

# Target Tracking Using Delayed Measurements with Implicit Constraints

Umut Orguner

Department of Electrical Engineering  
Linköping University  
Linköping, 581 83, Sweden  
Email: umut@isy.liu.se

Fredrik Gustafsson

Department of Electrical Engineering  
Linköping University  
Linköping, 581 83, Sweden  
Email: fredrik@isy.liu.se

**Abstract**—In target tracking, standard sensors as radar and vision observe the target with a negligible delay, since the speed of light is much larger than the speed of the target. This contribution studies the case where the ratio of the target and the propagation speed is not negligible, as is the case in sensor networks with microphones, geophones or sonars for instance, where the speed of air, ground waves and water cause a state dependent and stochastic delay of the observations. The proposed approach utilizes a temporary augmentation of the state vector with the observation delay, and sampling based approaches are suggested for modifying standard filters as the extended Kalman filter, unscented Kalman filters and possibly particle filters. An application example with bearing only tracking using a single microphone array is used as an illustration.

**Keywords:** Delayed measurements, state estimation, target tracking, implicit constraints, deterministic sampling, EKF, UKF, unscented transform.

## I. INTRODUCTION

Sensors used in target tracking all observe emitted (passive sensors) or reflected (active sensors) energy from the target. The classical standard sensor is the active radar, and common examples of passive sensors include vision (EOR, IR) and radar warning systems. The observation delay in these sensors is negligible since the speed of light is much larger than the speed of the target. One further example relating to sensor networks concerns positioning and tracking in wireless networks as mobile phone systems.

However, one trend in sensor networks is to use standard low cost sensors as microphones and geophones on land, and sonar in water. The assumption of negligible target speed compared to the speed of the media cannot always be made here. Consider for instance an array of sensors that by coherent signal processing techniques gives a bearing measurement to a previous position of the target. This time delay depends on the distance between the sensor and the target at that unknown time instant. This gives a further dimension to target tracking, where both the target state and the observation delays are unknown. With many (arrays of) sensors in the network, the observations are completely asynchronized over space.

We here make a simplified introduction to the problem whose solution is investigated in this work. The general and more formal problem formulation is given in Section II. Consider a case where we have perfect knowledge of the target

state vector  $x_{t_{k-1}}$  at time  $t_{k-1}$ , and that position  $p_{t_{k-1}}$  is a subset of the state vector. One sensor gets an observation related to  $x_{t_k - \Delta_k}$  at time  $t_k$  of for instance range, bearing or transversal speed (Doppler measurement). The delay  $\Delta_k$  can be described as a function of the position (and hence the state) of the target at time  $t_k - \Delta_k$  using the physics rules of signal propagation in the medium as

$$\Delta_k = d(x_{t_k - \Delta_k}) \quad (1)$$

On the other hand, using an assumed or known target dynamics, we can obtain a prediction of  $x_{t_k - \Delta_k}$  from perfectly known  $x_{t_{k-1}}$  as

$$x_{t_k - \Delta_k} = f_{t_k - \Delta_k, t_{k-1}}(x_{t_{k-1}}) \quad (2)$$

Now, (1) and (2) together define an implicit equation for  $\Delta_k$  whose solution can be derived with iterative techniques. This type of implicit and in general nonlinear constraints and their inclusion into the estimation process are examined in this paper. In this simplified scenario given above, we have neglected three sources of uncertainty:

- The initial state  $x_{t_{k-1}}$  is random and in a filtering context represented with a prior distribution.
- The solution of the implicit equation depends on the process noise which was not included in the simplified description (2). This introduces an uncertainty in the delay estimate.
- The propagation time itself through the simplified description (1) might be uncertain due to possible reasons such as the uncertain position of the sensor.

The proposed approach in this work, which can cover all the uncertainties mentioned above as well, is based on including the delay in the state vector temporarily while processing the observation taken at time  $t_k$ . The augmented state is then predicted using the implicit delay constraint which is followed by a measurement update. Then, using a non-standard marginalization step, the posterior distribution for  $[x_{t_k - \Delta_k}^T, \Delta_k]^T$  is marginalized and predicted to the closest unambiguous time instant.

Consideration of implicit delay constraints in the estimation process makes this work highly related to the area of constrained state estimation. The existing solutions which can

handle equality constraints in the estimation cycles consist of two main approaches. One and possibly the more popular of these is the use of constraints as the measurements where the constraint equation is considered as a (possibly noisy) information source about the state [1]–[3]. A conceptually similar method which uses the so called three block Kalman filter of [4] is given in [5]. The second approach of handling constraints is the projection type methods in which the unconstrained estimates are projected onto the constraint surface (manifold) at the end of each estimation cycle [6]. Another more useful form of projection based method has been proposed recently in [7]. In [8], a comparison of these two types of algorithms is presented along with some cases which yield equivalent results. This study, although being closer to projection type approaches, is different from these existing methodology in two aspects. The first one is that implicit constraints can be handled with our methodology whereas the existing ones are more suitable for explicit constraint. The second and the more important aspect that differentiates our study from the existing ones is that we consider the inclusion of the constraints as early as possible in the prediction update of the estimation process. This alleviates the performance reducing effects of non-constrained prediction. In order to observe this, we present an application of the idea of considering the constraints as measurement to our problem as an additional algorithm in Section IV. We make the comparison of these two methods in Section V.

## II. PROBLEM FORMULATION

We consider the following discrete-time nonlinear state space model defined on a probability space  $(\Omega, \mathcal{F}, P)$

$$x_{t_{k+1}} = f_{t_{k+1}, t_k}(x_{t_k}) + w_{t_{k+1}, t_k} \quad (3)$$

where  $\{x_{t_k} \in \mathbb{R}^n\}$  is the state sequence with initial distribution  $x_{t_0} \sim p_0(x_{t_0})$ . We here adopt an implicit simplified notation such that the system state dynamics given by (3) is a discretized version of a corresponding continuous time dynamics

$$\dot{x}_t = f(x_t) + w_t. \quad (4)$$

In (3),  $t_k \in \mathbb{R}$  is an arbitrary time value and  $f_{t_{k+1}, t_k}(\cdot)$  is the state transition function transforming  $x_{t_k}$  to  $x_{t_{k+1}}$  according to continuous time dynamics  $f(\cdot)$ . We also assume that the time sequence  $\{t_k\}_{k=0}^{\infty}$  is non-decreasing and therefore the transformation involved in (3) is not necessarily invertible.  $\{w_{t_{k+1}, t_k} \in \mathbb{R}^n\}$  is a white process noise sequence with distribution  $w_{t_k, t_{k-1}} \sim p_{t_k, t_{k-1}}^w(\cdot)$ . Here it is important to emphasize that  $w_{t_{k+1}, t_k}$  models the lumped effects of a continuous independent increment process noise  $w_t$  between the time instants  $t_k$  and  $t_{k+1}$ . The discrete delayed measurements  $\{y_k \triangleq y(t_k) \in \mathbb{R}^m\}$  of this system are given as

$$y_k = h_k(x_{t_k - \Delta_k}) + v_k \quad (5)$$

where  $\Delta_k$  is the amount of delay in the measurement  $y_k$  and  $\{v_k \in \mathbb{R}^m\}$  is a white measurement noise sequence independent from the process noise with distribution  $v_k \sim p_k^v(\cdot)$ . We here assume that we have a knowledge about the time delay  $\Delta_k$  in the form of an implicit equation (which we call as  $c_k$ ) as follows

$$c_k : \quad \Delta_k = d_k(x_{t_k - \Delta_k}, t_k) + \tau_k \quad (6)$$

where  $d_k(\cdot, \cdot)$  is in general a nonlinear function of the delayed state value  $x_{t_k - \Delta_k}$  and time  $t_k$  and  $\{\tau_k \in \mathbb{R}\}$  is a white noise sequence independent from the process and measurement noise with distribution  $\tau_k \sim p_k^\tau(\cdot)$ . Our main motivation for selecting such an expression for the time delay sequence  $\Delta_k$  is the case of a passive sound sensor whose delay expression is given as

$$\Delta_k = \frac{\|p(t_k - \Delta_k) - p_{t_k}^s\|}{v_s} + \tau_k \quad (7)$$

where  $p(t_k - \Delta_k)$  is the position of the target at time  $t_k - \Delta_k$  (which is a function of the delayed state  $x_{t_k - \Delta_k}$  and hence the form of (6).),  $p_{t_k}^s$  is the position of the sensor at time  $t_k$  and  $v_s$  is the speed of sound. The noise term  $\tau_k$  then represents unpredictable effects in the transmission of the sound (pressure) wave in the environmental conditions or the possible uncertainty in the position of the sensor. In this work, we consider each constraint  $c_k$  of (6) as a piece of information to include into the estimation process and we show the cumulative information of constraints up to and including time  $n$  as  $c_{0:n} \triangleq \{c_k\}_{k=0}^n$ . Although the constraints themselves are not random variables, in the following, we are going to use them as the arguments of the probability density functions in given conditions and Bayes rules and these must be interpreted information-wise. Note that this type of notation is unconventional in the literature and here adopted for ease of notation. Throughout the study, we assume that all measurement acquisition times  $\{t_k\}_{k=0}^{\infty}$  and auxiliary variables (like sensor positions etc.) used in the constraint evaluation (of course, other than state component  $x_{t_k - \Delta_k}$  in (6)) are known. Therefore, the constraints  $c_{0:\infty}$  are known beforehand but their availability to the estimators is limited for the purpose of recursive estimation. Our aim is to find a possibly approximate solution for the posterior density  $p(x_{t_k} | y_{0:k}, c_{0:k})$ . This problem can be approached in a Bayesian framework. The Bayesian density recursion for  $p(x_{t_k} | y_{0:k}, c_{0:k})$  is given as

$$p(x_{t_k} | y_{0:k}, c_{0:k}) = \frac{p(y_k | x_{t_k}, y_{0:k-1}, c_{0:k})}{p(y_k | y_{0:k-1}, c_{0:k})} p(x_{t_k} | y_{0:k-1}, c_{0:k}) \quad (8)$$

Here, an important problem is the calculation/approximation  $p(y_k | x_{t_k}, y_{0:k-1}, c_{0:k})$  which arises due to the fact that the observation  $t_k$  measures the delayed state  $x_{t_k - \Delta_k}$ . This necessitates one to have a considerable knowledge of  $\Delta_k$  before calculating an estimate for  $x_{t_k - \Delta_k}$  and hence for  $p(x_{t_k} | y_{0:k})$ . However, due to (6),  $\Delta_k$  and  $x_{t_k - \Delta_k}$  are implicitly constrained which makes it impossible to calculate/approximate one of

them without knowing the other. Therefore, one has to keep a joint density of the state  $x_{t_k}$  and the delay sequence  $\Delta_k$  together. We can write a similar equation to (8) for this case as follows

$$p(x_{t_k}, \Delta_k | y_{0:k}, c_{0:k}) = \frac{p(y_k | x_{t_k}, \Delta_k, y_{0:k-1}, c_{0:k})}{p(y_k | y_{0:k-1}, c_{0:k})} \times p(x_{t_k}, \Delta_k | y_{0:k-1}, c_{0:k}). \quad (9)$$

One can, however, get a more useful joint density considering

$$p(x_{t_k}, \Delta_k | y_{0:k}, c_{0:k}) = \int p(x_{t_k} | x_{t_k - \Delta_k}, \Delta_k, y_{0:k}, c_{0:k}) \times p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k}) dx_{t_k - \Delta_k} \quad (10)$$

Therefore, instead of obtaining a recursion for the joint density  $p(x_{t_k}, \Delta_k | y_{0:k}, c_{0:k})$ , one can find a recursion for  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k})$  and use (10) to obtain  $p(x_{t_k}, \Delta_k | y_{0:k}, c_{0:k})$ . The preference of the joint density  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k})$  must be intuitively clear because its arguments  $x_{t_k - \Delta_k}$  and  $\Delta_k$  appear in the delay constraint  $c_k$  directly.

**Problem Definition:** Given the state dynamics (3) and measurement relation (5), find a possibly approximate recursion for the joint density  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k})$  considering the implicit delay constraint (6).

### III. DETERMINISTIC SAMPLING BASED SOLUTION

We here examine the problem assuming that we are at an intermediate stage of an estimation process with an available representation of previous estimated density  $p(x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1} | y_{0:k-1}, c_{0:k-1})$ . The task is now to derive the current estimated density  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k})$  from the previous one using all possible available information in the measurement  $y_k$  and the constraint  $c_k$ . In this work, we always consider the case  $t_k - \Delta_k \geq t_{k-1} - \Delta_{k-1}$  and hence all the measurements  $\{y_k\}_{k=0}^{\infty}$  are in sequence measurements (i.e., not out of sequence measurements [9]–[11]). Using Bayes rule, we can write the probability density  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k})$  as

$$p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k}) = \frac{p(y_k | x_{t_k - \Delta_k})}{p(y_k | y_{0:k-1}, c_{0:k})} \times p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k-1}, c_{0:k}) \quad (11)$$

which shows that  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k})$  is the measurement updated version of the predicted density  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k-1}, c_{0:k})$ . Noting that the density  $p(y_k | x_{t_k - \Delta_k})$  is available and the corresponding measurement update can easily be approximated, the main problem reduces to calculating (or approximating) the prediction density  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k-1}, c_{0:k})$ . Therefore, the challenging part of the problem is to find  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k-1}, c_{0:k})$  from the previous estimated density  $p(x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1} | y_{0:k-1}, c_{0:k-1})$ . The main difficulty involved in this operation is that there is no directly available recursion connecting  $\Delta_k$  to  $\Delta_{k-1}$  other than the implicit relation  $c_k$  which connects them through the state sequence. Such a description is quite hard to obtain and

the resulting recursions if tractable are quite rough ones which could degrade prediction performance. There is, on the other hand, another method of including  $c_k$  in the estimation process by taking advantage of the implicit property of  $c_k$  which we describe in the following parts of this section. Deriving explicit recursions for  $\Delta_k$  from  $c_k$  (if possible) is investigated in Section IV which results in a heuristic method to compare to our main algorithm given below.

Suppose we have a probability density representation of  $p(x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1} | c_{0:k-1})$  in terms of  $N$  deterministically selected samples (called  $\sigma$ -points in the literature)  $\{x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1}\}_{i=1}^N$ . Then, we are going to find the density  $p(x_{t_k - \Delta_k}, \Delta_k | c_{0:k})$  which includes the constraint  $c_k$  using the modified versions  $\{\bar{x}_{t_k - \Delta_k}, \bar{\Delta}_k\}_{i=1}^N$  of the samples  $\{x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1}\}_{i=1}^N$  where

$$\bar{x}_{t_k - \Delta_k}^{(i)} = g_x(x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1}^{(i)}) \quad (12)$$

$$\bar{\Delta}_k^{(i)} = g_{\Delta}(x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1}^{(i)}). \quad (13)$$

The functions  $g_x(\cdot, \cdot)$  and  $g_{\Delta}(\cdot, \cdot)$  which must be dependent on the constraint relation  $c_k$  can be selected in many ways. We construct those functions implicitly using the constraint relation  $c_k$  itself as a recursion. Then, the transformed  $\sigma$ -point  $\{\bar{x}_{t_k - \Delta_k}, \bar{\Delta}_k\}$  becomes a limit point of this recursion initialized by the corresponding  $\sigma$ -point  $\{x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1}\}$ . This recursion which is the backbone of our study and our main assumption is presented in the following theorem.

*Theorem 1:* Let  $d_k(\cdot, \cdot)$  be a differentiable function of  $x$  and  $t$ . Consider the recursion

$$\Delta_k(m+1) = d_k(x_{t_k - \Delta_k(m)}, t_k) + \tau \quad (14)$$

which is initialized by  $\Delta_k(0) = d_k(\check{x}, t_k) + \tau$  where  $\tau$  is a scalar constant and  $x_{t_k - \Delta}$  is calculated using

$$x(t_k - \Delta_k) = f_{t_k - \Delta_k, t_{k-1} - \check{\Delta}}(\check{x}). \quad (15)$$

Here,  $\check{x}$  and  $\check{\Delta}$  are given state and delay values. The sequence  $\Delta_k(m)$  converges exponentially to a fixed point  $\Delta_k(\infty)$  satisfying

$$\Delta_k(\infty) = d_k(x_{t_k - \Delta_k(\infty)}, t_k) + \tau \quad (16)$$

if  $|\nabla_x d_k^T(x, t_k) f(x)| < 1$  for all  $x$ , where  $\nabla_x d_k(x, t_k) \triangleq \frac{\partial d_k}{\partial x}(x, t_k)$ .

*Proof:* Proof and discussion are skipped due to space considerations. ■

In the following parts of this section, we are going to propose a complete solution for the estimation problem described above using the result of Theorem 1 in its prediction step as a means of transforming the  $\sigma$ -points. The initial assumption to begin with is the Gaussianity of the densities  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k})$ .<sup>1</sup> Therefore, at each time step, our method holds a single mean and covariance value for

<sup>1</sup>Note, however, that the transformation idea can easily be extended to non-Gaussian densities represented with particles in a sequential Monte-Carlo filtering framework.

the description of the probability density. Now, defining the augmented vector  $\xi_k \triangleq [x_{t_k - \Delta_k}^T \quad \Delta_k]^T$ , we can write

$$p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k}) = \mathcal{N}(\xi_k; \hat{\xi}_{k|k,k}, \Xi_{k|k,k}) \quad (17)$$

where the notation  $\mathcal{N}(x; \bar{x}, \Sigma)$  stands for a Gaussian probability density function which has a mean  $\bar{x}$  and covariance  $\Sigma$  evaluated at the dummy variable  $x$ . In (17), we define the quantities  $\hat{\xi}_{k|k,k}, \Xi_{k|k,k}$  in a partitioned way given as

$$\hat{\xi}_{k|k,k} = \begin{bmatrix} \hat{\xi}_{k|k,k}^1 \\ \hat{\xi}_{k|k,k}^2 \end{bmatrix} \triangleq \begin{bmatrix} \hat{x}_{t_k - \Delta_k | k, k} \\ \hat{\Delta}_{k|k,k} \end{bmatrix} \quad (18)$$

$$\Xi_{k|k} = \begin{bmatrix} \Xi_{k|k,k}^{11} & \Xi_{k|k,k}^{12} \\ \Xi_{k|k,k}^{21} & \Xi_{k|k,k}^{22} \end{bmatrix} \triangleq \begin{bmatrix} P_{k|k,k}^{xx} & P_{k|k,k}^{x\Delta} \\ P_{k|k,k}^{\Delta x} & P_{k|k,k}^{\Delta\Delta} \end{bmatrix} \quad (19)$$

which we are going to refer to interchangeably. The method we propose will therefore at each step update  $\hat{\xi}_{k-1|k-1,k-1}$  and  $\Xi_{k-1|k-1,k-1}$  to obtain  $\hat{\xi}_{k|k,k}$  and  $\Xi_{k|k,k}$ . The structure of the method can be decomposed into three parts as follows:

- **Prediction Update:** Predict  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k-1}, c_{0:k})$  from  $p(x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1} | y_{0:k-1}, c_{0:k-1})$  using  $\sigma$ -points and Theorem 1. In other words, find  $\hat{\xi}_{k|k-1,k}$  and  $\Xi_{k|k-1,k}$  from the transformed  $\sigma$ -points generated from  $\hat{\xi}_{k-1|k-1,k-1}$  and  $\Xi_{k-1|k-1,k-1}$ .
- **Measurement Update:** Update the predicted density  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k-1}, c_{0:k})$  with the measurement  $y_k$ . This part is simply the implementation of (11). If the measurement is a linear function of  $\xi_k$ , then the update can be done using Kalman filter measurement update optimally. If nonlinearity is involved, suboptimal approaches like extended Kalman filter or unscented Kalman filter measurement updates can be applied.
- **Final Estimate Calculation:** This third part is the calculation of the final estimate  $\hat{x}_{t_k|k}$  from the density  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k}, c_{0:k})$  using the system dynamics

$$x_{t_k} = f_{t_k, t_k - \Delta_k}(x_{t_k - \Delta_k}) + w_{t_k, t_k - \Delta_k}. \quad (20)$$

This is the implementation of the marginalized version of the integral (10). Even if the original system dynamics  $f(\cdot)$  is linear, the overall transformation from  $\xi_k$  to  $x_{t_k}$  becomes nonlinear due to the delay term. Suboptimal approaches like EKF or UKF transformation strategies can be used.

We examine all three parts in different subsections in the following.

### A. Prediction Update

This part makes a prediction update while adding the next constraint  $c_k$  into the prediction using transformed  $\sigma$ -points. Suppose that we are given a mean  $\psi$  and a covariance  $\Psi$  of size  $n_\psi$  and  $n_\psi \times n_\psi$  respectively. Then the generation of  $2n_\psi + 1$   $\sigma$ -points  $\{\psi^{(j)}\}_{j=0}^{2n_\psi}$  and their corresponding weights

$\{\alpha^{(j)}\}_{j=0}^{2n_\psi}$  is done as follows.

$$\psi^{(0)} = \psi \quad \alpha^{(0)} = \alpha^{(0)} \quad (21)$$

$$\psi^{(i)} = \psi + \gamma \left[ \sqrt{\frac{n_\psi}{1 - \alpha^{(0)}}} \Psi \right]_i \quad \alpha^{(i)} = \frac{1 - \alpha^{(0)}}{2n_\psi} \quad (22)$$

$$\psi^{(i+n_\epsilon)} = \psi - \gamma \left[ \sqrt{\frac{n_\psi}{1 - \alpha^{(0)}}} \Psi \right]_i \quad \alpha^{(i+n_\epsilon)} = \frac{1 - \alpha^{(0)}}{2n_\psi} \quad (23)$$

for  $i = 1, \dots, n_\epsilon$  where the notation  $[\cdot]_n$  and  $\sqrt{\cdot}$  denotes the  $n$ th column and positive semi-definite square root of a matrix respectively and  $\gamma \ll 1$  is a real positive number. This type of  $\sigma$ -point generation is usually applied in unscented Kalman filter [12], [13] and the resulting transformation is called as the *unscented transform* [14] when  $\gamma = 1$ . When each  $\sigma$ -point is transformed by a transformation  $g(\cdot)$  such that  $\bar{\psi}^{(i)} \triangleq g(\psi^{(i)})$ , the resulting mean and covariance of the transformed random variable  $\bar{\psi}$  are calculated as follows.

$$\bar{\psi} = \sum_{i=0}^{2n_\psi} \alpha^{(i)} \bar{\psi}^{(i)} \quad (24)$$

$$\bar{\Psi}_{k|k-1,k} = \alpha^{(0)} (\bar{\psi}^{(0)} - \hat{\psi})(\cdot)^T + \frac{1}{\gamma^2} \sum_{i=1}^{2n_\psi} \alpha^{(i)} (\bar{\psi}^{(i)} - \hat{\psi})(\cdot)^T. \quad (25)$$

Noting the multiplication factor of  $\frac{1}{\gamma^2}$  in front of the summation in (25), one can see that this type of covariance calculation is different from that of UKF. It can be shown that when  $\gamma$  goes to zero, this update mechanism converge to an EKF type update.<sup>2</sup>

The general  $\sigma$ -point based strategy described above is going to be used in different ways in the following depending on whether the random variable  $\tau_k$  in  $c_k$  (i.e., in (6)) is identically zero or not.

Suppose that  $p(x_{t_{k-1} - \Delta_{k-1}}, \Delta_{k-1} | y_{0:k-1}, c_{0:k-1})$  is Gaussian and we are given its mean  $\hat{\xi}_{k-1|k-1,k-1}$  and its covariance  $\Xi_{k-1|k-1,k-1}$ .

1) *Case  $\tau_k = 0$ :* In this case, one sets  $\psi = \hat{\xi}_{k-1|k-1,k-1}$  and  $\Psi = \Xi_{k-1|k-1,k-1}$  in the  $\sigma$ -point generation algorithm and then generates the  $\sigma$ -points. Note that each generated  $\sigma$ -point  $\psi^{(i)}$  can be partitioned as  $\xi_k$  given in (18). Then, for  $i = 0, \dots, 2n_\epsilon$ , one sets

$$\check{x} = \psi^{1,(i)} \quad \check{\Delta} = \psi^{2,(i)} \quad \tau = 0 \quad (26)$$

in Theorem 1 where  $\psi^{m,(i)}$  denotes the  $m$ th partition of the  $(i)$ th  $\sigma$ -point and obtains the corresponding  $\Delta_k(\infty)$  which we denote here as  $\Delta_k^{(i)}(\infty)$  using the recursion (14). We show an illustration of this prediction process in Figure 1. One can terminate the recursion (14) safely once  $|\Delta_k(m) - \Delta_k(m-1)| < \epsilon_{th}$  where  $\epsilon_{th}$  is a predefined threshold. Then the transformed  $\sigma$ -points  $\xi_{k|k-1,k}^{(i)}$  representing the predicted density  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k-1}, c_{0:k})$  are given as

$$\xi_{k|k-1,k}^{(i)} = \begin{bmatrix} x(t_k - \Delta_k^{(i)}(\infty)) \\ \Delta_k^{(i)}(\infty) \end{bmatrix} \quad (27)$$

<sup>2</sup>Proof of this is skipped for space considerations.



Note that although the measurement  $y_k$  is not a direct function of  $\Delta_k$ , the update (35) might still affect the final delay estimate  $\hat{\Delta}_k|k,k$  due to the correlation between  $\hat{\xi}_{k|k-1,k}^1$  and  $\hat{\xi}_{k|k-1,k}^2$  given by the cross covariance matrix  $\Xi_{k|k-1,k}^{12}$  (or  $P_{k|k-1,k}^{x\Delta}$ ).

### C. Final Estimate Calculation

In this part one has to derive a final estimate  $\hat{x}_{t_k|k}$  and covariance  $P_{t_k|k}$  from the measurement updated mean  $\hat{\xi}_{k|k,k}$  and the covariance  $\Xi_{k|k,k}$  using the relation (20) repeated for convenience below.

$$x_{t_k} = f_{t_k, t_k - \Delta_k}(x_{t_k - \Delta_k}) + w_{t_k, t_k - \Delta_k}. \quad (37)$$

The most basic approach to be used here is the EKF prediction update which suggests

$$\hat{x}_{t_k|k} = f_{t_k, t_k - \hat{\xi}_{k|k,k}^2}(\hat{\xi}_{k|k,k}^1) \quad (38)$$

$$P_{t_k|k} = F_{t_k, t_k - \hat{\xi}_{k|k,k}^2}(\hat{\xi}_{k|k,k}^1) \Xi_{k|k,k} F_{t_k, t_k - \hat{\xi}_{k|k,k}^2}^T(\hat{\xi}_{k|k,k}^1) + W_k$$

where  $F_{t_k, t_k - \hat{\xi}_{k|k,k}^2}(\hat{\xi}_{k|k,k}^1) \triangleq \frac{\partial f_{t_k, t_k - \hat{\xi}_{k|k,k}^2}(\hat{\xi}_{k|k,k}^1)}{\partial \xi_k} \Big|_{\xi_k = \hat{\xi}_{k|k,k}}$  and  $W_k = \text{Cov}(w_{t_k, t_k - \hat{\xi}_{k|k,k}^2})$ . One can also use a  $\sigma$ -point approach by defining the related vectors as

$$\psi = \begin{bmatrix} \hat{\xi}_{k|k,k} \\ \mathbf{0}_{n_x \times 1} \end{bmatrix} \quad \Psi = \text{blkdiag}(\Xi_{k|k,k}, W_k) \quad (39)$$

where  $W_k$  is defined the same as above. Then the transformation of the resulting  $\sigma$ -points by (37) and the calculation of their mean and covariance straightforwardly follows.

### IV. A DIFFERENT METHOD TO HANDLE THE IMPLICIT CONSTRAINTS

One can always propose the following heuristic model for the unknown variables

$$x_{t_k - \Delta_k} = f_{t_k - \Delta_k, t_{k-1} - \Delta_{k-1}}(x_{t_{k-1} - \Delta_{k-1}}) + w_{t_k, t_{k-1}} \quad (40)$$

$$\Delta_k = \Delta_{k-1} + \delta_{t_k, t_{k-1}}(x_{t_{k-1} - \Delta_{k-1}}) \quad (41)$$

which uses the implicit assumption that  $\Delta_k$  is close to  $\Delta_{k-1}$ . This model is still a non-standard one since in (40), both sides contain  $\Delta_k$ . However, one can design a two step prediction process described as follows.

- First, predict  $\Delta_k$  using (41) to obtain  $\hat{\Delta}_k|k-1$ .
- Predict  $x_{t_k - \Delta_k}$  by using  $f_{t_k - \hat{\Delta}_k|k-1, t_{k-1} - \Delta_{k-1}}$  instead of  $f_{t_k - \Delta_k, t_{k-1} - \Delta_{k-1}}$  in (40).

This type of prediction enables one to obtain an approximation to the density function  $p(x_{t_k - \Delta_k}, \Delta_k | y_{0:k-1}, c_{0:k-1})$ . Therefore, in the prediction step, the constraint information is still not added into the density although it is already known. Therefore one has to add this missing information in the measurement update by defining the augmented measurement  $\bar{y}_k$  as

$$\bar{y}_k = \begin{bmatrix} y_k \\ 0 \end{bmatrix} = \begin{bmatrix} h_k(x_{t_k - \Delta_k}) + v_k \\ d_k(x_{t_k - \Delta_k}, t_k) - \Delta_k + \tau_k \end{bmatrix}. \quad (42)$$

This measurement update can be done in a standard way utilizing the EKF or UKF approaches as proposed in Section III-B. The method also requires a final estimate calculation like the one given in Section III-C.

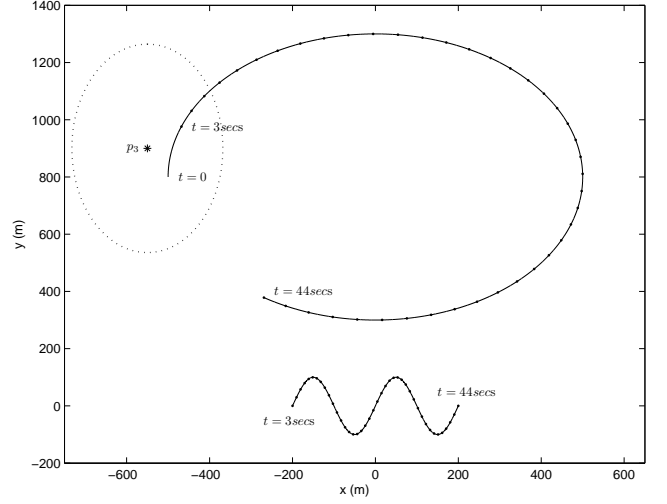


Figure 2. The target and sensor trajectory used in the example. The target and sensor positions at measurement times  $t_k = 3, 4, \dots, 44$ secs are emphasized with black dots. The initial position estimate  $p_3$  given to the filters is illustrated with an asterisk. The 99 percent confidence interval corresponding to the initial covariance is shown with the dotted ellipse.

### V. SIMULATION STUDY

In this section, the performance of our deterministic sampling based method is going to be compared on a simulated target tracking scenario with other possible approaches which include:

- Heuristic approach of Section IV.
- An EKF which totally neglects that there is a delay in sensing.
- An EKF which tries to compensate for the delay by using its state estimate.

We consider a two-dimensional bearing only tracking problem with a single maneuvering sensor. The single target in the scenario makes a clockwise coordinated turn of radius  $500m$  with a speed about  $200km/h$  beginning in  $y$ -direction with the initial position  $[-500m, 800m]$  for  $44$ secs. The tracking sensor called  $S_1$  acquires bearing data of the target corrupted by a Gaussian measurement noise with zero mean and standard deviation of  $0.001rads$  with sampling period  $T = 1sec$  beginning at  $t_0 = 3secs$ . The sensor, therefore, gets a total number of 42 measurements in the interval  $[3secs, 44secs]$ . The sensor trajectory is selected to lie on the curve  $y = 100 \sin(\frac{2\pi}{200}x)$  when  $x$  ranges in the interval  $[-200m, 200m]$  beginning at  $x = -200m$  at time  $t_0 = 3secs$  with constant  $x$ -speed.

The true target trajectory and the sensor positions used in the example are illustrated in Figure 2. The target motion is modeled in the filters with a discretized coordinated turn model with unknown constant turn rate (i.e., the turn rate is also a state variable) and with Cartesian velocity. Therefore, the state of the target is given as  $x_k = [p_k^x, p_k^y, v_k^x, v_k^y, \omega_k]^T$  where  $p, v$  and  $\omega$  variables denote the position, velocity and turn rate respectively. In all simulations, we selected the standard deviations for the turn rate and speed as  $\sigma_{\dot{\omega}} = 0.01rad/sec^2$  and  $\sigma_{\dot{v}} = 1m/sec^2$  respectively.

The delay expression used in the simulations is given by (7) with  $v_s \approx 344m/sec$  and the mean and the variance of  $\tau_k$  has been taken as 0, i.e.,  $\tau_k = 0$ .

Four different algorithms are tested with the following brief descriptions (and abbreviations).

- **EKF**: Standard EKF algorithm which uses the measurement equation

$$y_k = h_k(x_{t_k}) + v_k. \quad (43)$$

This algorithm does not take into account that there is a delay in sensing and ignores it completely. In other words it thinks that the measurement time is equal to the target time.

- **EKFD1**: Our deterministic sampling based method as explained in Section III. Except the prediction step which uses deterministic sampling with  $\gamma = 10^{-6}$ , all the other steps of the algorithm uses EKF type procedure as described in Section III. In the prediction step, the recursions of Theorem 1 are applied to  $\sigma$ -points with a stopping rule  $|\Delta_k^{(i)}(m) - \Delta_k^{(i)}(m-1)| < 10^{-10}secs$ . In order to apply the Theorem 1 with convergence, we need to satisfy the condition  $|\nabla_x d_k^T(x, t_k)f(x)| < 1$ . It is easy to show that this is satisfied if  $|\hat{v}_k| < v_s$  i.e., if the estimated target velocity is smaller than the speed of sound. In some cases where the filter diverges, this can be a serious problem. During the simulations, we observed that this happens only at the times where target position estimate crosses the sensor trajectory which results in divergent estimates due to very large innovations caused by the discontinuity of the arctangent function. At those rare cases, we just give a divergence decision and do not include the results into the Monte Carlo averages.

This algorithm takes care of both the constraint (7) and the dynamics of  $\Delta_k$  implied by (7) using the sampling based prediction step.

- **EKFD2**: The heuristic algorithm described in Section IV. All parts of the algorithm uses EKF type procedures. The standard deviation of the zero-mean noise  $\delta_{t_k, t_{k-1}}$  of (41) driving the random walk model of  $\Delta_k$  has been taken to be 0.5secs. Although taking care of the constraint, the algorithm neglects the dynamics of  $\Delta_k$  since it incorporates the constraint in the measurement update.
- **EKFD3**: This algorithm is similar to **EKF** but it knows that what it estimates is  $x_{t_k - \Delta_k}$ . Therefore, it calculates a delay estimate  $\hat{\Delta}_k$  from (7) and its estimate of  $x_{t_k - \Delta_k}$ . Then it extrapolates its estimate by  $\hat{\Delta}_k$  to form its output estimate. This is the most straightforward estimation algorithm used in practice but it ignores the dynamics of  $\Delta_k$  and therefore its estimate of  $x_{t_k - \Delta_k}$  is wrong which can cause unpredictable errors.

All the filters to be run have been initialized with the state value  $x_3 = [-550, 900, 10, 10, -0.1]^T$  and an initial covariance  $P_3 = \text{diag}(100^2, 200^2, 50^2, 50^2, 0.1^2)$ . The initial position estimate and covariance are illustrated in Figure 2 with an asterisk sign and a dotted curve respectively.

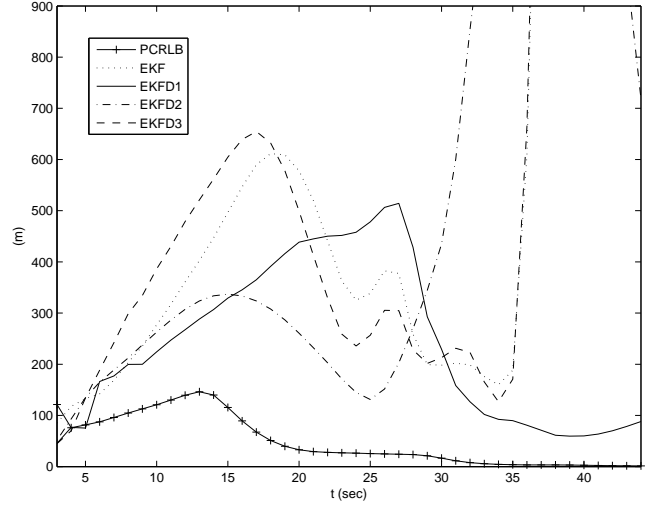


Figure 3. RMS-position errors of the algorithms with the parametric Cramer-Rao lower bound (PCRLB) calculated with known delays.

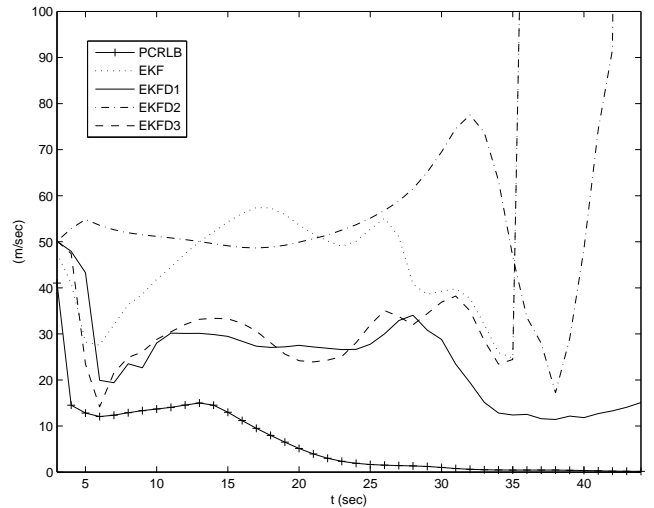


Figure 4. RMS-velocity errors of the algorithms with the parametric Cramer-Rao lower bound (PCRLB) calculated with known delays.

A total number of 1000 Monte-Carlo runs have been made by changing the realization of the measurement noise in each one. It has been detected that, in 13 of the MC runs, EKFD1 failed to apply Theorem 1 due to the fact that the magnitude of the estimated velocity exceeded the speed of sound. For a fair comparison, 13 of the runs showing the worst performance for other algorithms have also been excluded while calculating averages. RMS position and velocity errors of the algorithms are shown along with the corresponding parametric Cramer-Rao lower bounds (PCRLBs) (calculated with known delays) in Figures 3 and 4 respectively. It is immediately recognizable from the figures that some algorithms except for EKFD1 show divergent behavior in both curves. While it cannot be said that the position and velocity errors of EKFD1 are lower than the others all the time, it is evident that it is the only non-diverging algorithm in both position and velocity. This divergence of

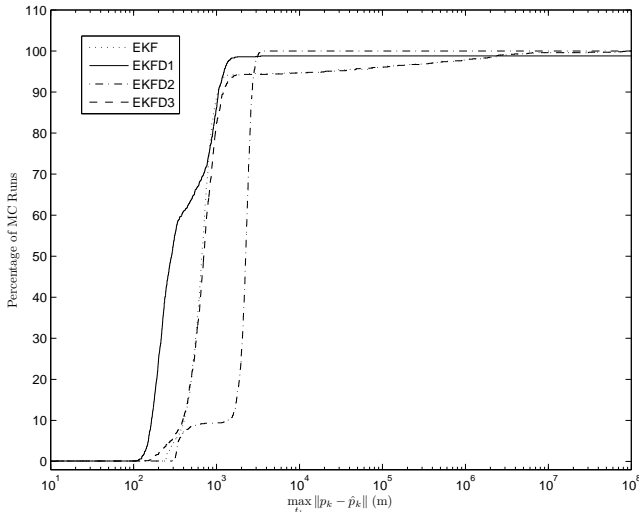


Figure 5. Percentage of MC runs in which the maximum position estimation errors (MPEE)  $\max_k \|p_k - \hat{p}_k\|$  of the algorithms is below a threshold. The threshold is changed between 10 to  $10^8$  and the percentage of MC runs (out of 1000) where each algorithm's MPEE is below the threshold is shown.

the other algorithms take place in the interval  $[25secs, 35secs]$  which is the time interval in which the true delay sequence  $\Delta_k$  changes most rapidly since the target begins to come closer to the sensor. Although the algorithm EKFD1 shows worse position estimation performance than the others in the interval  $[20secs, 30secs]$  its convergence when the target gets closer to the sensors is faster and as a result it gets the closest result to the PCRLB at the last sampling periods. Its velocity estimation performance, on the other hand, can be said to be the best among all of the algorithms.

As a final result, we compared the percentage of MC runs for which the maximum position estimation error (MPEE) of the algorithms are below a threshold in Figure 5. Here in the figure, we change the threshold from  $10m$  to  $10^8m$  and we find the percentage of MC runs (out of 1000) where each algorithm's MPEE is below that threshold. This figure can give an opinion on what performance was achieved in what percentage of the MC runs by the different algorithms. For example if an algorithm's curve rises earlier in the figure, then this means that in a high percentage of the MC runs, the algorithm got smaller position error values. When we look at the figure, we see that the curve for EKFD1 rises the earliest and in about 60% of the MC runs (i.e., in about 600 of the runs) its MPEE was below  $200m$ 's. On the other hand, the corresponding MPEE values for 60% of the runs for other algorithms are  $700m$ 's and  $2000m$ 's for EKF&EKFD3 and EKFD2 respectively. These clearly show the positive effects of not neglecting the dynamics of  $\Delta_k$  and taking care of the constraint as early as possible in the estimation cycle.

## VI. CONCLUSIONS

This paper has presented a new approach to handle non-uniform and dynamically changing delays in measurements

using a deterministic sampling based approach in the prediction step of EKF and UKF like filters. The presented transformation approach based on the implicit property of the delay constraint can be applied in particle filters as well. The simulation results show that the resulting filter is much more stable than some other applicable alternatives and obtains better estimation performance in most of the cases due to its following distinguishing properties:

- It does not neglect the dynamics of  $\Delta_k$  implied by the constraint.
- It includes the constraint information in the estimation cycle as early as possible in the prediction step.

## ACKNOWLEDGMENTS

The authors gratefully acknowledge fundings from SSF (Swedish Foundation for Strategic Research) Strategic Research Center MOVIII and the Vinnova/FMV TAIS project ARCUS.

## REFERENCES

- [1] M. Tahk and J. L. Speyer, "Target tracking problems subject to kinematic constraints," *IEEE Trans. Automat. Contr.*, vol. 35, no. 3, pp. 324–326, Mar. 1990.
- [2] A. T. Alouani and W. D. Blair, "Use of a kinematic constraint in tracking constant speed, maneuvering targets," *IEEE Trans. Automat. Contr.*, vol. 38, no. 7, pp. 1107–1111, Jul. 1993.
- [3] J. De Geeter, H. Van Brussel, J. De Schutter, and M. Decréton, "A smoothly constrained Kalman filter," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, no. 10, pp. 1171–1177, Oct. 1997.
- [4] R. Nikoukhah, A. S. Willsky, and B. C. Levy, "Kalman filtering and Riccati equations for descriptor systems," *IEEE Trans. Automat. Contr.*, vol. 37, no. 9, pp. 1325–1342, Sep. 1992.
- [5] L. S. Wang, Y. T. Chiang, and F. R. Chang, "Filtering method for nonlinear systems with constraints," *IEE Proceedings on Control Theory and Applications*, vol. 149, no. 6, pp. 525–531, Nov. 2002.
- [6] D. Simon and T. Chia, "Kalman filtering with state equality constraints," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 38, no. 1, pp. 2774–2784, Jan. 2002.
- [7] S. Julier and J. J. LaViola, Jr., "On Kalman filtering with nonlinear equality constraints," *IEEE Trans. Signal Processing*, vol. 55, no. 6, pp. 2774–2784, Jun. 2007.
- [8] G. Nachi, "Kalman filtering in the presence of state space equality constraints," in *Proceedings of Chinese Control Conference, 2007. CCC 2007*, Jun. 2007, pp. 107–113.
- [9] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: Wiley, 2001.
- [10] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Norwood, MA: Artech House, 1999.
- [11] Y. Bar-Shalom, "Update with out of sequence measurements in tracking: Exact solution," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 38, no. 3, pp. 769–778, Jul. 2002.
- [12] S. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.
- [13] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium. AS-SPCC*, Oct. 2000, pp. 153–158.
- [14] S. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Automat. Contr.*, vol. 45, no. 3, pp. 477–482, Mar. 2000.