}_{j-1 j-1}, \quad \mathbf{P}_{j-1 j-1} = \mathbf{Y}_{j-1 j-1}^{-1}]$ $P [\hat{\mathbf{x}}_{j j-1} = \mathbf{F}_{j-1} \hat{\mathbf{x}}_{j-1 j-1} + \mathbf{u}_{j-1}, \quad \mathbf{P}_{j j-1} = \mathbf{F}_{j-1} \mathbf{P}_{j-1 j-1} \mathbf{F}_{j-1}^T + \mathbf{Q}_{j-1}]$ $\perp [\hat{\mathbf{y}}_{j j-1} = \mathbf{P}_{j j-1}^{-1} \hat{\mathbf{x}}_{j j-1}, \quad \mathbf{Y}_{j j-1} = \mathbf{P}_{j j-1}^{-1}]$ $A [\hat{\mathbf{y}}_{j j} = \hat{\mathbf{y}}_{j j-1} + \mathbf{i}_j, \quad \mathbf{Y}_{j j} = \mathbf{Y}_{j j-1} + \mathbf{I}_j]$ <p>end</p> $\perp [\hat{\mathbf{x}}_{t t} = \mathbf{Y}_{t t}^{-1} \hat{\mathbf{y}}_{t t}, \quad \mathbf{P}_{t t} = \mathbf{Y}_{t t}^{-1}]$

The order of the operations of the assimilation step makes IFAsyn obtain the same results as IF and KF when the out-of-sequence measurements received by IFAsyn are provided to IF or KF without delays. So, IFAsyn is a generalization of those algorithms and an optimal solution for the OOSP.

C. IFAsyn vs. other algorithms for the OOSP

The third naïve solution proposed in the introduction restarts the fusion process from the timestamp k of the new measurement $\zeta_{s,k,t}$ received by the fusion centre. It needs to store for each step time j the measurements received by the algorithm up to time t $\{\zeta_{s,j,a} \mid s \in 1:n, a \leq t\}$, the information state $(\hat{\mathbf{y}}_{j|j-1}, \mathbf{Y}_{j|j-1})$ and the control signal (\mathbf{u}_{j-1}) . IFAsyn stores five variables $(\hat{\mathbf{y}}_{j|j-1}, \mathbf{Y}_{j|j-1}, \mathbf{i}_j, \mathbf{I}_j, \mathbf{u}_{j-1})$ for each time step, and so it reduces the memory needs of the naïve solution when the total space used to store the measurements $\{\zeta_{s,j,a} \mid s \in 1:n, a \leq t\}$ is bigger than the space needed to store the accumulated sensorial information already projected onto the information space $(\mathbf{i}_j, \mathbf{I}_j)$. The computational reduction in IFAsyn is due to the fact that the sensorial information has already been projected onto the information space and accumulated for all the measurements received, while the naïve approach will need to re-perform those tasks whenever new delayed data arrives to the fusion centre.

The comparison of IFAsyn with other recent algorithms for the OOSP ([1], [5]-[9]) is presented in Table 3, where the first column identifies each analysed algorithm by the paper where it is presented (and an identification name in the cases where there is more than one algorithm in the same paper). The second shows its underlying algorithm, the third a short description of the supporting idea used to develop it, the fourth if the algorithm makes the assumption of having a model with an invertible transition matrix F (assumption valid for the discrete system obtained discretizing a continuous system), the fifth if the formulation presented in the paper includes the control signal (and so if it can be used for control dynamic systems without being modified), the sixth if it is already designed for a system with multiple sensors providing measurements with different rates and delays, the seventh if the algorithms are equivalent to the IF and KF (when those algorithms receive the same data without delays), and the eighth the variables stored for a time window of size W .

TABLE 3: IFASYN VS. OTHER ALGORITHMS FOR THE OOSP.

Algorithm	Based on	Supporting idea	\mathbf{F}^{-1}	\mathbf{u}_t	N sensors	Optimum	Stored info (time window size =W)
IFAsyn	IF	Separation in IF assimilation step		√	√	√	$\hat{\mathbf{y}}_{j j-1}, \mathbf{Y}_{j j-1}, \mathbf{i}_j, \mathbf{I}_j$ and \mathbf{u}_j with $j=t-W:t$
[1]	KF	State retrodiction	√			√	Last W innovations or measurements
[6]	KF	State retrodiction	√				$\hat{\mathbf{x}}_{t t-r}$ and $\mathbf{P}_{t t-r}$ with $r=W+1:0$
[4]	KF	State retrodiction	√				K_j with $j=t-W:t$
[8]-AI	KF	Precalculating variables					W sets = $\{\hat{\mathbf{x}}_{j j}, \mathbf{P}_{j j}, \hat{\mathbf{x}}_{k j}, \mathbf{P}_{k j}, \mathbf{T}_j, \mathbf{W}_j\}$
[8]-AII	KF	Precalculating variables				√	W sets = $\{\hat{\mathbf{x}}_{j j}, \mathbf{P}_{j j}, \mathbf{M}_j^*, \tilde{\mathbf{M}}_j\}$
[5]	IF	Minimizing variables	√	√	√	√	$\hat{\mathbf{y}}_{j j}, \mathbf{Y}_{j j}$ and \mathbf{u}_j with $j=t-W:t$
[9]-AI	KF	Minimizing variables				√	$\hat{\mathbf{x}}_{j j}, \mathbf{P}_{j j}$ with $j=t-W:t$
[9]-AII	KF	Minimizing variables					$\mathbf{P}_{j j}$ with $j=t-W:t$
[7]	KF	Augmented state variables				√	Augmented X and augmented P

It is important to highlight the fact that the only algorithms which can already handle in an optimum way the data provided by multiple sensors with different delays are IFAsyn and [5]. They are also the only two algorithms that use IF as their supporting algorithm. Both properties are directed related by the fact that the IF assimilation step expressions are the same for the monosensorial ($n=1$) and multisensorial ($n>1$) case, while the KF assimilation step expressions are different for both cases and so generalising a monosensorial version of a KF-based algorithm for the multisensorial case with measurements with different delays can not be made straightforward.

The computational complexity of the different algorithm is not compared in the table, because the differences in their supporting ideas make that comparison difficult without implementing them. However, it is close to the algorithms of the minimising variables group and quite similar to the one presented in [5]. IFAsyn has a lower computational cost than [5] and doesn't assume that F is invertible, but it needs more memory. However, the use of the extra-memory gives IFAsyn another advantage over [5]: it makes it numerically more accurate.

3. ESTIMATING THE STATE OF GENERIC CONTROL SYSTEMS WITH OUT-OF-SEQUENCE MEASUREMENTS

A. Problem Statement

The behaviour of a generic control system which has n sensors providing data related with the state of the system is modelled by the following equations:

$$\begin{aligned} \mathbf{x}_t &= f(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}, t-1) + \mathbf{v}_{t-1} \\ \mathbf{z}_{s,t} &= h_s(\mathbf{x}_t, t) + \mathbf{v}_{s,t} \quad \text{with } s = 1 : n \end{aligned}$$

where \mathbf{x}_t , $\mathbf{z}_{s,t}$, \mathbf{u}_{t-1} , \mathbf{v}_{t-1} , $\mathbf{v}_{s,t}$, \mathbf{Q}_{t-1} and $\mathbf{R}_{s,t}$ have the same meaning as in the section 2.A, and $f(\cdot)$ and $h_s(\cdot)$ are the functions used to model the transition and the measurements. In the general case, each function can be either linear ($f(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}, t-1) = \mathbf{F}_{t-1}\mathbf{x}_{t-1} + \mathbf{u}_{t-1}$, $h_s(\mathbf{x}_t, t) = \mathbf{H}_{s,t}\mathbf{x}_t$) or non-linear, and so the problem can turn out being completely linear, completely non-linear, or a mixture of both cases.

The objective of the fusion centre is the same as in the linear case, with the unique difference that the sequence of model parameters and control signal is now defined by $\{f(\cdot), \mathbf{Q}_{t-1}, h_s(\cdot), \mathbf{R}_{s,k}, \mathbf{u}_{k-1}, | s = 1 : n, k = 1 : t\}$.

When there are no delays in the arrival of data ($\zeta_{s,k,a}$, $a=k$), the estimate of the system state and its covariance ($\hat{\mathbf{x}}_{t|t}$, $\mathbf{P}_{t|t}$) can be obtained approximately by the EKF and EIF (suboptimal extensions of the KF and IF for the non-linear case) and other algorithms. EKF and EIF have the same two steps as their basic versions with new expressions that take into account the non linearities of the system, and which are presented, in their simpler working space, in Table 4.

TABLE 4: EKF PREDICTION (P) STEP AND EIF ASSIMILATION (A) STEP

P	$\begin{aligned} \mathbf{F}_{t-1} &= (\mathbf{J}f)_{\mathbf{x}}(\hat{\mathbf{x}}_{t-1 t-1}, \mathbf{u}_{t-1}, t-1) \\ \hat{\mathbf{x}}_{t t-1} &= f(\hat{\mathbf{x}}_{t-1 t-1}, \mathbf{u}_{t-1}, t-1) \\ \mathbf{P}_{t t-1} &= \mathbf{F}_{t-1}\mathbf{P}_{t-1 t-1}\mathbf{F}_{t-1}^T + \mathbf{Q}_{t-1} \end{aligned}$
A	$\begin{aligned} \mathbf{H}_{s,t} &= (\mathbf{J}h_s)_{\mathbf{x}}(\hat{\mathbf{x}}_{t t-1}, t) \\ \zeta_{s,t}^C &= \zeta_{s,t} + (\mathbf{H}_{s,t}\hat{\mathbf{x}}_{t t-1} - h_s(\hat{\mathbf{x}}_{t t-1}, t)) \\ \mathbf{i}_{s,t} &= \mathbf{H}_{s,t}^T \mathbf{R}_{s,t}^{-1} \zeta_{s,t}^C, \quad \mathbf{I}_{s,t} = \mathbf{H}_{s,t}^T \mathbf{R}_{s,t}^{-1} \mathbf{H}_{s,t} \\ \hat{\mathbf{y}}_{t t} &= \hat{\mathbf{y}}_{t t-1} + \sum_{s=1}^S \mathbf{i}_{s,t}, \quad \mathbf{Y}_{t t} = \mathbf{Y}_{t t-1} + \sum_{s=1}^S \mathbf{I}_{s,t} \end{aligned}$

When some data $\zeta_{s,k,a}$ arrive delayed to the fusion centre ($a>k$) the EKF and EIF can not incorporate it into the state of the system and so new algorithms need to be developed. Building them over the EKF and EIF will make them already approximated solutions to the OOSP general problem.

B. EIFAsyn: an algorithm for OOSP based on the EIF

EIFAsyn is developed over IFAsyn taking into account the differences which exist between the expressions of Table 1 and Table 4, for each of the basic steps.

The expressions of the prediction step of the KF and EKF are not significantly different: the main change is the new operation of the EKF that obtains the transition matrix \mathbf{F}_{t-1} . More over, in IFAsyn the KF prediction operations are performed from scratch (all the expression are used) whenever they are needed in the prediction (P) and assimilation (A.P) steps. So, substituting in IFAsyn the KF prediction operations by the EKF prediction ones will make directly the new algorithm suitable for general systems with a non-linear transition model and linear measurement ones.

The discrepancies of the assimilation expressions of the IF and EIF are more important. On one hand, the number of operations is not only incremented to obtain the measurement matrix $\mathbf{H}_{s,t}$ but also to correct the measurement ($\zeta_{s,t,t}$) with the discrepancy that exists between the linearized measurement model ($\mathbf{H}_{s,t}\hat{\mathbf{x}}_{t|t-1}$) and the non-linear one ($h_s(\hat{\mathbf{x}}_{t|t-1}, t)$). The corrected measurement ($\zeta_{s,t,t}^C$) is then projected onto the information space and so, for the non-linear case, the projected measurement variables ($\mathbf{i}_{s,t}, \mathbf{I}_{s,t}$) depend on the value of $\hat{\mathbf{x}}_{t|t-1}$.

Due to that dependency, substituting the operations of the IF assimilation step by the EIF assimilation ones in IFAsyn and accumulating the projected measurement variables ($\mathbf{i}_{s,k}, \mathbf{I}_{s,k}$) in the total projected measurement ones ($\mathbf{i}_k, \mathbf{I}_k$) will return an approximate solution to the OOSP, different to the one obtained by EKF or EIF when they use the same data without delays. The reason for that is that in the loop of the association (A) step of IFAsyn there is a prediction (P) which updates the values of $\hat{\mathbf{x}}_{j|j-1}$ in each iteration, what would

modify the values of $\mathbf{i}_{s,j}$ and $\mathbf{I}_{s,j}$ for the nonlinear sensors if those values were recalculated inside the loop instead of being accumulated first, outside the loop, in the total projected measurement ones ($\mathbf{i}_j, \mathbf{I}_j$).

To obtain the same solution as the EKF and EIF when the data is provided to those algorithms without delays the measurements ($\zeta_{s,j,a}$) of the non-linear sensors of the systems need to be stored and corrected ($\zeta_{s,j,a}^C$), projected onto the information space ($\mathbf{i}_{s,j}, \mathbf{I}_{s,j}$) and assimilated directly to the variables $\hat{\mathbf{y}}_{j|j}$ and $\mathbf{Y}_{j|j}$ inside the loop.

TABLE 5: EIFASYN PREDICTION (P) AND ASSIMILATION (A) STEPS, WITH THE PROJECTIONS (\perp) BETWEEN THE STATE AND INFORMATION SPACES

P	$\mathbf{F}_{t t-1} = (\mathbf{Jf})_x(\hat{\mathbf{x}}_{t-1 t-1}, \mathbf{u}_{t-1}, t-1)$ $\hat{\mathbf{x}}_{t t-1} = f(\hat{\mathbf{x}}_{t-1 t-1}, \mathbf{u}_{t-1}, t-1)$ $\mathbf{P}_{t t-1} = \mathbf{F}_{t t-1} \mathbf{P}_{t-1 t-1} \mathbf{F}_{t t-1}^T + \mathbf{Q}_{t-1}$ $\perp [\hat{\mathbf{y}}_{t t-1} = \mathbf{P}_{t t-1}^{-1} \hat{\mathbf{x}}_{t t-1}, \mathbf{Y}_{t t-1} = \mathbf{P}_{t t-1}^{-1}]$ $\mathbf{i}_t = \mathbf{0}, \mathbf{I}_t = \mathbf{0}, \mathbf{i}_t^{rec} = \mathbf{0}, \mathbf{I}_t^{rec} = \mathbf{0}$
A	$\perp [\hat{\mathbf{x}}_{k k-1} = \mathbf{Y}_{k k-1}^{-1} \hat{\mathbf{y}}_{k k-1}]$ $I [\mathbf{H}_{s,k} = (\mathbf{Jh}_s)_x(\hat{\mathbf{x}}_{k k-1}, t)$ $\zeta_{s,k,t}^C = \zeta_{s,k,t} + (\mathbf{H}_{s,k} \hat{\mathbf{x}}_{k k-1} - h_s(\hat{\mathbf{x}}_{k k-1}, k))$ $\mathbf{i}_{s,k} = \mathbf{H}_{s,k}^T \mathbf{R}_{s,k}^{-1} \zeta_{s,k,t}^C, \mathbf{I}_{s,k} = \mathbf{H}_{s,k}^T \mathbf{R}_{s,k}^{-1} \mathbf{H}_{s,k}$ $\text{if } (s \notin \mathbf{rec}) \quad \mathbf{i}_k = \mathbf{i}_k + \mathbf{i}_{s,k}, \mathbf{I}_k = \mathbf{I}_k + \mathbf{I}_{s,k}$ $\text{else } \mathbf{i}_k^{rec} = \mathbf{i}_k^{rec} + \mathbf{i}_{s,k}, \mathbf{I}_k^{rec} = \mathbf{I}_k^{rec} + \mathbf{I}_{s,k}, \text{ store } \zeta_{s,k,t}]$ $A [\hat{\mathbf{y}}_{k k} = \hat{\mathbf{y}}_{k k-1} + \mathbf{i}_k + \mathbf{i}_k^{rec}, \mathbf{Y}_{k k} = \mathbf{Y}_{k k-1} + \mathbf{I}_k + \mathbf{I}_k^{rec}]$ <p>for $j=k+1:t$</p> $\perp [\hat{\mathbf{x}}_{j-1 j-1} = \mathbf{Y}_{j-1 j-1}^{-1} \hat{\mathbf{y}}_{j-1 j-1}, \mathbf{P}_{j-1 j-1} = \mathbf{Y}_{j-1 j-1}^{-1}]$ $P [\mathbf{F}_{j-1} = (\mathbf{Jf})_x(\hat{\mathbf{x}}_{j-1 j-1}, \mathbf{u}_{j-1}, j-1)$ $\hat{\mathbf{x}}_{j j-1} = f(\hat{\mathbf{x}}_{j-1 j-1}, \mathbf{u}_{j-1}, j-1)$ $\mathbf{P}_{j j-1} = \mathbf{F}_{j-1} \mathbf{P}_{j-1 j-1} \mathbf{F}_{j-1}^T + \mathbf{Q}_{j-1}]$ $\perp [\hat{\mathbf{y}}_{j j-1} = \mathbf{P}_{j j-1}^{-1} \hat{\mathbf{x}}_{j j-1}, \mathbf{Y}_{j j-1} = \mathbf{P}_{j j-1}^{-1}]$ $I [\mathbf{i}_j^{rec} = \mathbf{0}, \mathbf{I}_j^{rec} = \mathbf{0}]$ <p>for $s \in \mathbf{rec}$</p> $\mathbf{H}_{s,j} = (\mathbf{Jh}_s)_x(\hat{\mathbf{x}}_{j j-1}, j)$ $\zeta_{s,j,a}^C = \zeta_{s,j,a} + (\mathbf{H}_{s,j} \hat{\mathbf{x}}_{j j-1} - h_s(\hat{\mathbf{x}}_{j j-1}, k))$ $\mathbf{i}_j^{rec} = \mathbf{i}_j^{rec} + \mathbf{H}_{s,j}^T \mathbf{R}_{s,j}^{-1} \zeta_{s,j,a}^C, \mathbf{I}_j^{rec} = \mathbf{I}_j^{rec} + \mathbf{H}_{s,j}^T \mathbf{R}_{s,j}^{-1} \mathbf{H}_{s,j}$ $\text{end}]$ $A [\hat{\mathbf{y}}_{j j} = \hat{\mathbf{y}}_{j j-1} + \mathbf{i}_j + \mathbf{i}_j^{rec}, \mathbf{Y}_{j j} = \mathbf{Y}_{j j-1} + \mathbf{I}_j + \mathbf{I}_j^{rec}]$ <p>end</p> $\perp [\hat{\mathbf{x}}_{t t} = \mathbf{Y}_{t t}^{-1} \hat{\mathbf{y}}_{t t}, \mathbf{P}_{t t} = \mathbf{Y}_{t t}^{-1}]$

However, if the changes of the value of $\hat{\mathbf{x}}_{j|j-1}$ that occur during the prediction (P) part of the assimilation (A) step of

IFAsyn doesn't modify significantly the values of $\zeta_{s,j,a}^C$ and $\mathbf{H}_{s,j}$ for the non-linear sensor measurements whose values were already assimilated, recalculating all those variables will increase the computational load of the fusion centre without making it obtain a significantly different solution. More over, the EKF and EIF are already approximated solutions to the problem and so pursuing the objective of obtaining exactly the values that they get with the same data without delays will not make our algorithm an optimal solution to the OOSP. Nevertheless, there can be cases where that values need to be recalculated, and so our algorithm supports that option.

With all these observations, we have developed EIFAsyn, whose expressions are presented in Table 5, for the general OOSP. It uses the variable **rec** to define the set of non-linear sensors whose values need to be recalculated. Additionally to the storage variables of IFAsyn ($\hat{\mathbf{y}}_{j|j-1}, \mathbf{Y}_{j|j-1}, \mathbf{i}_j, \mathbf{I}_j, \mathbf{u}_{j-1}$), EIFAsyn must store the measurements $\zeta_{s,k,a}$ of the non-linear sensors belonging to **rec**. The variables \mathbf{i}_j and \mathbf{I}_j are used to accumulate the projected information of the linear sensors and of the non-linear ones whose projected measurements don't need to be recalculated, while the variables \mathbf{i}_j^{rec} and \mathbf{I}_j^{rec} , which are also stored in the algorithm presented in Table 5, store the values of the sensors whose projected measurements will be recalculated. The use of \mathbf{i}_j^{rec} and \mathbf{I}_j^{rec} is optional, but it makes the version of EIFAsyn presented in the table not to have to recalculate the projected measurement for the timestamp k of the measurement $\zeta_{s,k,t}$ which has triggered the assimilation step. For the linear sensors of the system, which obviously don't belong to **rec**, it is not necessary to calculate $(\mathbf{Jh}_s)_x(\cdot)$ or $\zeta_{s,k,t}^C$, and when the system has a linear transition model $(\mathbf{Jf})_x(\cdot)$ is not used.

As the number of non-linear sensors of the system whose projected measurements need to be recalculated increases, EIFAsyn gets closer to the third naïve solution presented in the introduction for OOSP, being equivalent solutions when all the sensors of the system are non-linear and need to have their projected measurements recalculated.

EIFAsyn is not directly comparable with other solutions for non-linear systems presented in the literature ([10]-[11]), because those algorithms are not based in the EIF or EKF.

It is important to highlight that the performance EIFAsyn strongly depends on which non-linear sensors need to recalculate their values when older measurements of any sensor arrive at the fusion centre, and so this decision must be taken with care after analysing (via the expressions or with simulations) how small changes in $\hat{\mathbf{x}}_{t|t-1}$ affect to the values of $\zeta_{s,t,t}^C$ and $\mathbf{H}_{s,j}$.

An example of how that decision influences in the result of the algorithm is presented in Table 6 and Figure 2. The table

shows the equations of the system and sensors. The figure shows the discrepancies obtained for two executions of EIFAsyn when the non-linear sensors belonged to **rec** and when they don't. They are due to the strong non-linearities of S3, which were introduced after series of simulations with other S3 with weaker non-linearities where the discrepancies were not significant.

TABLE 6: NON LINEAR SYSTEM (T) WITH 3 SENSORS (S1, S2 AND S3)

T (NL)	$\begin{pmatrix} x(t+1) \\ y(t+1) \\ \theta(t+1) \end{pmatrix} = \begin{pmatrix} x(t) + \frac{u_R(t) + u_L(t)}{2} \cos\left(\theta(t) + \frac{u_R(t) - u_L(t)}{b}\right) \\ y(t) + \frac{u_R(t) + u_L(t)}{2} \text{sen}\left(\theta(t) + \frac{u_R(t) - u_L(t)}{b}\right) \\ \theta(t) + \frac{u_R(t) - u_L(t)}{b} \end{pmatrix}$
S1 (L)	$z_{1\theta}(t+1) = \theta(t+1)$
S2 (NL)	$z_{2x}(t+1) = \sin(x(t+1)) \quad z_{2y}(t+1) = \sin(y(t+1))$ $z_{2\theta}(t+1) = \theta(t+1)$
S3 (NL)	$z_{3d}(t+1) = \sin^3(2x(t+1)) + \sin^3(2y(t+1))$

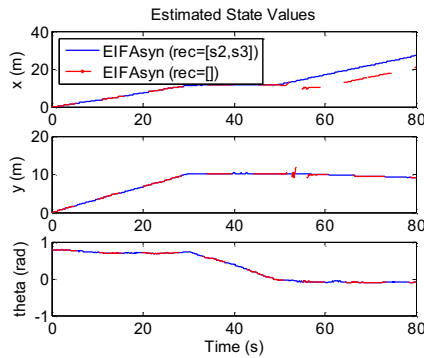


Fig. 2: Estimated state of the system modelled in Table 5 by EIFAsyn considering that the non-linear sensors belong to **rec** or don't belong to it.

4. CONCLUSIONS

We have presented two new algorithms for the OOSP, IFAsyn for pure linear systems and EIFAsyn for systems with linear and-or non-linear expressions.

IFAsyn is equivalent to other algorithms found in the literature, although an analysis of Table 3 shows that it is more general for dynamic control systems. On one hand it doesn't make any assumption about the invertibility of F. On the other, its formulation already includes the control signal (which the majority doesn't have) and the possibility of having multiple sensors providing data with different delays.

EIFAsyn is a new solution to the problem for non-linear systems, which is equivalent to the EIF when all the non-linear sensors of the system recalculate the projections of their measurements and an approximation to the EIF when some of them don't do it.

REFERENCES

- [1] Y. Bar-Shalom, "Update with out of Sequence Measurements in Tracking: Exact Solution", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 3, pp 769-778. Feb. 2002.
- [2] B. Sinopoli *et. al*, "Kalman Filtering with Intermittent Observations", *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp 1453-1464. Sep. 2004.
- [3] J.A. Lopez-Orozco, J.M. de la Cruz, E. Besada, P. Rupiezed, "An Asynchronous Robust and Distributed Multisensor Fusion System for Mobile Robot", *The International Journal of Robotics Research*, vol. 19, no. 10, pp 914-932. Oct. 2000.
- [4] M. Mallick, S. Coraluppi, C. Carthel, "Advances in Asynchronous and Decentralized Estimation", in *Proc. of IEEE Aerospace Conference*. pp 1873-1888, Mar. 2001.
- [5] E.W. Nettleton, H. Durrant-Whyte, "Delayed and Asequent Data in Decentralized Sensing Networks", in *Proc. of SPIE Conf #4571*. pp 1-9, 2001.
- [6] Y. Bar-Shalom, M. Mallick, H. Chen, R. Washburn, "One Step Solution for the General Out of Sequence Measurement Problem in Tracking", in *Proc. of the IEEE Aerospace Conference*. pp 1-9. 2002.
- [7] S. Challa, R. Evans, X. Wang. "A Bayesian Solution and its Approximation to Out-of-Sequence Measurement Problems", *Information Fusion 4*. pp 185- 199. Elsevier, 2003.
- [8] K. Zhang, X. Li, Y. Chen, "Optimal Update with Out of Sequence Measurements for Distributed Filtering", in *Proc. of 5th International Conference of Information Fusion (ISIF2002)*. pp 1519- 1526. 2002.
- [9] K. Zhang, X. Li, Y. Chen, "Multi-sensor Multi-Tracking with Out-of-Sequence Measurements" in *Proc. of ISIF 2003*. pp. 672-679. 2003.
- [10] M. Orton, A. Marrs, "A Bayesian Approach to Multi-target Tracking and Data Fusion with Out-of-Sequence Measurements", in *Proc. of IEE International Workshop on Target Tracking Algorithms and Applications*, pp v1-v15. 2001.
- [11] P. Mookerjee, F. Reifler, "Application of Reduced State Estimation to Multisensor Fusion with Out-Of-Sequence Measurements", in *Proc. of IEEE Radar Conference*. pp 111-116. 2004.
- [12] E. Besada-Portas, "Fusion Multisensorial de Medidas Temporalmente Desordenadas. Aplicación a Robots Autónomos Móbiles", *phD Dissertation, Universidad Complutense de Madrid*, 2004.
- [13] S.L. Lauritzen, "Propagation of Probabilities, Means and Variances in Mixed Graphical Models", *Journal of the American Statistical Association*. pp 1098-1108. 1992.
- [14] K.P. Murphy, "Filtering, Smoothing and the Junction Tree Algorithm", *Technical Report, Universidad de California at Berkeley, Department of CS*, 1998.
- [15] E. Besada-Portas, J.A. Lopez-Orozco, J.M de la Cruz, "Unified Fusion System Based on BN for Autonomous Mobile Robots", in *Proc. of 5th International Conference of Information Fusion*. 873-880. 2002.