Marginalized Particle Filters for Mixed Linear/Nonlinear State-space Models

Thomas Schön, Fredrik Gustafsson, *Member, IEEE,* and Per-Johan Nordlund

Abstract— The particle filter offers a general numerical tool to approximate the posterior density function for the state in nonlinear and non-Gaussian filtering problems. While the particle filter is fairly easy to implement and tune, its main drawback is that it is quite computer intensive, with the computational complexity increasing quickly with the state dimension. One remedy to this problem is to marginalize out the states appearing linearly in the dynamics. The result is that one Kalman filter is associated with each particle. The main contribution in this article is to derive the details for the marginalized particle filter for a general nonlinear state-space model. Several important special cases occurring in typical signal processing applications will also be discussed. The marginalized particle filter is applied to an integrated navigation system for aircraft. It is demonstrated that the complete high-dimensional system can be based on a particle filter using marginalization for all but three states. Excellent performance on real flight data is reported.

Index Terms— State estimation, Particle filter, Kalman filter, Marginalization, Navigation systems, Nonlinear systems.

I. INTRODUCTION

THE nonlinear non-Gaussian filtering problem considered here consists of recursively computing the posterior probability density function of the state vector in a general discrete-time state-space model, given the observed measurements. Such a general model can be formulated as

$$x_{t+1} = f(x_t, w_t), \qquad (1a)$$
$$y_t = h(x_t, e_t). \qquad (1b)$$

Here, y_t is the measurement at time t, x_t is the state variable, w_t is the process noise, e_t is the measurement noise, and f, h are two arbitrary nonlinear functions. The two noise densities p_{w_t} and p_{e_t} are independent and are assumed to be known.

The posterior density $p(x_t|Y_t)$, where $Y_t = \{y_i\}_{i=0}^t$, is given by the following general measurement recursion

$$p(x_t|Y_t) = \frac{p(y_t|x_t)p(x_t|Y_{t-1})}{p(y_t|Y_{t-1})}, \qquad (2a)$$

$$p(y_t|Y_{t-1}) = \int p(y_t|x_t)p(x_t|Y_{t-1})dx_t, \qquad (2b)$$

and the following time recursion

$$p(x_{t+1}|Y_t) = \int p(x_{t+1}|x_t)p(x_t|Y_t)dx_t, \qquad (2c)$$

T. Schön is with the Department of Electrical Engineering, Linköping University, Linköping, Sweden (e-mail: schon@isy.liu.se).
F. Gustafsson is with the Department of Electrical Engineering, Linköping University, Linköping, Sweden (e-mail: fredrik@isy.liu.se).
P-J. Nordlund is with the Department for Flight Data and Navigation, Saab Aerospace, Linköping, Sweden (e-mail: Per-Johan.Nordlund@saab.se).

initiated by $p(x_0|Y_{-1}) = p(x_0)$ [20]. For linear Gaussian models, the integrals can be solved analytically with a finite dimensional representation. This leads to the Kalman filter recursions, where the mean and the covariance matrix of the state are propagated [1]. More generally, no finite dimensional representation of the posterior density exists. Thus, several numerical approximations of the integrals (2) have been proposed. A recent important contribution is to use simulation based methods from mathematical statistics, sequential Monte Carlo methods, commonly referred to as particle filters [11], [12], [16].

Integrated navigation is used as a motivation and application example. Briefly, the integrated navigation system in the Swedish fighter aircraft Gripen consists of an inertial navigation system (INS), a terrain-aided positioning (TAP) system and an integration filter. This filter fuses the information from INS with the information from TAP. For a more thorough description of this system the reader is referred to [32], [33]. TAP is currently based on a point-mass filter as presented in [6], where it is also demonstrated that the performance is quite good, close to the Cramér-Rao lower bound. Field tests conducted by the Swedish air force have confirmed the good precision. Alternatives based on the extended Kalman filter have been investigated [5], but have been shown to be inferior particularly in the transient phase (the EKF requires the gradient of the terrain profile, which is unambiguous only very locally). The point-mass filter, as described in [6], is likely to be changed to a marginalized particle filter in the future for Gripen.

TAP and INS are the primary sensors. Secondary sensors (GPS and so on) are used only when available and reliable. The current terrain-aided positioning filter has three states (horizontal position and heading), while the integrated navigation system estimates the accelerometer and gyroscope errors, and some other states. The integration filter is currently based on a Kalman filter with 27 states, taking INS and TAP as primary input signals.

The Kalman filter which is used for integrated navigation requires Gaussian variables. However, TAP gives a multimodal un-symmetric distribution in the Kalman filter measurement equation and it has to be approximated with a Gaussian distribution before being used in the Kalman filter. This results in severe performance degradation in many cases, and is a common cause for filter divergence and system re-initialization.

The appealing new strategy is to merge the two state vectors into one, and solve integrated navigation and terrain-aided positioning in one filter. This filter should include all 27 states, which effectively would prevent application of the particle fil-

ter. However, the state equation is almost linear, and only three states enter the measurement equation nonlinearly, namely horizontal position and heading. Once linearization (and the use of EKF) is absolutely ruled out, marginalization would be the only way to overcome the computational complexity. More generally, as soon as there is a linear sub-structure available in the general model (1) this can be utilized in order to obtain better estimates and possibly reduce the computational demand. The basic idea is to partition the state vector as

$$x_t = \begin{bmatrix} x_t^l \\ x_t^n \end{bmatrix}, \tag{3}$$

where x_t^l denotes the state variable with conditionally linear dynamics and x_t^n denotes the nonlinear state variable [14], [32]. Using Bayes' theorem we can then marginalize out the linear state variables from (1) and estimate them using the Kalman filter [22], which is the optimal filter for this case. The nonlinear state variables are estimated using the particle filter. This technique is sometimes referred to as Rao-Blackwellization [14]. The idea has been around for quite some time, see e.g., [12], [7], [8], [2], [14], [31]. The contribution of this article is the derivation of the details for a general nonlinear state-space model with a linear sub-structure. Models of this type are common and important in engineering applications, e.g., positioning, target tracking and collision avoidance [18], [4]. The marginalized particle filter has been successfully used in several applications, for instance in aircraft navigation [32], underwater navigation [24], communications [9], [37], nonlinear system identification [28], [35], and audio source separation [3].

Section II explains the idea behind using marginalization in conjunction with general linear/nonlinear state-space models. Three nested models are used, in order to make the presentation easy to follow. Some important special cases and generalizations of the noise assumptions are discussed in Section III. To illustrate the use of the marginalized particle filter a synthetic example is given in Section IV. Finally, the application example is given in Section V and the conclusions are stated in Section VI.

II. MARGINALIZATION

The variance of the estimates obtained from the standard particle filter can be decreased by exploiting linear sub-structures in the model. The corresponding variables are marginalized out and estimated using an optimal linear filter. This is the main idea behind the marginalized particle filter. The goal of this section is to explain how the marginalized particle filter works by using three nested models. The models are nested in the sense that the first model is included in the second, which in turn is included in the third. The reason for presenting it in this fashion is to facilitate reader understanding, by incrementally extending the standard particle filter.

A. The Standard Particle Filter

The particle filter is used to get an approximation of the posterior density $p(x_t|Y_t)$ in the general model (1). This

approximation can then be used to obtain an estimate of some inference function, $g(\cdot)$, according to

$$I(g(x_t)) = E_{p(x_t|Y_t)}[g(x_t)] = \int g(x_t)p(x_t|Y_t)dx_t. \tag{4}$$

In the following the particle filter, as it was introduced in [16], will be referred to as the standard particle filter. For a thorough introduction to the standard particle filter the reader is referred to [11], [12]. The marginalized and the standard particle filter are closely related. The marginalized particle filter is given in Algorithm 1 and neglecting steps 4a and 4c results in the standard particle filter algorithm.

ALGORITHM 1: The marginalized particle filter

1) Initialization: For $i = 1, \dots, N$, initialize the particles, $x_{0|-1}^{n,(i)} \sim p_{x_0^n}(x_0^n)$ and set $\{x_{0|-1}^{l,(i)}, P_{0|-1}^{(i)}\} = \{\bar{x}_0^l, \bar{P}_0\}$.

2) For $i = 1, \dots, N$, evaluate the importance weights $q_t^{(i)} = p(y_t|X_t^{n,(i)}, Y_{t-1})$ and normalize
$$\tilde{q}_t^{(i)} = \frac{q_t^{(i)}}{\sum_{j=1}^{N} q_t^{(j)}}.$$

3) Particle filter measurement update (resampling): Resample N particles with replacement,
$$\Pr(x_{t|t}^{n,(i)} = x_{t|t-1}^{n,(j)}) = \tilde{q}_t^{(j)}.$$

4) Particle filter time update and Kalman filter:

   a) Kalman filter measurement update:
   Model 1: (10),
   Model 2: (10),
   Model 3: (22).

   b) Particle filter time update (prediction): For $i = 1, \dots, N$, predict new particles,
   $$x_{t+1|t}^{n,(i)} \sim p(x_{t+1|t}^n|X_t^{n,(i)}, Y_t).$$

   c) Kalman filter time update:
   Model 1: (11),
   Model 2: (17),
   Model 3: (23).

5) Set $t := t + 1$ and iterate from step 2.

The particle filter algorithm 1 is quite general and several improvements are available in the literature. It is quite common to introduce artificial noise in step 3 in order to counteract the degeneracy problem. Depending on the context various importance functions can be used in step 4b. In [11] several refinements to the particle filter algorithm are discussed.

B. Diagonal Model

The explanation of how the marginalized particle filter works is started by considering the following model,
Model 1:

$$x_{t+1}^n = f_t^n(x_t^n) \qquad\qquad + w_t^n, \tag{5a}$$

$$x_{t+1}^l = \qquad\qquad A_t^l(x_t^n)x_t^l + w_t^l, \tag{5b}$$

$$y_t = h_t(x_t^n) + C_t(x_t^n)x_t^l + e_t. \tag{5c}$$

The gaps in the equations above are placed there intentionally, in order to make the comparison to the general model (18) easier. The state noise is assumed white and Gaussian distributed

according to

$$w_t = \begin{bmatrix} w_t^l \\ w_t^n \end{bmatrix} \sim \mathcal{N}(0, Q_t), \quad Q_t = \begin{bmatrix} Q_t^l & 0 \\ 0 & Q_t^n \end{bmatrix}. \quad (6a)$$

The measurement noise is assumed white and Gaussian distributed according to

$$e_t \sim \mathcal{N}(0, R_t). \quad (6b)$$

Furthermore, x_0^l is Gaussian,

$$x_0^l \sim \mathcal{N}(\bar{x}_0, \bar{P}_0). \quad (6c)$$

The density of x_0^n can be arbitrary, but it is assumed known. The A^l and C matrices are arbitrary. $\square$

Model 1 is called "diagonal model" due to the diagonal structure of the state equation (5a) – (5b). The aim of recursively estimating the posterior density $p(x_t|Y_t)$ can be accomplished using the standard particle filter. However, conditioned on the nonlinear state variable, x_t^n, there is a linear sub-structure in (5), given by (5b). This fact can be used to obtain better estimates of the linear states. Analytically marginalizing out the linear state variables from $p(x_t|Y_t)$ and using Bayes' theorem gives ($X_t^n = \{x_i^n\}_{i=0}^t$)

$$p(x_t^l, X_t^n|Y_t) = \underbrace{p(x_t^l|X_t^n, Y_t)}_{\text{Optimal KF}} \underbrace{p(X_t^n|Y_t)}_{\text{PF}}, \quad (7)$$

where $p(x_t^l|X_t^n, Y_t)$ is analytically tractable. It is given by the Kalman filter (KF), see Lemma 2.1 below for the details. Furthermore, $p(X_t^n|Y_t)$ can be estimated using the particle filter (PF). If the same number of particles are used in the standard particle filter and the marginalized particle filter the latter will, intuitively, provide better estimates. The reason for this is that the dimension of $p(x_t^n|Y_t)$ is smaller than the dimension of $p(x_t^l, x_t^n|Y_t)$, implying that the particles occupy a lower dimensional space. Another reason is that optimal algorithms are used in order to estimate the linear state variables. Let $\hat{I}_N^s(g(x_t))$ denote the estimate of (4) using the standard particle filter with N particles. When the marginalized particle filter is used the corresponding estimate is denoted by $\hat{I}_N^m(g(x_t))$. Under certain assumptions the following central limit theorem holds,

$$\sqrt{N}(\hat{I}_N^s(g(x_t)) - I(g(x_t))) \implies \mathcal{N}(0, \sigma_s^2), \quad N \to \infty \quad (8a)$$

$$\sqrt{N}(\hat{I}_N^m(g(x_t)) - I(g(x_t))) \implies \mathcal{N}(0, \sigma_m^2), \quad N \to \infty \quad (8b)$$

where $\sigma_s^2 \geq \sigma_m^2$. A formal proof of (8) is provided in [14], [13]. For the sake of notational brevity the dependence of x_t^n in A_t, C_t, and h_t are suppressed below.

Lemma 2.1: Given Model 1, the conditional probability density functions for $x_{t|t}^l$ and $x_{t+1|t}^l$ are given by

$$p(x_t^l|X_t^n, Y_t) = \mathcal{N}(\hat{x}_{t|t}^l, P_{t|t}), \quad (9a)$$

$$p(x_{t+1}^l|X_{t+1}^n, Y_t) = \mathcal{N}(\hat{x}_{t+1|t}^l, P_{t+1|t}), \quad (9b)$$

where

$$\hat{x}_{t|t}^l = \hat{x}_{t|t-1}^l + K_t(y_t - h_t - C_t\hat{x}_{t|t-1}^l), \quad (10a)$$

$$P_{t|t} = P_{t|t-1} - K_tC_tP_{t|t-1}, \quad (10b)$$

$$S_t = C_tP_{t|t-1}C_t^T + R_t, \quad (10c)$$

$$K_t = P_{t|t-1}C_t^TS_t^{-1}, \quad (10d)$$

and

$$\hat{x}_{t+1|t}^l = A_t^l\hat{x}_{t|t}^l, \quad (11a)$$

$$P_{t+1|t} = A_t^lP_{t|t}(A_t^l)^T + Q_t^l. \quad (11b)$$

The recursions are initiated with $\hat{x}_{0|-1}^l = \bar{x}_0$ and $P_{0|-1} = \bar{P}_0$.

Proof: Straightforward application of the Kalman filter [22], [21]. $\blacksquare$

The second density, $p(X_t^n|Y_t)$, in (7) will be approximated using the standard particle filter. Bayes' theorem and the Markov property inherent in the state-space model can be used to write $p(X_t^n|Y_t)$ as

$$p(X_t^n|Y_t) = \frac{p(y_t|X_t^n, Y_{t-1})p(x_t^n|X_{t-1}^n, Y_{t-1})}{p(y_t|Y_{t-1})}p(X_{t-1}^n|Y_{t-1}), \quad (12)$$

where an approximation of $p(X_{t-1}^n|Y_{t-1})$ is provided by the previous iteration of the particle filter. In order to perform the update (12) analytical expressions for $p(y_t|X_t^n, Y_{t-1})$ and $p(x_t^n|X_{t-1}^n, Y_{t-1})$ are needed. They are provided by the following lemma.

Lemma 2.2: For Model 1 $p(y_t|X_t^n, Y_{t-1})$ and $p(x_{t+1}^n|X_t^n, Y_t)$ are given by

$$p(y_t|X_t^n, Y_{t-1}) = \mathcal{N}(h_t + C_t\hat{x}_{t|t-1}^l, C_tP_{t|t-1}C_t^T + R_t), \quad (13a)$$

$$p(x_{t+1}^n|X_t^n, Y_t) = \mathcal{N}(f_t^n, Q_t^n). \quad (13b)$$

Proof: Basic facts about conditionally linear models, see e.g., [19], [36]. $\blacksquare$

The linear system (5b) – (5c) can now be formed for each particle, $\{x_t^{n,(i)}\}_{i=1}^N$ and the linear states can be estimated using the Kalman filter. This requires one Kalman filter associated with each particle. The overall algorithm for estimating the states in Model 1 is given in Algorithm 1. From this algorithm it should be clear that the only difference from the standard particle filter is that the time update (prediction) stage has been changed. In the standard particle filter the prediction stage is given solely by step 4b in Algorithm 1. Step 4a is referred to as the *measurement update* in the Kalman filter [21]. Furthermore, the prediction of the nonlinear state variables, $\hat{x}_{t+1|t}^n$ is obtained in step 4b. According to (5a) the prediction of the nonlinear state variables does not contain any information about the linear state variables. This implies that $\hat{x}_{t+1|t}^n$ cannot be used to improve the quality of the estimates of the linear state variables. However, if Model 1 is generalized by imposing a dependence between the linear and the nonlinear state variables in (5a) the prediction of the nonlinear state variables can be used to improve the estimates of the linear state variables. In the subsequent section it will be elaborated on how this affects the state estimation.

C. Triangular Model

Model 1 is now extended by including the term $A_t^n(x_t^n)x_t^l$ in the nonlinear state equation. This results in a "triangular model", defined below.

Model 2:

$$x_{t+1}^n = f_t^n(x_t^n) + A_t^n(x_t^n)x_t^l + w_t^n, \tag{14a}$$

$$x_{t+1}^l = \qquad\quad A_t^l(x_t^n)x_t^l + w_t^l, \tag{14b}$$

$$y_t = h_t(x_t^n) + C_t(x_t^n)x_t^l + e_t, \tag{14c}$$

with the same assumptions as in Model 1. □

Now, from (14a) it is clear that $\hat{x}_{t+1|t}^n$ does indeed contain information about the linear state variables. This implies that there will be information about the linear state variable, x_t^l, in the prediction of the nonlinear state variable, $\hat{x}_{t+1|t}^n$. To understand how this affects the derivation it is assumed that step 4b in Algorithm 1 has just been completed. This means that the predictions, $\hat{x}_{t+1|t}^n$, are available and the model can be written (the information in the measurement, y_t, has already been used in step 4a)

$$x_{t+1}^l = A_t^l x_t^l + w_t^l, \tag{15a}$$

$$z_t = A_t^n x_t^l + w_t^n, \tag{15b}$$

where

$$z_t = x_{t+1}^n - f_t^n. \tag{15c}$$

It is possible to interpret z_t as a measurement and w_t^n as the corresponding measurement noise. Since (15) is a linear state-space model with Gaussian noise, the optimal state estimate is given by the Kalman filter according to

$$\hat{x}_{t|t}^{l*} = \hat{x}_{t|t}^l + L_t(z_t - A_t^n \hat{x}_{t|t}^l), \tag{16a}$$

$$P_{t|t}^* = P_{t|t} - L_t N_t L_t^T, \tag{16b}$$

$$L_t = P_{t|t}(A_t^n)^T N_t^{-1}, \tag{16c}$$

$$N_t = A_t^n P_{t|t}(A_t^n)^T + Q_t^n, \tag{16d}$$

where "$*$" has been used to distinguish this second measurement update from the first one. Furthermore, $\hat{x}_{t|t}^l$, and $P_{t|t}$ are given by (10a) and (10b) respectively. The final step is to merge this second measurement update with the time update to obtain the predicted states. This results in

$$\hat{x}_{t+1|t}^l = A_t^l \hat{x}_{t|t}^l + L_t(z_t - A_t^n \hat{x}_{t|t}^l), \tag{17a}$$

$$P_{t+1|t} = A_t^l P_{t|t}(A_t^l)^T + Q_t^l - L_t N_t L_t^T, \tag{17b}$$

$$L_t = A_t^l P_{t|t}(A_t^n)^T N_t^{-1}, \tag{17c}$$

$$N_t = A_t^n P_{t|t}(A_t^n)^T + Q_t^n. \tag{17d}$$

For a formal proof of this the reader is referred to Appendix I. To make Algorithm 1 valid for the more general Model 2 the time update equation in the Kalman filter (11) has to be replaced by (17).

The second measurement update is called measurement update due to the fact that the mathematical structure is the same as a measurement update in the Kalman filter. However, strictly speaking it is not really a measurement update, since there does not exist any new measurement. It is better to think of this second update as a correction to the real measurement update, using the information in the prediction of the nonlinear state variables.

D. The General Case

In the previous two sections the mechanisms underlying the marginalized particle filter have been illustrated. It is now time to apply the marginalized particle filter to the most general model.

Model 3:

$$x_{t+1}^n = f_t^n(x_t^n) + A_t^n(x_t^n)x_t^l + G_t^n(x_t^n)w_t^n, \tag{18a}$$

$$x_{t+1}^l = f_t^l(x_t^n) + A_t^l(x_t^n)x_t^l + G_t^l(x_t^n)w_t^l, \tag{18b}$$

$$y_t = h_t(x_t^n) + C_t(x_t^n)x_t^l + e_t, \tag{18c}$$

where the state noise is assumed white and Gaussian distributed with

$$w_t = \begin{bmatrix} w_t^l \\ w_t^n \end{bmatrix} \sim \mathcal{N}(0, Q_t), \quad Q_t = \begin{bmatrix} Q_t^l & Q_t^{ln} \\ (Q_t^{ln})^T & Q_t^n \end{bmatrix}. \tag{19a}$$

The measurement noise is assumed white and Gaussian distributed according to

$$e_t \sim \mathcal{N}(0, R_t). \tag{19b}$$

Furthermore, x_0^l is Gaussian,

$$x_0^l \sim \mathcal{N}(\bar{x}_0, \bar{P}_0). \tag{19c}$$

The density of x_0^n can be arbitrary, but it is assumed known. □

In certain cases some of the assumptions can be relaxed. This will be discussed in the subsequent section. Before moving on it is worthwhile to explain how models used in some applications of marginalization relate to Model 3. In [23] the marginalized particle filter was applied to underwater navigation using a model corresponding to (18), save the fact that $G_t^n = I, G_t^l = I, f_t^l = 0, A_t^n = 0$. In [18] a model corresponding to linear state equations and a nonlinear measurement equation is applied to various problems, such as car positioning, terrain navigation, and target tracking. Due to its relevance this model will be discussed in more detail in Section III. Another special case of Model 3 has been applied to problems in communication theory in [9], [37]. The model used there is linear. However, depending on an indicator variable the model changes. Hence, this indicator variable can be thought of as the nonlinear state variable in Model 3. A good and detailed explanation of how to use the marginalized particle filter for this case can be found in [14]. They refer to the model as a jump Markov linear system.

Analogously to what has been done in (7), the filtering distribution, $p(x_t|Y_t)$, is split using Bayes' theorem,

$$p(x_t^l, X_t^n|Y_t) = p(x_t^l|X_t^n, Y_t)p(X_t^n|Y_t). \tag{20}$$

The linear state variables are estimated using the Kalman filter in a slightly more general setting than which was previously discussed. However, it is still the same three steps that are executed in order to estimate the linear state variables. The first step is a measurement update using the information available in y_t. The second step is a measurement update using the information available in $\hat{x}_{t+1|t}^n$ and finally there is a time update. The following theorem explains how the linear state variables are estimated.

Theorem 2.1: Using Model 3 the conditional probability density functions for x_t^l and x_{t+1}^l are given by

$$p(x_t^l|X_t^n, Y_t) = \mathcal{N}(\hat{x}_{t|t}^l, P_{t|t}), \tag{21a}$$

$$p(x_{t+1}^l|X_{t+1}^n, Y_t) = \mathcal{N}(\hat{x}_{t+1|t}^l, P_{t+1|t}), \tag{21b}$$

where

$$\hat{x}_{t|t}^l = \hat{x}_{t|t-1}^l + K_t(y_t - h_t - C_t\hat{x}_{t|t-1}^l), \tag{22a}$$

$$P_{t|t} = P_{t|t-1} - K_t M_t K_t^T, \tag{22b}$$

$$M_t = C_t P_{t|t-1} C_t^T + R_t, \tag{22c}$$

$$K_t = P_{t|t-1} C_t^T M_t^{-1}, \tag{22d}$$

and

$$\hat{x}_{t+1|t}^l = \bar{A}_t^l \hat{x}_{t|t}^l + G_t^l(Q_t^{ln})^T(G_t^n Q_t^n)^{-1} z_t$$
$$+ f_t^l + L_t(z_t - A_t^n \hat{x}_{t|t}^l), \tag{23a}$$

$$P_{t+1|t} = \bar{A}_t^l P_{t|t}(\bar{A}_t^l)^T + G_t^l \bar{Q}_t^l(G_t^l)^T - L_t N_t L_t^T, \tag{23b}$$

$$N_t = A_t^n P_{t|t}(A_t^n)^T + G_t^n Q_t^n(G_t^n)^T, \tag{23c}$$

$$L_t = \bar{A}_t^l P_{t|t}(A_t^n)^T N_t^{-1}, \tag{23d}$$

where

$$z_t = x_{t+1}^n - f_t^n, \tag{24a}$$

$$\bar{A}_t^l = A_t^l - G_t^l(Q_t^{ln})^T(G_t^n Q_t^n)^{-1} A_t^n, \tag{24b}$$

$$\bar{Q}_t^l = Q_t^l - (Q_t^{ln})^T(Q_t^n)^{-1} Q_t^{ln}. \tag{24c}$$

Proof: See Appendix I. ∎

It is worth noting that if the cross-covariance, Q_t^{ln}, between the two noise sources w_t^n and w_t^l is zero, then $\bar{A}_t^l = A_t^l$ and $\bar{Q}_t^l = Q_t^l$. The first density, $p(x_t^l|X_t^n, Y_t)$, on the right hand side in (20) is now taken care of. In order for the estimation to work the second density, $p(X_t^n|Y_t)$, in (20) is taken care of according to (12). The analytical expressions for $p(y_t|X_t^n, Y_{t-1})$ and $p(x_t^n|X_{t-1}^n, Y_{t-1})$ are provided by the following theorem.

Theorem 2.2: For Model 3 $p(y_t|X_t^n, Y_{t-1})$ and $p(x_{t+1}^n|X_t^n, Y_t)$ are given by

$$p(y_t|X_t^n, Y_{t-1}) = \mathcal{N}(h_t + C_t\hat{x}_{t|t-1}^l, \ C_t P_{t|t-1} C_t^T + R_t), \tag{25a}$$

$$p(x_{t+1}^n|X_t^n, Y_t) = \mathcal{N}(f_t^n + A_t^n\hat{x}_{t|t}^l, \ A_t^n P_{t|t}(A_t^n)^T$$
$$+ G_t^n Q_t^n(G_t^n)^T). \tag{25b}$$

Proof: Basic facts about conditionally linear models, see [19]. The details for this particular case can be found in [36]. ∎

The details for estimating the states in Model 3 have now been derived, and the complete algorithm is Algorithm 1. As pointed out before, the only difference between this algorithm and the standard particle filtering algorithm is that the prediction stage is different. If steps 4a and 4c are removed from Algorithm 1 the standard particle filter algorithm is obtained.

In this article the most basic form of the particle filter has been used. Several more refined variants exist, which in certain applications can give better performance. However, since the aim of this article is to communicate the idea of marginalization in a general linear/nonlinear state-space model the standard particle filter has been used. It is straightforward

to adjust the algorithm given in this paper to accommodate e.g., the auxiliary particle filter [34] and the Gaussian particle filter [26], [27]. Several ideas are also given in the articles collected in [11].

The estimates as expected means of the linear state variables and their covariances are given by [32]

$$\hat{x}_{t|t}^l = \sum_{i=1}^N \tilde{q}_t^{(i)} \hat{x}_{t|t}^{l,(i)} \approx E_{p(x_t^l|Y_t)}\left[x_t^l\right], \tag{26a}$$

$$\hat{P}_{t|t} = \sum_{i=1}^N \tilde{q}_t^{(i)}\left(P_{t|t}^{(i)} + (\hat{x}_{t|t}^{l,(i)} - \hat{x}_{t|t}^l)(\hat{x}_{t|t}^{l,(i)} - \hat{x}_{t|t}^l)^T\right) \tag{26b}$$

$$\approx E_{p(x_t^l|Y_t)}\left[\left((x_t^l)^2 - E_{p(x_t^l|Y_t)}\left[(x_t^l)^2\right]\right)^2\right]. \tag{26c}$$

where $\tilde{q}_t^{(i)}$ are the normalized importance weights, provided by step 2 in Algorithm 1.

III. IMPORTANT SPECIAL CASES AND EXTENSIONS

Model 3 is quite general indeed and in most applications special cases of it are used. This fact, together with some extensions will be the topic of this section.

The special cases are just reductions of the general results presented in the previous section. However, they still deserve some attention in order to highlight some important mechanisms. It is worth mentioning that linear sub-structures can enter the model more implicitly as well, for example, by modeling colored noise and by sensor offsets and trends. These modeling issues are treated in several introductory texts on Kalman filtering, see e.g., Section 8.2.4 in [17]. In the subsequent section some noise modeling aspects are discussed. This is followed by a discussion of a model with linear state equations and a nonlinear measurement equation.

A. Generalized Noise Assumptions

The Gaussian noise assumption can be relaxed in two special cases. First, if the measurement equation (18c) does not depend on the linear state variables, x_t^l, i.e., $C_t(x_t^n) = 0$, the measurement noise can be arbitrarily distributed. In this case (18c) does not contain any information about the linear state variables, and hence cannot be used in the Kalman filter. It is solely used in the particle filter part of the algorithm, which can handle all probability density functions.

Second, if $A_t^n(x_t^n) = 0$ in (18a), the nonlinear state equation will be independent of the linear states, and hence cannot be used in the Kalman filter. This means that the state noise, w_t^n, can be arbitrarily distributed.

The noise covariances can depend on the nonlinear state variables, i.e., $R_t = R_t(x_t^n)$ and $Q_t = Q_t(x_t^n)$. This is useful for instance in terrain navigation, where the nonlinear state variable includes information about the position. Using the horizontal position and a geographic information system (GIS) on-board the aircraft noise covariances depending on the characteristics of the terrain at the current horizontal position can be motivated. This issue will be elaborate upon in Section V.

B. An Important Model Class

A quite important special case of Model 3, is a model with linear state equations and a nonlinear measurement equation. In Model 4 below such a model is defined.

Model 4:

$$x_{t+1}^n = A_{n,t}^n x_t^n + A_{l,t}^n x_t^l + G_t^n w_t^n, \qquad (27a)$$

$$x_{t+1}^l = A_{n,t}^l x_t^n + A_{l,t}^l x_t^l + G_t^l w_t^l, \qquad (27b)$$

$$y_t = h_t(x_t^n) \qquad + e_t, \qquad (27c)$$

with $w_t^n \sim \mathcal{N}(0, Q_t^n)$ and $w_t^l \sim \mathcal{N}(0, Q_t^l)$. The distribution for e_t can be arbitrary, but it is assumed known. $\square$

The measurement equation (27c) does not contain any information about the linear state variable, x_t^l. Hence, as far as the Kalman filter is concerned (27c) cannot be used in estimating the linear states. Instead all information from the measurements enter the Kalman filter implicitly via the second measurement update using the nonlinear state equation (27a) and the prediction of the nonlinear state, $\hat{x}_{t+1|t}^n$, as a measurement. This means that in Algorithm 1, step 4a can be left out. In this case the second measurement update is much more than just a correction to the first measurement update. It is the only way in which the information in y_t enters the algorithm.

Model 4 is given special attention as several important state estimation problems can be modeled in this way. Examples include positioning, target tracking and collision avoidance [18], [4]. For more information on practical matters concerning modeling issues, see e.g., [30], [29], [4], [32]. In the applications mentioned above the nonlinear state variable, x_t^n, usually corresponds to the position, whereas the linear state variable, x_t^l, corresponds to velocity, acceleration and bias terms.

If Model 4 is compared to Model 3 it can be seen that the matrices A_t^n, A_t^l, G_t^n, and G_t^l are independent of x_t^n in Model 4, which implies that

$$P_{t|t}^{(i)} = P_{t|t} \qquad \forall i = 1, \ldots, N. \qquad (28)$$

This follows from (23b) – (23d) in Theorem 2.1. According to (28) only one instead of N Riccati recursions is needed, which leads to a substantial reduction in computational complexity. This is of course very important in real-time implementations. A further study of the computational complexity of the marginalized particle filter can be found in [25].

If the dynamics in (18a) – (18b) is almost linear it can be linearized to obtain a model described by (27a) – (27b). Then the extended Kalman filter can be used instead of the Kalman filter. As is explained in [30], [29] it is common that the system model is almost linear, whereas the measurement model is severely nonlinear. In these cases use the particle filter for the severe nonlinearities and the extended Kalman filter for the mild nonlinearities.

IV. An Illustrating Example

In order to make things as simple as possible the following two dimensional model will be used

$$x_{t+1} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_t + w_t, \qquad (29a)$$

$$y_t = h(z_t) + e_t, \qquad (29b)$$

where the state vector is $x_t = \begin{bmatrix} z_t & \dot{z}_t \end{bmatrix}^T$. Hence, the state consists of a physical variable and its derivative. Models of this kind are very common in applications. One example is bearings only tracking, where the objective is to estimate the angle and angular velocity and the nonlinear measurement depends on the antenna diagram. Another common application is state estimation in a DC-motor, where the angular position is assumed to be measured nonlinearly. As a final application terrain navigation in one dimension is mentioned, where the measurement is given by a map. A more realistic terrain navigation example is discussed in Section V.

Model (29) is linear in $\dot{z}_t$ and nonlinear in z_t. The state vector can thus be partitioned as $x_t = \begin{bmatrix} x_t^n & x_t^l \end{bmatrix}^T$, which implies that (29) can be written as

$$x_{t+1}^n = x_t^n \qquad + T x_t^l + w_t^n, \qquad (30a)$$

$$x_{t+1}^l = \qquad x_t^l + w_t^l, \qquad (30b)$$

$$y_t = h_t(x_t^n) \qquad + e_t, \qquad (30c)$$

This corresponds to the triangular model given in Model 2. Hence, the Kalman filter for the linear state variable is given by (22) – (24) where the nonlinear state is provided by the particle filter. The estimate of the linear state variable is given by (23a) which for this example is

$$\hat{x}_{t+1|t}^l = (1 - l_t T)\hat{x}_{t|t}^l + l_t T \frac{x_{t+1}^n - x_t^n}{T}, \qquad (31)$$

where

$$n_t = T^2 p_{t|t} + q_t^n, \qquad l_t = \frac{T}{n_t} p_{t|t}. \qquad (32)$$

Intuitively (31) makes sense, since the velocity estimate is given as a weighted average of the current velocity and the estimated momentary velocity, where the weights are computed from the Kalman filter quantities. In cases where (29a) is motivated by Newtons' force law the unknown force is modeled as a disturbance and $q_t^n = 0$. This implies that (31) is reduced to

$$\hat{x}_{t+1|t}^l = \frac{x_{t+1}^n - x_t^n}{T}. \qquad (33)$$

Again this can intuitively be understood, since conditioned on the knowledge of the nonlinear state variable, (30a) can be written

$$x_t^l = \frac{x_{t+1}^n - x_t^n}{T}. \qquad (34)$$

Thus, (30b) does not add any information for the Kalman filter, since x_t^l is a deterministic function of the known nonlinear state variable.

V. Integrated Aircraft Navigation

As was explained in the introduction, the integrated navigation system in the Swedish fighter aircraft Gripen consists of an inertial navigation system (INS), a terrain-aided positioning (TAP) system and an integration filter. This filter fuses the information from INS with the information from TAP, see Fig. 1. The currently used integration filter, is likely to be changed to a marginalized particle filter in the future for Gripen, see Fig. 2. A first step in this direction was taken
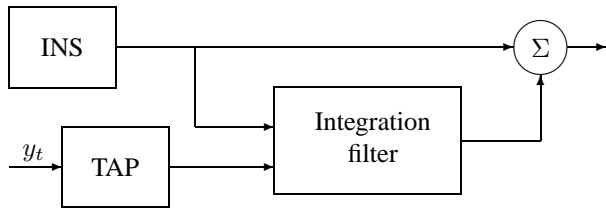
Fig. 1. The integrated navigation system consists of an inertial navigation system (INS), a terrain-aided positioning (TAP) system and an integration filter. The integration filter fuse the information from INS with the information from TAP.
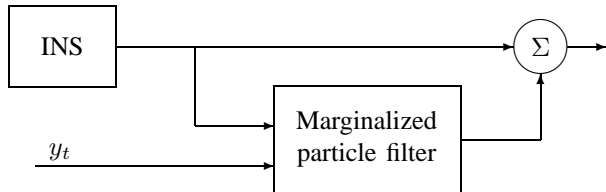

Fig. 2. Using the marginalized particle filter for navigation. The terrain information is now incorporated directly in the marginalized particle filter. The radar altimeter delivers the hight measurement y_t.

in [18], where a six dimensional model was used for integrated navigation. In six dimensions, the particle filter is possible to use, but better performance can be obtained. As demonstrated in [18], 4000 particles in the marginalized filter outperforms 60000 particles in the standard particle filter.

The feasibility study presented here applies marginalization to a more realistic nine dimensional sub-model of the total integrated navigation system. Already here, the dimensionality has proven to be too large for the particle filter to be applied directly. The example contains all ingredients of the total system, and the principle is scalable to the full 27-dimensional state vector. The model can be simulated and evaluated in a controlled fashion, see [32] for more details. In the subsequent sections the results from field trials are presented.

A. The Dynamic Model

In order to apply the marginalized particle filter to the navigation problem a dynamic model of the aircraft is needed. In this article the overall structure of this model is discussed. For details the reader is referred to [32] and the references therein. The errors in the states are estimated instead of the absolute states. The reason is that the dynamics of the errors are typically much slower than the dynamics of the absolute states. The model has the following structure

$$x_{t+1}^n = A_{n,t}^n x_t^n + A_{l,t}^n x_t^l + G_t^n w_t^n, \tag{35a}$$

$$x_{t+1}^l = A_{n,t}^l x_t^n + A_{l,t}^l x_t^l + G_t^l w_t^l, \tag{35b}$$

$$y_t = h\left(\begin{bmatrix} L_t \\ l_t \end{bmatrix} + x_t^n\right) + e_t. \tag{35c}$$

There are 7 linear states, and 2 nonlinear states. The linear states consist of 2 velocity states and 3 states for the aircraft in terms of heading, roll, and pitch. Finally there are 2 states for the accelerometer bias. The nonlinear states correspond to the error in the horizontal position, which is expressed in latitude, L_t, and longitude, l_t.

The total dimension of the state vector is thus 9, which is too large to be handled by the particle filter. The highly nonlinear nature of measurement equation (35c), due to the terrain elevation database, implies that an extended Kalman filter cannot be used. However, the model described by (35) clearly fits into the framework of the marginalized particle filter.

The measurement noise in (35c) deserves some special attention. The radar altimeter, which is used to measure the ground clearance, interprets any echo as the ground. This is a problem when flying over trees. The tree tops will be interpreted as the ground, with a false measurement as a result. One simple, but effective, solution to this problem is to model the measurement noise as

$$p_{e_t}(\cdot) = \pi \mathcal{N}(m_1, \sigma_1) + (1 - \pi)\mathcal{N}(m_2, \sigma_2), \tag{36}$$

where π is the probability of obtaining an echo from the ground, and $(1 - \pi)$ is the probability of obtaining an echo from the tree tops. The probability density function (36) is shown in Fig. 3. Experiments have shown that this, in spite
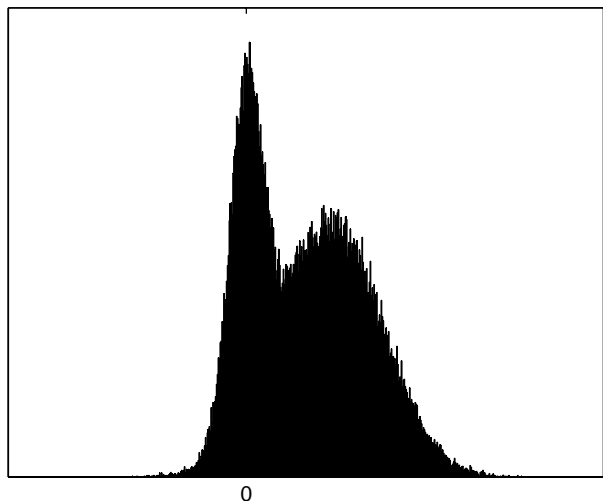

Fig. 3. A typical histogram of the error in the data from the radar altimeter. The first peak corresponds to the error in the ground reading and the second peak corresponds to the error in the readings from the tree tops.

of its simplicity, is a quite accurate model [10]. Furthermore, m_1, m_2, σ_1, σ_2, and π in (36) can be allowed to depend on the current horizontal position, L_t, l_t. In this way information from the terrain data base can be inferred on the measurement noise in the model. Using this information it is possible to model whether the aircraft is flying over open water or over a forest.

B. Result

The flight that has been used is shown in Fig. 4. This is a fairly tough flight for the algorithm, in the sense that during some intervals data are missing, and sometimes the radar altimeter readings become unreliable. This happens at high altitudes and during sharp turns (large roll angle), respectively. In order to get a fair understanding of the algorithms performance, 100 Monte Carlo simulations with the same data have
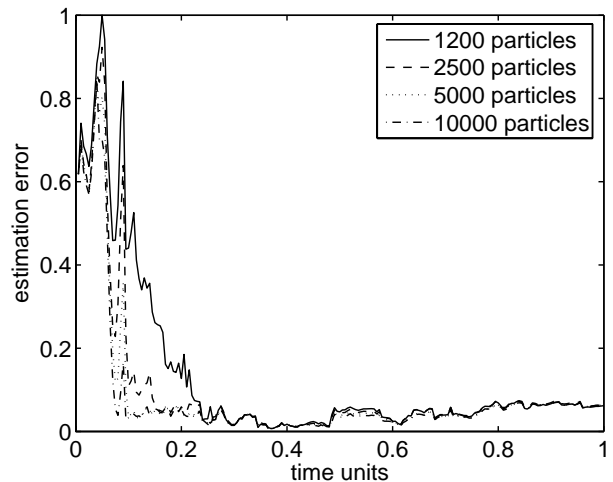
Fig. 5. The horizontal position error as a function of time units for different numbers of particles. The marginalized particle filter given in Algorithm 1 has been used.

Fig. 4. The flight path used for testing the algorithm. The flight path is clockwise and the dark regions in the figure are open water.

been performed, where only the noise realizations have been changed from one simulation to the other. Many parameters have to be chosen, but only the number of particles used are commented here (see [15] for more details). In Fig. 5 a plot of the error in horizontal position as a function of time is presented, for different number of particles. The true position is provided by the differential GPS (DGPS). From this figure it is obvious that the estimate improves as more particles are used. This is natural since the theory states that the densities are approximated better the more particles used. The difference in performance is mainly during the transient, where it can be motivated to use more particles. By increasing the number of particles the convergence time is significantly reduced and a better estimate is obtained. This is true up to 5000 particles. Hence, 5000 particles where used in this study. The algorithm can be further improved, and in [15] several suggestions are given.

The conclusion from this study is that the marginalized particle filter performs well, and provides an interesting and powerful alternative to methods currently used in integrated aircraft navigation systems.

VI. CONCLUSIONS

The marginalization techniques have systematically been applied to general nonlinear and non-Gaussian state-space models, with linear sub-structures. This has been done in several steps, where each step implies a certain modification of the standard particle filter. The first step was to associate one Kalman filter with each particle. These Kalman filters were used to estimate the linear states. The second step was to use the prediction of the nonlinear state as an additional measurement. This was used to obtain better estimates of the linear state variables. The complete details for the marginalized particle filter were derived for a general nonlinear and non-Gaussian state-space model. Several important special cases were also described. Conditions implying that all the Kalman filters will obey the same Riccati recursion were given.

Finally, a terrain navigation application with real data from the Swedish fighter aircraft Gripen was presented. The particle filter is not a feasible algorithm for the full nine-state model since a huge number of particles would be needed. However, since only two states (the aircrafts horizontal position) appear nonlinearly in the measurement equation, a special case of the general marginalization algorithm can be applied. A very good result can be obtained with only 5000 particles, which is readily possible to implement in the computer currently used in the aircraft.

APPENDIX I
PROOF FOR THEOREM 2.1

The proof of (16) and (17) is provided as a special case of the proof below.

Proof: For the sake of notational brevity the dependence on x_t^n in (18) is suppressed in this proof. Write (18) as

$$x_{t+1}^l = f_t^l + A_t^l x_t^l + G_t^l w_t^l, \tag{37a}$$

$$z_t^1 = A_t^n x_t^l + G_t^n w_t^n, \tag{37b}$$

$$z_t^2 = C_t x_t^l + e_t, \tag{37c}$$

where z_t^1 and z_t^2 are defined as

$$z_t^1 = x_{t+1}^n - f_t^n, \tag{37d}$$

$$z_t^2 = y_t - h_t. \tag{37e}$$

Inspection of the above equations gives that z_t^1 and z_t^2 can both be thought of as measurements, since mathematically (37b) and (37c) possess the structure of measurement equations. The fact that there is a cross-correlation between the two noise processes w_t^l and w_t^n, since $Q_t^{ln} \neq 0$ in (19a), has to be taken care of. This can be accomplished using the Gram-Schmidt procedure to de-correlate the noise [17], [21]. Instead of w_t^l the following can be used

$$\bar{w}_t^l = w_t^l - \mathrm{E}[w_t^l(w_t^n)^T](\mathrm{E}[w_t^n(w_t^n)^T])^{-1}w_t^n$$
$$= w_t^l - Q_t^{ln}(Q_t^n)^{-1}w_t^n, \tag{38}$$

resulting in $\mathrm{E}[\bar{w}_t^l(w_t^n)^T] = 0$ and

$$\bar{Q}_t^l = \mathrm{E}[\bar{w}_t^l(\bar{w}_t^l)^T] = Q_t^l - Q_t^{ln}(Q_t^n)^{-1}Q_t^{ln}. \tag{39}$$

Using (37b) and (38), (37a) can be rewritten according to (G_t^n is assumed invertible. The case of a non-invertible G_t^n is treated in [5])

$$x_{t+1}^l = A_t^l x_t^l + G_t^l[\bar{w}_t^l + Q_t^{ln}(Q_t^n)^{-1}(G_t^n)^{-1}(z_t^1$$
$$- A_t^n x_t^l)] + f_t^l, \tag{40}$$

$$= \bar{A}_t^l x_t^l + G_t^l \bar{w}_t^l + G_t^l Q_t^{ln}(G_t^n Q_t^n)^{-1}z_t^1 + f_t^l, \tag{41}$$

where

$$\bar{A}_t^l = A_t^l - G_t^l Q_t^{ln}(G_t^n Q_t^n)^{-1}A_t^n. \tag{42}$$

The de-correlated system is

$$x_{t+1}^l = f_t^l + \bar{A}_t^l x_t^l + G_t^l Q_t^{ln}(G_t^n Q_t^n)^{-1}z_t^1 + G_t^l \bar{w}_t^l, \tag{43a}$$

$$z_t^1 = A_t^n x_t^l + G_t^n w_t^n, \tag{43b}$$

$$z_t^2 = C_t x_t^l + e_t, \tag{43c}$$

which is a linear system with Gaussian noise. Moreover, from (37d) and (37e) it can be seen that Z_t^1 and Z_t^2 are known if X_{t+1}^n and Y_t are known. The actual proof, using induction, of the theorem can now be started. At time zero; $p(x_0^l|X_0^n, Y_{-1}) = p(x_0^l|x_0^n) = \mathcal{N}(\bar{x}_0^l, \bar{P}_0)$. Now, assume that $p(x_t^l|X_t^n, Y_{t-1})$ is Gaussian at an arbitrary time, t.

The recursions are divided into three parts. First, the information available in the actual measurement, y_t, i.e., z_t^2 is used. Once the measurement update has been performed the estimates, $\hat{x}_{t|t}^l$ and $P_{t|t}$ are available. These can now be used to calculate the predictions of the nonlinear state, $\hat{x}_{t+1|t}^n$. These predictions will provide new information about the system. Second, this new information is incorporated by performing a second measurement update using the artificial measurement, z_t^1. Finally a time update, using the result from the second step, is performed.

Part 1: Assume that both $p(x_t^l|X_t^n, Y_{t-1}) = \mathcal{N}(\hat{x}_{t|t-1}^l, P_{t|t-1})$ and z_t^2 are available. This means that $p(x_t^l|X_t^n, Y_t)$ can be computed,

$$p(x_t^l|X_t^n, Y_t) = \frac{p(y_t|x_t^n, x_t^l)p(x_t^l|X_t^n, Y_{t-1})}{\int p(y_t|x_t^n, x_t^l)p(x_t^l|X_t^n, Y_{t-1})dx_t^l}. \tag{44}$$

Using the fact that the measurement noise and thereby $p(y_t|x_t^n, x_t^l)$ is Gaussian and the Kalman filter [1] it can be seen that $p(x_t^l|X_t^n, Y_t) = \mathcal{N}(\hat{x}_{t|t}^l, P_{t|t})$ where

$$\hat{x}_{t|t}^l = \hat{x}_{t|t-1}^l + K_t(z_t^2 - C_t\hat{x}_{t|t-1}^l), \tag{45a}$$

$$P_{t|t} = P_{t|t-1} - K_t M_t K_t^T, \tag{45b}$$

$$K_t = P_{t|t-1}C_t^T M_t^{-1}, \tag{45c}$$

$$M_t = C_t P_{t|t-1}C_t^T + R_t. \tag{45d}$$

Part 2: At this stage z_t^1 becomes available. Use

$$p(x_t^l|X_{t+1}^n, Y_t) = \frac{p(x_{t+1}^n|x_t^n, x_t^l)p(x_t^l|X_t^n, Y_t)}{\int p(x_{t+1}^n|x_t^n, x_t^l)p(x_t^l|X_t^n, Y_t)dx_t^l} \tag{46}$$

analogously to part 1 $p(x_t^l|X_{t+1}^n, Y_t) = \mathcal{N}(\hat{x}_{t|t}^{l*}, P_{t|t}^*)$ where

$$\hat{x}_{t|t}^{l*} = \hat{x}_{t|t}^l + L_t(z_t^1 - A_t^n \hat{x}_{t|t}^l), \tag{47a}$$

$$P_{t|t}^* = P_{t|t} - L_t N_t^* L_t^T, \tag{47b}$$

$$L_t = P_{t|t}(A_t^n)^T(N_t^*)^{-1}, \tag{47c}$$

$$N_t^* = A_t^n P_{t|t}(A_t^n)^T + G_t^n Q_t^n(G_t^n)^T. \tag{47d}$$

Part 3: The final part is the time update, i.e., to compute

$$p(x_{t+1}^l|X_{t+1}^n, Y_t) =$$
$$\int p(x_{t+1}^l|x_{t+1}^n, x_t^n, x_t^l)p(x_t^l|X_{t+1}^n, Y_t)dx_t^l. \tag{48}$$

Since the state noise is Gaussian this corresponds to the time update handled by the Kalman filter. Hence, $p(x_{t+1}^l|X_{t+1}^n, Y_t) = \mathcal{N}(\hat{x}_{t+1|t}^l, P_{t+1|t})$ where

$$\hat{x}_{t+1|t}^l = \bar{A}_t^l \hat{x}_{t|t}^l + G_t^l(Q_t^{ln})^T(G_t^n Q_t^n)^{-1}z_t^1$$
$$+ f_t^l + L_t(z_t^1 - A_t^n \hat{x}_{t|t}^l), \tag{49a}$$

$$P_{t+1|t} = \bar{A}_t^l P_{t|t}(\bar{A}_t^l)^T + G_t^l \bar{Q}_t^l(G_t^l)^T - L_t N_t L_t^T, \tag{49b}$$

$$L_t = \bar{A}_t^l P_{t|t}(A_t^n)^T N_t^{-1}, \tag{49c}$$

$$N_t = A_t^n P_{t|t}(A_t^n)^T + G_t^n Q_t^n(G_t^n)^T. \tag{49d}$$

∎

REFERENCES

[1] B.D.O. Anderson and J.B. Moore. *Optimal Filtering*. Information and system science series. Prentice Hall, Englewood Cliffs, New Jersey, 1979.

[2] C. Andrieu and A. Doucet. Particle filtering for partially observed Gaussian state space models. *Journal of the Royal Statistical Society*, 64(4):827–836, 2002.

[3] C. Andrieu and S.J. Godsill. A particle filter for model based audio source separation. In *International Workshop on Independent Component Analysis and Blind Signal Separation (ICA 2000)*, Helsinki, Finland, June 2000.

[4] Y. Bar-Shalom and X-R. Li. *Estimation and Tracking: Principles, Techniques, and Software*. Artech House, 1993.

[5] N. Bergman. *Recursive Bayesian Estimation: Navigation and Tracking Applications*. PhD thesis, Linköping University, 1999. Dissertation No. 579.

[6] N. Bergman, L. Ljung, and F. Gustafsson. Terrain navigation using Bayesian statistics. *IEEE Control Systems Magazine*, 19(3):33–40, June 1999.

[7] G. Casella and C.P. Robert. Rao-Blackwellisation of sampling schemes. *Biometrika*, 83(1):81–94, 1996.

[8] R. Chen and J.S. Liu. Mixture Kalman filters. *Journal of the Royal Statistical Society*, 62(3):493–508, 2000.

[9] R. Chen, X. Wang, and J.S. Liu. Adaptive joint detection in flat-fading channels via mixture Kalman filtering. *IEEE Transactions on Information Theory*, 46(6):2079–2094, 2000.

[10] C. Dahlgren. Non-linear black box modelling of JAS 39 Gripen's radar altimeter. LiTH-ISY-EX-1958, Automatic control and communication systems, Linköping university, Oct. 1998.

[11] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer Verlag, 2001.

[12] A. Doucet, S.J. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[13] A. Doucet, N. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. Technical Report CUED/F-INFENG/TR 359, Signal Processing Group, Department of Engineering, University of Cambridge, Trupington street, CB2 1PZ Cambridge, 1999.

[14] A. Doucet, N. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. *IEEE Transactions on Signal Processing*, 49(3):613–624, 2001.

[15] P. Frykman. Applied particle filters in integrated aircraft navigation. LiTH-ISY-EX-3406, Automatic control and communication systems, Linköping university, Apr. 2003.

[16] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. A novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEE Proceedings on Radar and Signal Processing*, volume 140, pages 107–113, 1993.

[17] F. Gustafsson. *Adaptive Filtering and Change Detection*. John Wiley & Sons, 2000.

[18] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P-J. Nordlund. Particle filters for positioning, navigation and tracking. *IEEE Transactions on Signal Processing*, 50(2):425–437, Feb 2002.

[19] A.C. Harvey. *Forecasting, structural time series models and the Kalman filter*. Cambridge University Press, Cambridge, UK, 1989.

[20] A.H. Jazwinski. *Stochastic processes and filtering theory*. Mathematics in science and engineering. Academic Press, New York, 1970.

[21] T. Kailath, A.H. Sayed, and B. Hassibi. *Linear Estimation*. Information and System Sciences Series. Prentice Hall, Upper Saddle River, New Jersey, 2000.

[22] R. E. Kalman. A new approach to linear filtering and prediction problems. *Trans. AMSE, J. Basic Engineering*, 82:35–45, 1960.

[23] R. Karlsson and F. Gustafsson. Particle filter and Cramér-Rao lower bound for underwater navigation. In *proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Hong Kong, Apr. 2003.

[24] R. Karlsson and F. Gustafsson. Particle filter for underwater navigation. In *Statistical Signal Processing Workshop (SSP'03)*, pages 509–512, St. Louis, USA, Sep. 2003.

[25] R. Karlsson, T. Schön, and F. Gustafsson. Complexity analysis of the marginalized particle filter. Technical Report LiTH-ISY-R-2611, Department of Electrical Engineering, Linköping University, 2004.

[26] J.H. Kotecha and P.M. Djuric. Gaussian particle filtering. *IEEE Transactions on Signal Processing*, 51(10):2592–2601, 2003.

[27] J.H. Kotecha and P.M. Djuric. Gaussian sum particle filtering. *IEEE Transactions on Signal Processing*, 51(10):2602–2610, 2003.

[28] P. Li, R. Goodall, and V. Kadirkamanathan. Parameter estimation of railway vehicle dynamic model using Rao-Blackwellised particle filter. In *proceedings of the European Control Conference (ECC)*, Cambridge, UK, Sep. 2003.

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

[30] X.R. Li and V.P. Jilkov. Survey of maneuvering target tracking-part i: dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364, Oct. 2003.

[31] J. S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Series in Statistics. Springer, New York, USA, 2001.

[32] P-J. Nordlund. *Sequential Monte Carlo Filters and Integrated Navigation*. Licentiate thesis, Linköping university, 2002. Thesis No. 945.

[33] J. Palmqvist. *On Integrity Monitoring of Integrated Navigation Systems*. Licentiate thesis, Linköping university, 1997. Thesis No. 600.

[34] M.K. Pitt and N. Shephard. Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, 94(446):590–599, June 1999.

[35] T. Schön and F. Gustafsson. Particle filters for system identification of state-space models linear in either parameters or states. In *proceedings of the 13th IFAC Symposium on System Identification*, pages 1287–1292, Rotterdam, The Netherlands, Sep. 2003.

[36] T. Schön. *On Computational Methods for Nonlinear Estimation*. Licentiate thesis, Linköping University, Oct. 2003. Thesis No. 1047.

[37] X. Wang, R. Chen, and D. Guo. Delayed-pilot sampling for mixture Kalman filter with application in fading channels. *IEEE Transactions on Signal Processing*, 50(2):241–254, Feb. 2002.

Thomas Schön received the BSc degree in Business Administration and Economics in Feb. 2001, in Sep. 2001 he received the MSc degree in Applied Physics and Electrical Engineering and in Oct. 2003 he received the Licentiate of Engineering degree, all from Linköping University, Linköping, Sweden. He has since Dec. 2001 been pursuing the PhD degree at the division of automatic control and communication systems, Linköping University.

Fredrik Gustafsson received the MSc degree in Electrical Engineering in 1988 and the PhD degree in Automatic Control in 1992, both from Linköping University, Linköping, Sweden.

He is Professor of communication systems with the department of Electrical Engineering, Linköping University. His research is focused on sensor fusion and statistical methods in signal processing, with applications to aerospace, automotive, audio and communication systems. He is the author of four books, over hundred international papers and fourteen patents. He is also a co-founder of three spin-off companies in these areas. Prof. Gustafsson is an associate editor of IEEE TRANSACTIONS ON SIGNAL PROCESSING.
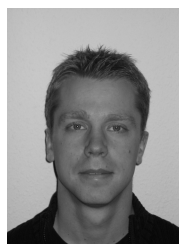
Per-Johan Nordlund received his MSc degree in Electrical Engineering in 1997 and his Licentiate degree of Engineering degree in automatic control in 2002, both from Linköping University, Linköping, Sweden. He is currently working as manager for the navigation department at Saab Aerospace, Linköping, Sweden.