A novel solution to the filtering problem for mixed linear/nonlinear state-space models: turbo filtering

Giorgio M. Vitetta, *Senior Member*, IEEE, Francesco Montorsi, *Member*, IEEE, Matteo Sola, *Member*, IEEE and Marco Casparriello [1]

Technical Report
University of Modena and Reggio Emilia, Italy

[1]Giorgio M. Vitetta, Matteo Sola and Marco Casparriello are with the Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia, Modena, Italy (e-mail: giorgio.vitetta@unimore.it, matteo.sola@unimore.it, marco.casparriello@unimore.it).
Francesco Montorsi is with the company Empirix Inc. (Modena, Italy) (e-mail: francesco.montorsi@gmail.com).

Abstract

In this manuscript the application of a factor graph approach to the filtering problem for a mixed linear/nonlinear state-space model is investigated. In particular, after developing a factor graph for the considered model, a novel approximate recursive technique for solving such a problem is derived applying the sum-product algorithm and a specific scheduling procedure for message passing to this graph. Then, the application of this technique, dubbed turbo filtering for its conceptual resemblance with turbo decoding of concatenated channel codes, to linear Gaussian systems is investigated. Numerical results for specific state-space models show that turbo filtering can achieve a good performance-complexity tradeoff.

0.1 Introduction

The nonlinear filtering problem consists of inferring the posterior distribution of the hidden state of a nonlinear dynamic system from a set of past and present measurements [7]. It is well known that, if a nonlinear dynamic system can be described by a *hidden Markov model* (HMM), a general sequential procedure, based on the Bayes' rule and known as *Bayesian filtering*, can be easily derived for recursively computing the posterior distribution of the system current state [7]. Unluckily, Bayesian filtering is analytically tractable in few cases for the following two reasons [18]: a) one of the two update steps it consists of requires multidimensional integration which, in most cases, does not admit a closed form solution; b) the functional form of the required posterior distribution may not be preserved over successive recursions. For this reason, sequential techniques employed in practice are based on various analytical approximations and, consequently, generate a functional approximation of the desired distribution. Such techniques are commonly divided into *local* and *global* methods on the basis of the way posterior distributions are approximated [15, 21, 26]. Local methods, like *extended Kalman filtering* [23] and *unscented filtering* [22] are computationally efficient but may suffer from the problem of error accumulation over time. On the contrary, global methods, like sequential Monte Carlo methods [19, 20] (also known as *particle filtering*, PF, methods [1, 2]) and *point mass filtering* [27, 26] may achieve high accuracy at the price, however, of an unmanageable complexity and numerical problems when the dimension of system state is large [3]. These considerations have motivated various research activities focused on the development of novel Bayesian filters that can achieve high accuracy under given computational constraints. Significant results in this research area concern the use of the new representations for complex distributions, like *belief condensation filtering* [15], and the development of novel filtering techniques combining local and global methods, like *marginalized particle filtering* [6, 17] and other methods originating from it [21, 28]. It is also worth mentioning that marginalized particle filtering and its variants apply to systems represented by *mixed linear/nonlinear models*. In fact, in these methods the availability of a 'linear' portion in system state (i.e., of one or more state variables appearing linearly in system dynamics) is exploited; this allows to combine a global method (e.g., particle filtering) with a local technique (e.g., Kalman filtering).

In this manuscript the problem of recursive Bayesian filtering for mixed linear/nonlinear models is revisited from a different perspective. In fact, first a *factor graph* (FG) approach [5, 9] is employed to develop a graphical representation of Bayesian filtering for this class of models. Then, it is shown that: a) applying the *sum-product algorithm* (SPA) [5, 9], together with a specific message scheduling procedure, to this representation results in a novel type of iterative filtering technique, called *turbo filtering*; b) in the specific case of Gaussian systems turbo filtering algorithms can be implemented in a computationally efficient way by combining a global technique (namely, particle filtering) with a local technique (in particular, a variant of Kalman filtering). It is important to point that, even if turbo filtering for linear Gaussian systems combines local and global approximations similar to that employed by marginalized particle filtering, it is characterized by a substantially different structure. In fact, marginalized particle filtering requires multiple parallel conditional filtering updates, since sufficient statistics are required for each particle trajectory, and these updates are carried out once in each recursion. On the contrary, turbo filtering employs a single Kalman filter, run as many times as the number of accomplished iterations in each recursion, with the aim of progressively refining state estimates. For this reason, it can provide a significant gain in terms of computational complexity with respect to MPF when the size of the state vector for the considered system is large; our simulation results, referring to specific systems, evidence that this result is achieved at the price of a small performance loss.

It is worth pointing out that turbo filtering has been mainly inspired by the following ideas:

- A mixed linear/nonlinear Markov system can be represented as the *concatenation* of two interacting subsystems, one governed by linear dynamics, the other accounting for a nonlinear behavior; conceptually related (finite state) Markov models can be found in data communications and, in particular, in concatenated channel coding (e.g., turbo coding [30]) and in coded transmissions over inter-symbol interference channels for which *turbo decoding methods* [13, 30] and *turbo equalization techniques* [25] have been developed, respectively[1].

- Factor graphs play an essential role in the derivation and interpretation of turbo decoding and equalization [5] (for instance, turbo decoding techniques emerge in a natural fashion from graphical models of codes [10]).

- Both Kalman filtering and particle methods can be viewed as *message passing procedures on factor graphs*, as shown in [5, 9] and [4], respectively.

However, multiple connections of our approach with previous work on *Bayesian inference on graphical models* and *variational Bayes methods* [29] can be also established. In fact, the relevant principle of progressively refining distributional approximations through multiple iterations has been also exploited in previous work about Bayesian inference on dynamic systems, and, in particular, in *expectation propagation* in Bayesian networks [24] and in *variational Bayesian filtering* [21].

The remaining part of this manuscript is organized as follows. A description of the mathematical model for the considered mixed linear/nonlinear system is illustrated in Section 0.2. In Section 0.3 it is proved how the filtering problem for this system can be described by a proper FG which, unluckily, is not cycle free. In Section 0.4 it is shown how the turbo filtering method emerges in natural fashion from applying SPA and proper message scheduling strategies (i.e., a *loopy belief propagation* strategy) to this FG; moreover, specific implementations of this method for the class of linear Gaussian systems are derived and the computational complexity of one of them is analysed in detail. Turbo filtering for specific linear Gaussian systems are compared with other filtering techniques in terms of performance and complexity in Section 0.5. Finally, some conclusions are offered in Section 0.6.

Notations: The probability density function (pdf) of a random vector $\mathbf{R}$ evaluated at point $\mathbf{r}$ is denoted $f(\mathbf{r})$; $\mathcal{N}(\mathbf{r}; \eta, \mathbf{\Sigma})$ represents the the pdf of a Gaussian random vector $\mathbf{R}$ characterized by the mean η and covariance matrix $\mathbf{\Sigma}$ evaluated at point $\mathbf{r}$; x_i denotes the i-th element of the vector $\mathbf{x}$.

0.2 System Model

In the following we focus on a discrete-time mixed linear/nonlinear Markov system [6], whose *hidden state* in the t-th interval is represented by a D-dimensional real vector $\mathbf{x}_t \triangleq [x_{0,t}, x_{1,t}, ..., x_{D-1,t}]^T$. We assume that this vector can be partitioned as

$$\mathbf{x}_t = \left[\left(\mathbf{x}_t^{(L)} \right)^T, \left(\mathbf{x}_t^{(N)} \right)^T \right]^T, \tag{1}$$

where $\mathbf{x}_t^{(L)} \triangleq [x_{0,t}^{(L)}, x_{1,t}^{(L)}, ..., x_{D_L-1,t}^{(L)}]^T$ ($\mathbf{x}_t^{(N)} \triangleq [x_{0,t}^{(N)}, x_{1,t}^{(N)}, ..., x_{D_N-1,t}^{(L)}]^T$) is the so called *linear* (*nonlinear*) *component* of $\mathbf{x}_t$ (1), with $D_L < D$ ($D_N = D - D_L$). This partitioning of $\mathbf{x}_t$ is based on the following simple rule. First, $\mathbf{x}_t^{(L)}$ is identified as that portion of $\mathbf{x}_t$ (1) characterized by the following two properties:

[1]Note that these classes of algorithms can be seen as specific applications of the so called *turbo principle* [11]

1. *Conditionally linear dynamics* - This means that its update equation, conditioned on $\mathbf{x}_t^{(N)}$, is linear; in other words, we have that

$$\mathbf{x}_{t+1}^{(L)} = \mathbf{A}_t^{(L)}\mathbf{x}_t^{(L)} + \mathbf{f}_t^{(L)}\left(\mathbf{x}_t^{(N)}\right) + \mathbf{w}_t^{(L)}, \tag{2}$$

   where $\mathbf{f}_t^{(L)}(\mathbf{x})$ is a time-varying D_L-dimensional *differentiable* function, $\mathbf{A}_t^{(L)}$ is a time-varying $D_L \times D_L$ real matrix and $\mathbf{w}_t^{(L)}$ is the t-th element of the process noise sequence $\left\{\mathbf{w}_l^{(L)}\right\}$, which consists of D_L- dimensional independent and identically distributed noise vectors.

2. *Conditionally linear* (or *almost linear*) dependence of all the available measurements on it - In other words, these quantities, conditioned on $\mathbf{x}_t^{(N)}$, exhibit a linear dependence on $\mathbf{x}_t^{(L)}$ (additional details about this model feature are provided below).

Then, $\mathbf{x}_t^{(N)}$ is generated by putting together all the components of $\mathbf{x}_t$ that do not belong to $\mathbf{x}_t^{(L)}$. For this reason, generally speaking, this vector is characterized by at least one of the following two properties:

a) *Nonlinear dynamics* - In particular, the update equation

$$\mathbf{x}_{t+1}^{(N)} = \mathbf{f}_t^{(N)}\left(\mathbf{x}_t^{(N)}\right) + \mathbf{A}_t^{(N)}\mathbf{x}_t^{(L)} + \mathbf{w}_t^{(N)} \tag{3}$$

   is assumed, where $\mathbf{A}_t^{(N)}$ is a time-varying $D_L \times D_N$ real matrix, $\mathbf{f}_t^{(N)}(\mathbf{x})$ is a time-varying D_N-dimensional *differentiable* function and $\mathbf{w}_t^{(N)}$ is the t-th element of the process noise sequence $\left\{\mathbf{w}_l^{(N)}\right\}$, which consists of D_N-dimensional independent and identically distributed noise vectors and is statistically independent of $\left\{\mathbf{w}_l^{(L)}\right\}$.

b) *A nonlinear dependence of all the available measurements on it* (further details are provided below).

In the following Section we focus on the so-called *filtering problem*, which concerns the evaluation of the posterior *probability density function* (pdf) $f(\mathbf{x}_t | \mathbf{y}_{1:t})$ at an instant $t > 1$, given a) the pdf $f(\mathbf{x}_1)$ referring to the system state in the first observation interval and b) the $t \cdot P$-dimensional *measurement* vector

$$\mathbf{y}_{1:t} = \left[\mathbf{y}_1^T, \mathbf{y}_2^T, ..., \mathbf{y}_t^T\right]^T, \tag{4}$$

where $\mathbf{y}_l \triangleq [y_0, y_1, ..., y_{P-1}]^T$ denotes the P-dimensional real vector collecting all the noisy measurements available at time t. As already mentioned above, the measurement vector $\mathbf{y}_l$ exhibits a *linear* (*nonlinear*) dependence on $\mathbf{x}_t^{(L)}$ ($\mathbf{x}_t^{(N)}$); for this reason, in the following it is assumed that

$$\mathbf{y}_l = \mathbf{h}_l\left(\mathbf{x}_t^{(N)}\right) + \mathbf{C}_l\,\mathbf{x}_t^{(L)} + \mathbf{e}_l, \tag{5}$$

where $\mathbf{C}_l$ is a time-varying $P \times D_L$ real matrix, $\mathbf{h}_l\left(\mathbf{x}_t^{(N)}\right)$ is a time-varying P-dimensional function (further details about this function are provided below) and $\mathbf{e}_l$ the l-th element of the measurement noise sequence $\{\mathbf{e}_l\}$ consisting of P-dimensional independent and identically distributed noise vectors. As it will become clearer in the following Section, it is useful to partition the vector $\mathbf{y}_l$ (5) as

$$\mathbf{y}_l = \left[\left(\mathbf{y}_l^{(L)}\right)^T, \left(\mathbf{y}_l^{(NL)}\right)^T, \left(\mathbf{y}_l^{(N)}\right)^T\right]^T, \tag{6}$$

where the P_L-dimensional vector $\mathbf{y}_l^{(L)} \triangleq [y_{0,l}^{(L)}, y_{1,l}^{(L)}, ..., y_{P_L-1,l}^{(L)}]^T$ (P_N-dimensional vector $\mathbf{y}_l^{(N)} \triangleq [y_{0,t}^{(N)}, y_{1,t}^{(N)}, ..., y_{P_N-1,l}^{(N)}]^T$) depends on $\mathbf{x}_l^{(L)}$ ($\mathbf{x}_l^{(N)}$) only, whereas, generally speaking, the P_{NL}-dimensional vector $\mathbf{y}_l^{(NL)} \triangleq [y_{0,l}^{(NL)}, y_{1,l}^{(NL)}, ..., y_{P_{NL}-1,l}^{(NL)}]^T$ (with $P_{NL} = P - P_N - P_L$) exhibits a *mixed* dependence, i.e. a dependence on both $\mathbf{x}_l^{(L)}$ and $\mathbf{x}_l^{(N)}$; in other words, we have that[2]

$$\mathbf{y}_l^{(L)} = \mathbf{C}_l^{(L)} \mathbf{x}_l^{(L)} + \mathbf{e}_l^{(L)}, \tag{7}$$

$$\mathbf{y}_l^{(NL)} = \mathbf{h}_l^{(NL)} \left(\mathbf{x}_l^{(N)}\right) + \mathbf{C}_l^{(NL)} \mathbf{x}_l^{(L)} + \mathbf{e}_l^{(NL)} \tag{8}$$

and

$$\mathbf{y}_l^{(N)} = \mathbf{h}_l^{(N)} \left(\mathbf{x}_l^{(N)}\right) + \mathbf{e}_l^{(N)}, \tag{9}$$

where $\mathbf{h}_l^{(N)} \left(\mathbf{x}_l^{(N)}\right)$ ($\mathbf{h}_l^{(NL)} \left(\mathbf{x}_l^{(N)}\right)$) is a P_N-dimensional (P_{NL}-dimensional) function, and $\mathbf{C}_l^{(L)}$ ($\mathbf{C}_l^{(NL)}$) is a time-varying $P_L \times D_L$ ($P_{NL} \times D_L$) real matrix, and $\left\{\mathbf{e}_l^{(L)}\right\}$, $\left\{\mathbf{e}_l^{(N)}\right\}$ and $\left\{\mathbf{e}_l^{(NL)}\right\}$ are mutually independent noise processes. Note that (7)-(9) implicitly rely on the assumptions that $\mathbf{e}_l$ results from the ordered concatenation of $\mathbf{e}_l^{(L)}$, $\mathbf{e}_l^{(N)}$ and $\mathbf{e}_l^{(NL)}$, and that the matrix $\mathbf{C}_l$ and the function $\mathbf{h}_l \left(\mathbf{x}_l^{(N)}\right)$ are structured as

$$\mathbf{C}_l = \begin{bmatrix} \mathbf{C}_l^{(L)} & \mathbf{0}_{P_L \times D_N} \\ \mathbf{C}_l^{(NL)} & \mathbf{0}_{P_{NL} \times D_N} \\ \mathbf{0}_{P_N \times D_L} & \mathbf{0}_{P_N \times D_N} \end{bmatrix} \tag{10}$$

and

$$\mathbf{h}_l \left(\mathbf{x}_l^{(N)}\right) = \left[\mathbf{0}_{P_N}^T, \left(\mathbf{h}_l^{(NL)} \left(\mathbf{x}_l^{(N)}\right)\right)^T, \left(\mathbf{h}_l^{N} \left(\mathbf{x}_l^{(N)}\right)\right)^T\right]^T, \tag{11}$$

respectively, for any l; here $\mathbf{0}_N$ ($\mathbf{0}_{N \times M}$) represents a N-dimensional ($N \times M$) null vector (null matrix). In the following it is also assumed that $\mathbf{h}_l^{(NL)} \left(\mathbf{x}_l^{(N)}\right)$ is P_{NL}-dimensional *differentiable* function and that the measurement noise sequences $\{\mathbf{e}_{L,l}\}$, $\{\mathbf{e}_{N,l}\}$ and $\{\mathbf{e}_{NL,l}\}$ are mutually independent, and are independent of $\left\{\mathbf{w}_l^{(N)}\right\}$ and $\left\{\mathbf{w}_l^{(L)}\right\}$.

0.3 Representation of the Filtering Problem via Factor Graphs

Generally speaking, the *filtering problem* for a system described by the *Markov model* $f(\mathbf{x}_{t+1} | \mathbf{x}_t)$ and the *observation model* $f(\mathbf{y}_t | \mathbf{x}_t)$ concerns the computation of the conditional pdf $f(\mathbf{x}_t | \mathbf{y}_{1:t})$ (i.e., the posterior pdf of the state $\mathbf{x}_t$ given the measurement vector $\mathbf{y}_{1:t}$ (4)) for $t \geq 1$ by means of a recursive procedure [7]. It is well known that, if the pdf $f(\mathbf{x}_1)$ is known, a general *Bayesian recursive procedure* can be employed; its l-th recursion (with $l = 1, 2, ..., t$) consists of the following two steps:

[2] Note that this partitioning of the measurement vector $\mathbf{y}_l$ (6) has been inspired by the way the noisy data available at the output of a communication channel are processed by a turbo decoder for a couple of parallely concatenated channel codes. In fact, in that case the noisy data are partitioned in three blocks, one common to the two decoders for the constituent channel codes, the other two feeding each a distinct decoder (e.g., see [30, Fig. 6]).

- *Measurement update* - Given the pdf $f\left(\mathbf{x}_l \left|\mathbf{y}_{1:(l-1)}\right.\right)$ (evaluated in the last step of the previous recursion[3]) and the present measurement vector $\mathbf{y}_l$, the conditional pdf $f\left(\mathbf{x}_l \left|\mathbf{y}_{1:l}\right.\right)$ is computed as

$$f\left(\mathbf{x}_l \left|\mathbf{y}_{1:l}\right.\right) = f\left(\mathbf{x}_l \left|\mathbf{y}_{1:(l-1)}\right.\right) f\left(\mathbf{y}_l \left|\mathbf{x}_l\right.\right) \frac{1}{f\left(\mathbf{y}_l \left|\mathbf{y}_{1:(l-1)}\right.\right)}, \tag{12}$$

  where

$$f\left(\mathbf{y}_l \left|\mathbf{y}_{1:(l-1)}\right.\right) = \int f\left(\mathbf{y}_l \left|\mathbf{x}_l\right.\right) f\left(\mathbf{x}_l \left|\mathbf{y}_{1:(l-1)}\right.\right) d\mathbf{x}_l. \tag{13}$$

- *Time update* - The pdf $f\left(\mathbf{x}_l \left|\mathbf{y}_{1:l}\right.\right)$ (12) generated by the measurement update is exploited to compute the pdf

$$f\left(\mathbf{x}_{l+1} \left|\mathbf{y}_{1:l}\right.\right) = \int f\left(\mathbf{x}_{l+1} \left|\mathbf{x}_l\right.\right) f\left(\mathbf{x}_l \left|\mathbf{y}_{1:l}\right.\right) d\mathbf{x}_l, \tag{14}$$

  which represents a *prediction* about the future state $\mathbf{x}_{l+1}$.

It is important to point out that: 1) the term $1/f\left(\mathbf{y}_l \left|\mathbf{y}_{1:(l-1)}\right.\right)$ appearing in the *right hand side* (RHS) of (12) represents a *normalization factor*; 2) both (13) and (14) require integration with respect to $\mathbf{x}_l$ and this may represent a formidable task when the dimensionality of $\mathbf{x}_l$ is large and/or the pdfs appearing in the integrands are not Gaussian; 3) this recursive procedure lends itself to be efficiently represented by a *message passing algorithm* over a proper *factor graph* (FG) [5]. The derivation of the FG mentioned in the last point rely on the fact that the a posteriori pdf $f\left(\mathbf{x}_t \left|\mathbf{y}_{1:t}\right.\right)$ has the *same* FG as the joint pdf $f\left(\mathbf{x}_t, \mathbf{y}_{1:t}\right)$ (see [5, Sec. II]) and the last pdf can be computed recursively through a procedure similar to that illustrated above, but in which the measurement update (12) and the time update (14) are replaced by

$$f\left(\mathbf{x}_l, \mathbf{y}_{1:l}\right) = f\left(\mathbf{x}_l, \mathbf{y}_{1:(l-1)}\right) f\left(\mathbf{y}_l \left|\mathbf{x}_l\right.\right), \tag{15}$$

and

$$f\left(\mathbf{x}_{l+1}, \mathbf{y}_{1:l}\right) = \int f\left(\mathbf{x}_{l+1} \left|\mathbf{x}_l\right.\right) f\left(\mathbf{x}_l, \mathbf{y}_{1:l}\right) d\mathbf{x}_l, \tag{16}$$

respectively, so that the evaluation of the above mentioned normalization factor is no more required. In fact, eqs. (15) and (16) involve only *products of pdfs* and *sums* (i.e., integration) *of products*, so that they can be represented by means of the FG enclosed in the dashed rectangle of Fig. 1 (where, following [5], a simplified notation is used for the involved pdfs). Since this FG is *cycle free*, the pdf $f\left(\mathbf{x}_t, \mathbf{y}_{1:t}\right)$ can be evaluated applying the well known SPA to it, i.e. developing a proper mechanism for passing probabilistic messages along this FG (the flow of messages is indicated by red arrows in Fig. 1). In fact, if the input message $m_{in,l}\left(\mathbf{x}_l\right) = f\left(\mathbf{x}_l, \mathbf{y}_{1:(l-1)}\right)$ enters this FG, the message going out of the *equality node* is given by [5]

$$m_{y,l}\left(\mathbf{x}_l\right) = m_{in,l}\left(\mathbf{x}_l\right) f\left(\mathbf{y}_l \left|\mathbf{x}_l\right.\right), \tag{17}$$

so that $m_{y,l}\left(\mathbf{x}_l\right) = f\left(\mathbf{x}_l, \mathbf{y}_{1:l}\right)$ (see (15)); then, the message emerging from the *function node* referring to the pdf $f\left(\mathbf{x}_{l+1} \left|\mathbf{x}_l\right.\right)$ is given by [5]

$$m_{out,l}\left(\mathbf{x}_{l+1}\right) = \int f\left(\mathbf{x}_{l+1} \left|\mathbf{x}_l\right.\right) m_{y,l}\left(\mathbf{x}_l\right) d\mathbf{x}_l, \tag{18}$$

[3]Note that in the first recursion (i.e., for $l = 1$) $f\left(\mathbf{x}_l \left|\mathbf{y}_{1:(l-1)}\right.\right) = f\left(\mathbf{x}_1 \left|\mathbf{y}_{1:0}\right.\right) = f\left(\mathbf{x}_1\right)$ and $f\left(\mathbf{y}_l \left|\mathbf{y}_{1:(l-1)}\right.\right) = f\left(\mathbf{y}_1 \left|\mathbf{y}_{1:0}\right.\right) = f\left(\mathbf{y}_1\right)$, so that $f\left(\mathbf{x}_1 \left|\mathbf{y}_1\right.\right) = f\left(\mathbf{x}_1\right) f\left(\mathbf{y}_1 \left|\mathbf{x}_1\right.\right) / f\left(\mathbf{y}_1\right)$.

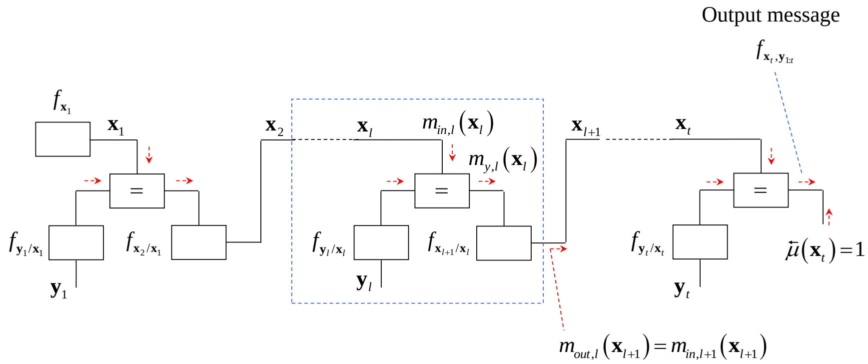

Figure 1: Factor graph for the evaluation of the pdf $f(\mathbf{x}_t, \mathbf{y}_{1:t})$ (the subgraph representing (15) and (16) is enclosed in the dashed rectangle). The SPA message flow is indicated by red arrows.

so that $m_{out,l}(\mathbf{x}_{l+1}) = f(\mathbf{x}_{l+1} | \mathbf{y}_{1:l}) = m_{in,l+1}(\mathbf{x}_{l+1})$ (see (16)). Consequently, the pdf $f(\mathbf{x}_t, \mathbf{y}_{1:t})$ (and, up to a scale factor, the pdf $f(\mathbf{x}_t | \mathbf{y}_{1:t})$) results from the application of the SPA to the overall FG shown in Fig. 1 and originating from the ordered concatenation of multiple subgraphs, each structured like the one contained in the rectangular box of the same figure. In this case, the flow of messages produced by the SPA proceeds from left to right, i.e. the pdf $f(\mathbf{x}_t, \mathbf{y}_{1:t})$ is generated by a *forward only* message passing. In principle, the desired pdf $f(\mathbf{x}_t, \mathbf{y}_{1:t})$ is computed as the product between two messages, one for each direction, reaching the rightmost edge of the FG; one of the incoming messages for that edge (denoted $\overleftarrow{\mu}(\mathbf{x}_t)$ in Fig. 1), however, is the constant function $\overleftarrow{\mu}(\mathbf{x}_t) = 1$.

Unluckily, as the size D of $\mathbf{x}_t$ (1) gets large, the computational burden associated with (12)-(14) becomes unmanageable. In principle, a substantial complexity reduction could be achieved by *decoupling*[4] the filtering problem for $\mathbf{x}_t^{(L)}$ from that for $\mathbf{x}_t^{(N)}$, i.e. the evaluation of $f\left(\mathbf{x}_t^{(L)} | \mathbf{y}_{1:t}\right)$ from that of $f\left(\mathbf{x}_t^{(N)} | \mathbf{y}_{1:t}\right)$. In fact, this approach potentially provides the following two benefits: a) a given filtering problem is turned into a couple of filtering problems of smaller dimensionality and b) some form of computationally efficient standard filtering (e.g., Kalman or extended Kalman filtering) can be hopefully exploited for the linear portion $\mathbf{x}_t^{(L)}$ of the state vector $\mathbf{x}_t$. These fundamental ideas have been exploited in devising *marginalized particle filtering* [2, 6]; as it will become clearer in the following Section, they are also employed in the derivation of turbo filtering, which results, however, from the application of different theoretical tools to the considered filtering problem. Before analysing this derivation, the measurement and state models on which turbo filtering relies need to be clearly defined; for this reason, these models are analysed in detail in the following part of this Section. To begin, let us concentrate on the models involved in the filtering problem for $\mathbf{x}_t^{(L)}$, i.e. the evaluation $f\left(\mathbf{x}_t^{(L)} | \mathbf{y}_{1:t}\right)$, under the assumption that the nonlinear portion $\mathbf{x}_l^{(N)}$ of the system state is *known* for

[4]Note that the coupling of the filtering problem for $\mathbf{x}_l^{(L)}$ with that for $\mathbf{x}_l^{(N)}$ is due not only to the structure of the update equations (2) and (3), but also to the measurement vector $\mathbf{y}_l^{(NL)}$ (8), that exhibits a mixed dependence on the two components of the state vector $\mathbf{x}_l$ (1).

any l. In this case, the evaluation of the pdf $\mathbf{x}_l^{(L)}$ can benefit not only from the knowledge of $\mathbf{y}_l^{(L)}$ (7), but also from that of a) the new measurement (see (8))

$$\tilde{\mathbf{y}}_l^{(L)} \triangleq \mathbf{y}_l^{(NL)} - \mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right) = \mathbf{C}_l^{(NL)}\mathbf{x}_l^{(L)} + \mathbf{e}_l^{(NL)} \tag{19}$$

and b) the quantity (see (3))

$$\mathbf{z}_l^{(L)} \triangleq \mathbf{x}_{l+1}^{(N)} - \mathbf{f}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right) = \mathbf{A}_l^{(N)}\mathbf{x}_l^{(L)} + \mathbf{w}_l^{(N)}, \tag{20}$$

which can be interpreted as a *pseudo-measurement* [6], since it does not originate from real measurements, but from the constraints expressed by the state equation (3). This leads to considering the *overall observation model*

$$\begin{aligned} & f\left(\mathbf{y}_l^{(L)}, \tilde{\mathbf{y}}_l^{(L)}, \mathbf{z}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) \\ & = f\left(\mathbf{y}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) f\left(\tilde{\mathbf{y}}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) f\left(\mathbf{z}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) \end{aligned} \tag{21}$$

for $\mathbf{x}_l^{(L)}$, where

$$f\left(\mathbf{y}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) = f\left(\mathbf{e}_l^{(L)}\right)\Big|_{\mathbf{e}_l^{(L)} = \mathbf{y}_l^{(L)} - \mathbf{C}_l^{(L)}\mathbf{x}_l^{(L)}}, \tag{22}$$

$$f\left(\tilde{\mathbf{y}}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) = f\left(\mathbf{e}_l^{(NL)}\right)\Big|_{\mathbf{e}_l^{(NL)} = \tilde{\mathbf{y}}_l^{(L)} - \mathbf{C}_l^{(NL)}\mathbf{x}_l^{(L)}} \tag{23}$$

and

$$f\left(\mathbf{z}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) = f\left(\mathbf{w}_l^{(N)}\right)\Big|_{\mathbf{w}_l^{(N)} = \mathbf{z}_l^{(L)} - \mathbf{A}_l^{(N)}\mathbf{x}_l^{(L)}}. \tag{24}$$

If the observation model (21) and the *state model* (see (2))

$$\begin{aligned} & f\left(\mathbf{x}_{l+1}^{(L)} \middle| \mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) \\ & = f_{\mathbf{w}^{(L)}}\left(\mathbf{x}_{l+1}^{(L)} - \mathbf{f}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right) - \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)}\right) \end{aligned} \tag{25}$$

are adopted for $\mathbf{x}_l^{(L)}$, the graph shown in Fig. 2 can be used, similarly as the FG shown in the dashed rectangle of Fig. 1, to describe a *new recursion* leading to the evaluation of $f\left(\mathbf{x}_t^{(L)} \middle| \mathbf{y}_{1:t}^{(L)}, \tilde{\mathbf{y}}_{1:t}^{(L)}, \mathbf{z}_{1:t}^{(L)}\right)$, under the assumption that the couple $\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right)$ is known for any l. Is important to point out that that:

- The graph shown in Fig. 2 is not a standard FG, but a mixed one, since it includes two subsystems[5] (represented by dashed rectangles) which do not refer to density factorizations; in fact, they represent the transformations from $\mathbf{y}_l^{(NL)}$ to $\tilde{\mathbf{y}}_l^{(L)}$ and from the couple $\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right)$ to $\mathbf{z}_l^{(L)}$ (see (19) and (20), respectively). As it will become clear later, this has to be carefully kept into account when deriving message passing algorithms.

[5]The presence of *oriented edges* indicates the existence of constraints in the message flow occurring in the represented graph.

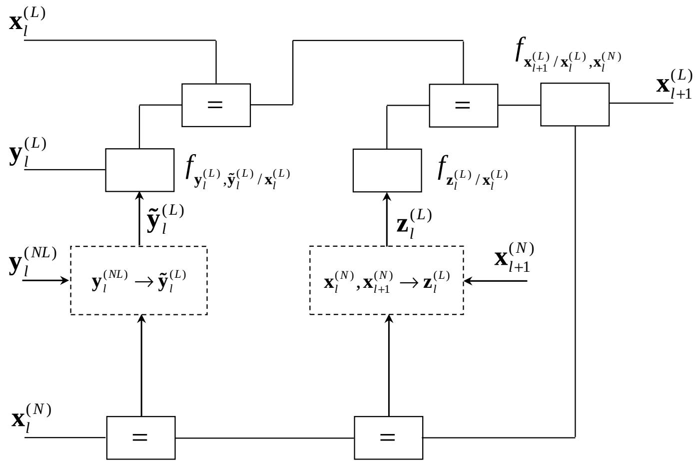

Figure 2: Factor graph associated with the observation model (21) and the state model (25).

- Generally speaking, the evaluation of the conditional pdf $f\left(\mathbf{z}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right)$ requires the knowledge of the joint pdf of $\mathbf{x}_{l+1}^{(N)}$ and $\mathbf{x}_l^{(N)}$ (see (20)), which can be evaluated as

$$f\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right) = f\left(\mathbf{x}_l^{(N)}\right) \int f\left(\mathbf{x}_{l+1}^{(N)} \middle| \mathbf{x}_l^{(N)}, \mathbf{x}_l^{(L)}\right) f\left(\mathbf{x}_l^{(L)}\right) d\mathbf{x}_l^{(L)}, \tag{26}$$

where $f\left(\mathbf{x}_{l+1}^{(N)} \middle| \mathbf{x}_l^{(N)}, \mathbf{x}_l^{(L)}\right)$ is given by (33) (see below).

The same line of reasoning can be followed for the models involved in the filtering problem for $\mathbf{x}_l^{(N)}$ under the assumption that the linear portion $\mathbf{x}_l^{(L)}$ of the system state is known for any l. In fact, in this case the new measurement (see (8))

$$\tilde{\mathbf{y}}_l^{(N)} \triangleq \mathbf{y}_l^{(NL)} - \mathbf{C}_l^{(NL)} \mathbf{x}_l^{(L)} = \mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right) + \mathbf{e}_l^{(NL)} \tag{27}$$

and the pseudo-measurement (see (2))

$$\mathbf{z}_l^{(N)} \triangleq \mathbf{x}_{l+1}^{(L)} - \mathbf{A}_l^{(L)} \mathbf{x}_l^{(L)} = \mathbf{f}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right) + \mathbf{w}_l^{(L)} \tag{28}$$

are defined. This leads to the *overall observation model*

$$f\left(\mathbf{y}_l^{(N)}, \tilde{\mathbf{y}}_l^{(N)}, \mathbf{z}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) = f\left(\mathbf{y}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) f\left(\tilde{\mathbf{y}}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) f\left(\mathbf{z}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) \tag{29}$$

for $\mathbf{x}_l^{(N)}$, where

$$f\left(\mathbf{y}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) = f\left(\mathbf{e}_l^{(N)}\right)\Big|_{\mathbf{e}_l^{(N)} = \mathbf{y}_l^{(N)} - \mathbf{h}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right)}, \tag{30}$$

$$f\left(\tilde{\mathbf{y}}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) = f\left(\mathbf{e}_l^{(NL)}\right)\Big|_{\mathbf{e}_l^{(NL)} = \tilde{\mathbf{y}}_l^{(N)} - \mathbf{h}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right)}, \tag{31}$$

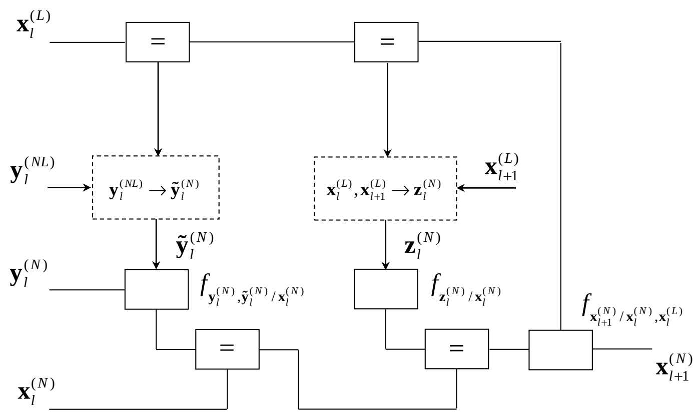

Figure 3: Factor graph associated with the observation model (29) and the state model (33).

and

$$f\left(\mathbf{z}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) = f\left(\mathbf{w}_l^{(L)}\right)\Big|_{\mathbf{w}_l^{(L)}=\mathbf{z}_l^{(N)}-\mathbf{f}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right)}. \tag{32}$$

If the observation model (29) and the *state model*

$$\begin{aligned} &f\left(\mathbf{x}_{l+1}^{(N)} \middle| \mathbf{x}_l^{(N)}, \mathbf{x}_l^{(L)}\right) \\ &= f_{\mathbf{w}_N}\left(\mathbf{x}_{l+1}^{(N)} - \mathbf{f}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right) - \mathbf{A}_l^{(N)}\mathbf{x}_l^{(L)}\right) \end{aligned} \tag{33}$$

are adopted for $\mathbf{x}_l^{(N)}$, the graph shown in Fig. 3 can be drawn, similarly as that shown in Fig. 2, to describe a new recursion leading to the evaluation of $f\left(\mathbf{x}_t^{(N)} \middle| \mathbf{y}_{1:t}^{(N)}, \tilde{\mathbf{y}}_{1:t}^{(N)}, \mathbf{z}_{1:t}^{(N)}\right)$, under the assumption that the couple $(\mathbf{x}_l^{(L)}, \mathbf{x}_{l+1}^{(L)})$ is known for any l. Note also that, similarly to what has been explained earlier about the conditional pdf $f\left(\mathbf{z}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right)$, the evaluation of the conditional pdf $f\left(\mathbf{z}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right)$ requires the knowledge of the joint pdf of $\mathbf{x}_{l+1}^{(L)}$ and $\mathbf{x}_l^{(L)}$ (see (28)), which can be evaluated as

$$f\left(\mathbf{x}_l^{(L)}, \mathbf{x}_{l+1}^{(L)}\right) = f\left(\mathbf{x}_l^{(L)}\right) \int f\left(\mathbf{x}_{l+1}^{(L)} \middle| \mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) f\left(\mathbf{x}_l^{(N)}\right) d\mathbf{x}_l^{(N)}, \tag{34}$$

where $f\left(\mathbf{x}_{l+1}^{(L)} \middle| \mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right)$ is given by (25).

Merging the FG shown in Fig. 2 with that of Fig. 3 (i.e., connecting the half edges labelled by the same state variables) produces the FG illustrated in Fig. 4. This graph, on which the l-th recursion of the algorithm proposed in this manuscript is based, is not cycle-free. For this reason, applying the SPA to it unavoidably leads to *iterative techniques producing approximate results* [9].

Finally, it is important to point out that the FG representation that can be adopted for deriving new filtering techniques in the considered problem is not unique; this is due to the fact that the overall FG of Fig. 1 can be partitioned into different subgraphs. For instance, the subgraphs appearing in Fig. 2 (Fig. 3) can be modified by removing the portion referring to the pdf $f\left(\mathbf{y}_l^{(N)}, \tilde{\mathbf{y}}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right)$ ($f\left(\mathbf{y}_l^{(L)}, \tilde{\mathbf{y}}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right)$), i.e., their leftmost portion, and by appending a new portion referring to the

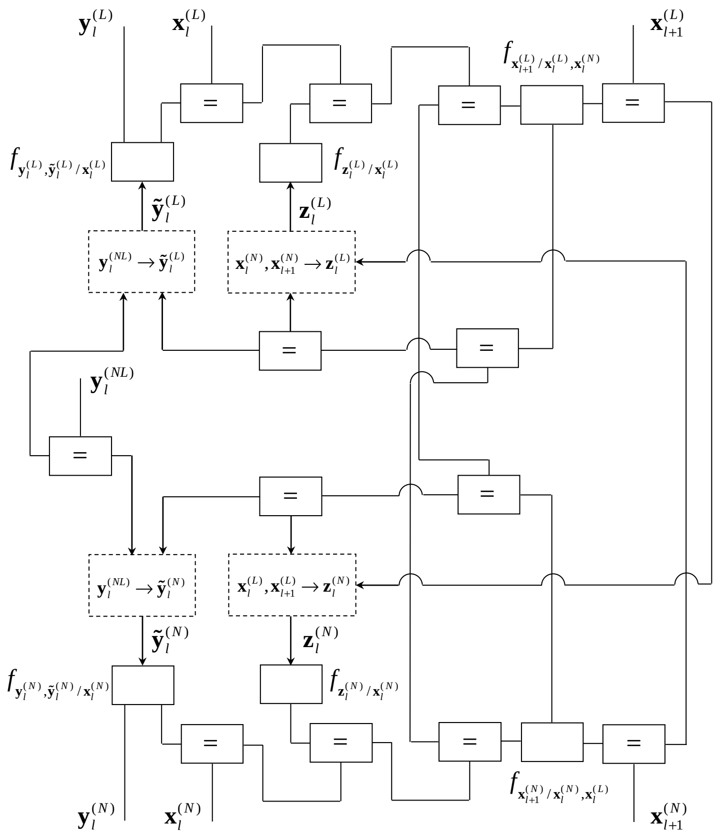

Figure 4: Factor graph resulting from the combination of the graphs shown in Figs. 2 and 3.

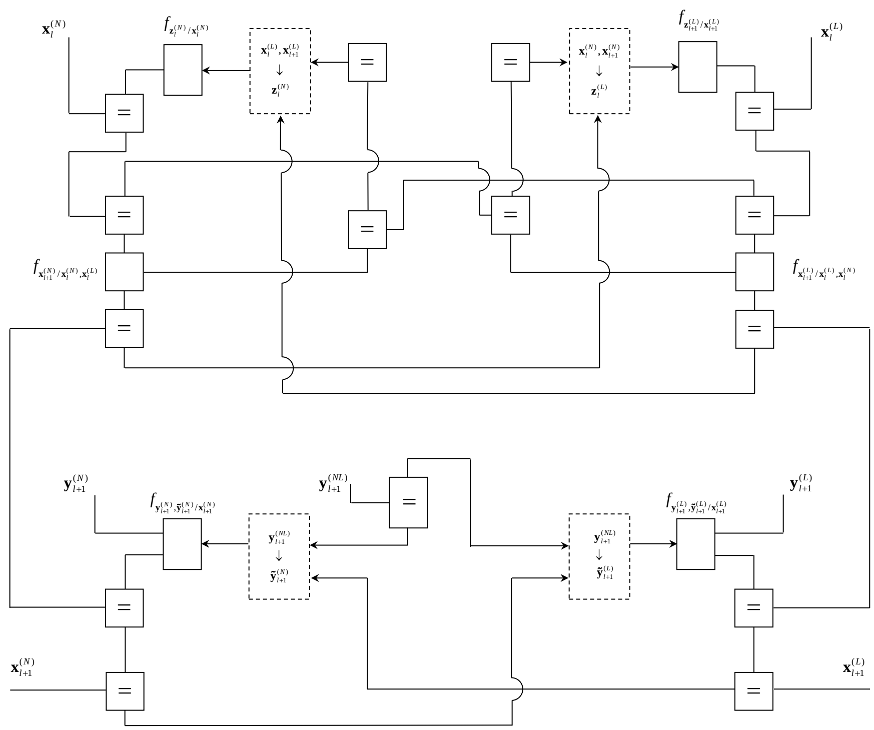

Figure 5: Possible FG representation that can be adopted as an alternative to the FG shown in 4.

pdf $f\left(\mathbf{y}_{l+1}^{(N)}, \tilde{\mathbf{y}}_{l+1}^{(N)} \middle| \mathbf{x}_{l+1}^{(N)}\right)$ ($f\left(\mathbf{y}_{l+1}^{(L)}, \tilde{\mathbf{y}}_{l+1}^{(L)} \middle| \mathbf{x}_{l+1}^{(L)}\right)$); then, merging these new subgraphs produces the FG shown in Fig. 5, which, as shown in the following, leads to different filtering techniques.

0.4 Message Passing in Turbo Filtering

In this Section the turbo filtering method is illustrated first in its most general form. Then, a specific implementation of this method, developed for the case of linear Gaussian systems, is devised and its computational complexity is assessed.

0.4.1 Derivation of the turbo filtering technique

As already mentioned in the previous Section, the turbo filtering technique developed in this manuscript results from the application of the SPA to the graph shown in Fig. 4. However, since this graph contains a cycle, different options should be considered for message scheduling [9, 10]. The first scheduling procedure adopted here has been inspired by marginalized particle filtering, as evidenced by the graph shown in Fig. 7; the message flow occurring in the k-th *iteration* within the l-th *recursion* is illustrated

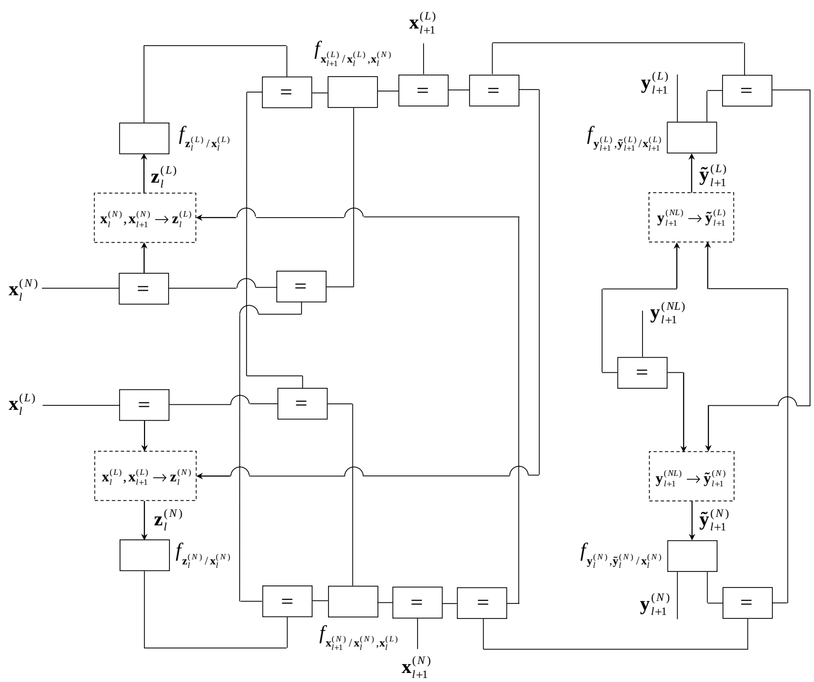

Figure 6: Factor graph resulting from a new combination of the graphs associated with the observation models (21) and (29), and the state models (25) and (33).

below for the considered graph (here $k = 1, 2, ..., N_{it}$, where N_{it} denotes the overall number of iterations accomplished in each recursion). In fact, in each iteration, on the basis of the message flow illustrated in this Figure, the following tasks are sequentially accomplished: 1) measurement update for $\mathbf{x}_l^{(N)}$; 2) measurement update for $\mathbf{x}_l^{(L)}$; 3) time update for both $\mathbf{x}_l^{(L)}$ and $\mathbf{x}_l^{(N)}$. It is important to point out that:

- The l-th recursion is fed by the messages $\vec{m}_l\left(\mathbf{x}^{(L)}\right)$ and $\vec{m}_l\left(\mathbf{x}^{(N)}\right)$, which have been generated in the previous (i.e., in the $(l-1)$-th) recursion and represent, up to a scale factor, approximations of $f\left(\mathbf{x}_l^{(L)}\left|\mathbf{y}_{1:(l-1)}\right.\right)$ and $f\left(\mathbf{x}_l^{(N)}\left|\mathbf{y}_{1:(l-1)}\right.\right)$, respectively. These messages represent the *a priori information* available to the l-th recursion. For this reason, generally speaking, at the beginning of the first recursion (i.e., for $l = 1$)
$$\vec{m}_1\left(\mathbf{x}^{(L)}\right) = \int f\left(\mathbf{x}_1\right) d\mathbf{x}_1^{(N)} \tag{35}$$
and
$$\vec{m}_1\left(\mathbf{x}^{(N)}\right) = \int f\left(\mathbf{x}_1\right) d\mathbf{x}_1^{(L)} \tag{36}$$
must be selected.

- The l-th recursion produces the new messages $\vec{m}_{l+1}\left(\mathbf{x}^{(L)}\right)$ and $\vec{m}_{l+1}\left(\mathbf{x}^{(N)}\right)$ at the end of its last iteration.

- In any iteration,
$$\overleftarrow{m}_{l+1}\left(\mathbf{x}^{(L)}\right) = \overleftarrow{m}_{l+1}\left(\mathbf{x}^{(N)}\right) = 1 \tag{37}$$
is assumed, since no information comes from the next recursion (it is like having a couple of half edges, one associated with $\mathbf{x}_{l+1}^{(L)}$, the other one with $\mathbf{x}_{l+1}^{(N)}$).

The proposed iterative technique consists of an *initialization procedure* followed by a *message passing procedure*. A detailed description of both procedures is provided below for the l-th recursion.

Message passing procedure - This procedure consists of an ordered sequence of steps, in which the messages labelling the arrows shown in Fig. 7 are evaluated; a general description of each step is provided below for the k-th iteration; additional mathematical details are provided in the next Paragraph for the case of a linear Gaussian system.

1. First *measurement update* for $\mathbf{x}_l^{(N)}$ - This step requires the knowledge of the message $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right)$ computed in the previous iteration; in fact, this message is exploited to generate, together with the measurement vector $\mathbf{y}_l^{(NL)}$ (8), the message $\vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(N)}\right)$ (see (27)). Then, the last message is employed, together with the measurement vector $\mathbf{y}_l^{(N)}$ (9) and a portion of the observation model (29) (see, in particular, (30) and (31)), to evaluate

$$\begin{aligned}\vec{m}_1^{(k)}\left(\mathbf{x}_l^{(N)}\right) &= \int \vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(N)}\right) f\left(\mathbf{y}_l^{(N)}, \tilde{\mathbf{y}}_l^{(N)}\left|\mathbf{x}_l^{(N)}\right.\right) d\tilde{\mathbf{y}}_l^{(N)} \\ &= f\left(\mathbf{y}_l^{(N)}\left|\mathbf{x}_l^{(N)}\right.\right) \int \vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(N)}\right) f\left(\tilde{\mathbf{y}}_l^{(N)}\left|\mathbf{x}_l^{(N)}\right.\right) d\tilde{\mathbf{y}}_l^{(N)}.\end{aligned} \tag{38}$$

Finally, multiplying $\vec{m}_1^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ by the *a priori information* $\vec{m}_l\left(\mathbf{x}^{(N)}\right)$ yields

$$\vec{m}_2^{(k)}\left(\mathbf{x}_l^{(N)}\right) = \vec{m}_1^{(k)}\left(\mathbf{x}_l^{(N)}\right) \vec{m}_l\left(\mathbf{x}^{(N)}\right). \tag{39}$$

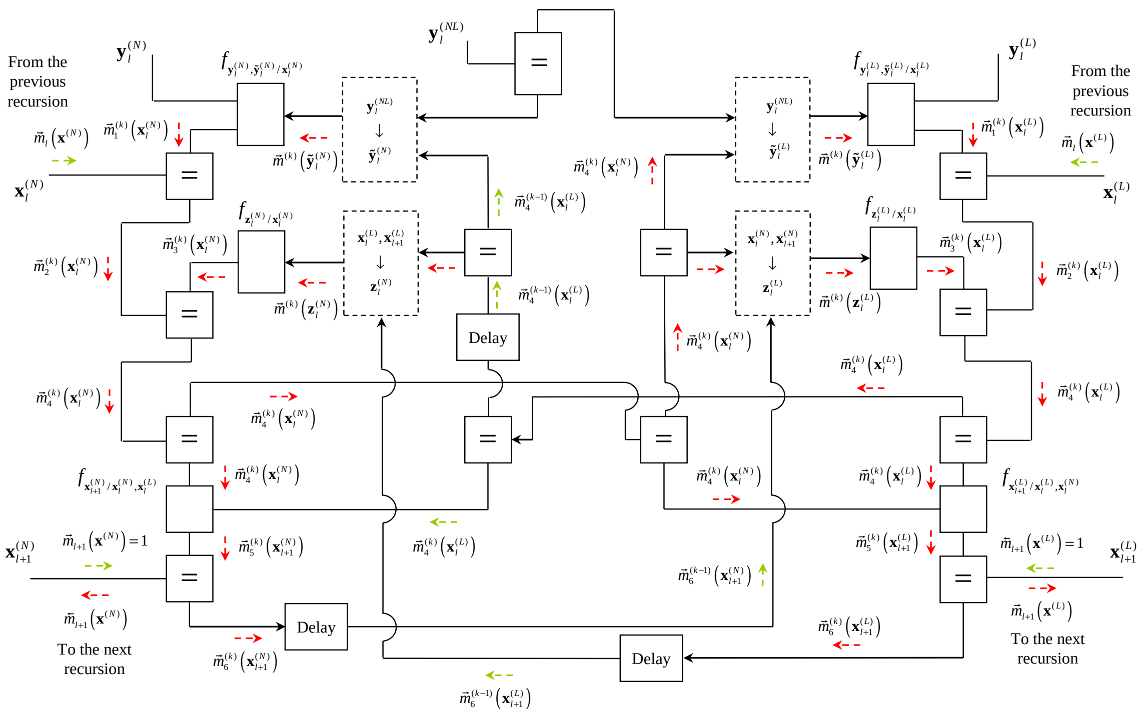

Figure 7: Message passing for the *turbo filtering* technique proposed in Paragraph 0.4.1. All the quantities appearing in this figure refer to the k-th iteration of the l-th recursion; the messages available at the beginning of the considered iteration are indicated by green arrows.

2. Second *measurement update* for $\mathbf{x}_l^{(N)}$ - In this step the messages $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right)$ and $\vec{m}_6^{(k-1)}\left(\mathbf{x}_{l+1}^{(L)}\right)$ can be exploited to evaluate any first order moment (e.g., the mean) of $\mathbf{z}_l^{(N)}$ (28)). However, the evaluation of any second order moment (i.e., the covariance matrix) or of the pdf of $\mathbf{z}_l^{(N)}$ (28) requires the knowledge of the joint pdf $\mathbf{x}_{l+1}^{(L)}$ and $\mathbf{x}_l^{(L)}$, which, in the considered iteration, can be approximated as (see (34))

$$\begin{aligned} & f\left(\mathbf{x}_l^{(L)}, \mathbf{x}_{l+1}^{(L)}\right) \cong f_{l,k}\left(\mathbf{x}_l^{(L)}, \mathbf{x}_{l+1}^{(L)}\right) \\ \triangleq\ & \vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right) \int f\left(\mathbf{x}_{l+1}^{(L)} \middle| \mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) \vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(N)}\right) d\mathbf{x}_l^{(N)}, \end{aligned} \tag{40}$$

so that $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(N)}\right)$ is also needed. Then, given $\vec{m}^{(k)}\left(\mathbf{z}_l^{(N)}\right)$ and the observation model (32), the message

$$\vec{m}_3^{(k)}\left(\mathbf{x}_l^{(N)}\right) = \int \vec{m}^{(k)}\left(\mathbf{z}_l^{(N)}\right) f\left(\mathbf{z}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) d\mathbf{z}_l^{(N)} \tag{41}$$

is computed. Finally, multiplying this message by $\vec{m}_2^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (39) gives

$$\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right) = \vec{m}_3^{(k)}\left(\mathbf{x}_l^{(N)}\right) \vec{m}_2^{(k)}\left(\mathbf{x}_l^{(N)}\right), \tag{42}$$

which represents the overall output of the two measurement updates referring to $\mathbf{x}_l^{(N)}$.

3. First *measurement update* for $\mathbf{x}_l^{(L)}$ - In this step message $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (42) is exploited to compute, together with the measurement vector $\mathbf{y}_l^{(NL)}$ (8), the message $\vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(L)}\right)$ (see (19)). Then, the last message is employed, together with the measurement vector $\mathbf{y}_l^{(L)}$ (7) and a portion of the observation model (21) (see, in particular, (22) and (23)), to evaluate

$$\begin{aligned} \vec{m}_1^{(k)}\left(\mathbf{x}_l^{(L)}\right) &= \int \vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(L)}\right) f\left(\mathbf{y}_l^{(L)}, \tilde{\mathbf{y}}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) d\tilde{\mathbf{y}}_l^{(L)} \\ &= f\left(\mathbf{y}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) \int \vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(L)}\right) f\left(\tilde{\mathbf{y}}_l^{(L)} \middle| \mathbf{x}_l^{(L)}\right) d\tilde{\mathbf{y}}_l^{(L)}. \end{aligned} \tag{43}$$

Finally, multiplying $\vec{m}_1^{(k)}\left(\mathbf{x}_l^{(L)}\right)$ by the *a priori information* $\vec{m}_l\left(\mathbf{x}^{(L)}\right)$ yields

$$\vec{m}_2^{(k)}\left(\mathbf{x}_l^{(L)}\right) = \vec{m}_1^{(k)}\left(\mathbf{x}_l^{(L)}\right) \vec{m}_l\left(\mathbf{x}^{(L)}\right). \tag{44}$$

4. Second *measurement update* for $\mathbf{x}_l^{(L)}$ - Similarly as step 2., in this step the messages $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ and $\vec{m}_6^{(k-1)}\left(\mathbf{x}_{l+1}^{(N)}\right)$ can be exploited to evaluate any first order moment (e.g., the mean) of $\mathbf{z}_l^{(L)}$ (20)). However, the evaluation of any second order moment (i.e., the covariance matrix) or of the pdf of $\mathbf{z}_l^{(L)}$ (20) requires the knowledge of the joint pdf $\mathbf{x}_{l+1}^{(N)}$ and $\mathbf{x}_l^{(N)}$, which, in the considered iteration, can be approximated as (see (26))

$$\begin{aligned} & f\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right) \cong \tilde{f}_{l,k}\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right) \\ \triangleq\ & \vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right) \int f\left(\mathbf{x}_{l+1}^{(N)} \middle| \mathbf{x}_l^{(N)}, \mathbf{x}_l^{(L)}\right) \vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right) d\mathbf{x}_l^{(L)}, \end{aligned} \tag{45}$$

so that $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right)$ is also required. Then, given $\vec{m}^{(k)}\left(\mathbf{z}_l^{(L)}\right)$ and the observation model (24), the message

$$\vec{m}_3^{(k)}\left(\mathbf{x}_l^{(L)}\right) = \int \vec{m}^{(k)}\left(\mathbf{z}_l^{(L)}\right) f\left(\mathbf{z}_l^{(L)}\middle|\mathbf{x}_l^{(L)}\right) d\mathbf{z}_l^{(L)} \tag{46}$$

is computed. Finally, multiplying the last message by $\vec{m}_2^{(k)}\left(\mathbf{x}_l^{(L)}\right)$ (44) yields

$$\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(L)}\right) = \vec{m}_3^{(k)}\left(\mathbf{x}_l^{(L)}\right) \vec{m}_2^{(k)}\left(\mathbf{x}_l^{(L)}\right), \tag{47}$$

which represents the overall output of the two measurement updates referring to $\mathbf{x}_l^{(L)}$.

5. *Time update* for $\mathbf{x}_l^{(N)}$ and $\mathbf{x}_l^{(L)}$ - In the time update step for $\mathbf{x}_l^{(N)}$ the message

$$\begin{aligned}\vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right) &= \int\int f\left(\mathbf{x}_{l+1}^{(N)}\middle|\mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) \\ &\cdot \vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right) \vec{m}_4^{(k)}\left(\mathbf{x}_l^{(L)}\right) d\mathbf{x}_l^{(L)} d\mathbf{x}_l^{(N)}.\end{aligned} \tag{48}$$

is computed. Then, multiplying this message by $\overleftarrow{m}_{l+1}\left(\mathbf{x}^{(N)}\right)$ (see (37)) gives

$$\begin{aligned}\vec{m}_6^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right) &= \vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right) \overleftarrow{m}_{l+1}\left(\mathbf{x}^{(N)}\right) \\ &= \vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right).\end{aligned} \tag{49}$$

Similarly, the time update step for $\mathbf{x}_l^{(L)}$ aims at evaluating the messages

$$\begin{aligned}\vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right) &= \int\int f\left(\mathbf{x}_{l+1}^{(L)}\middle|\mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) \\ &\cdot \vec{m}_4^{(k)}\left(\mathbf{x}_l^{(L)}\right) \vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right) d\mathbf{x}_l^{(L)} d\mathbf{x}_l^{(N)}\end{aligned} \tag{50}$$

and (see (37))

$$\begin{aligned}\vec{m}_6^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right) &= \vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right) \overleftarrow{m}_{l+1}\left(\mathbf{x}^{(L)}\right) \\ &= \vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right).\end{aligned} \tag{51}$$

Note that the time update for $\mathbf{x}_l^{(N)}$ and that for $\mathbf{x}_l^{(L)}$ can be carried out in parallel, since they do not interact.

6. *Stop or get ready for the next recursion* - After completing the previous step, the value of the iteration index k is compared with N_{it}. If $k < N_{it}$, the index k is increased by one and the next iteration starts (this means going back to step 1. of the message passing procedure). Otherwise, if $k = N_{it}$, the messages

$$\vec{m}_{l+1}\left(\mathbf{x}^{(N)}\right) = \vec{m}_5^{(N_{it})}\left(\mathbf{x}_{l+1}^{(N)}\right) \tag{52}$$

and

$$\vec{m}_{l+1}\left(\mathbf{x}^{(L)}\right) = \vec{m}_5^{(N_{it})}\left(\mathbf{x}_{l+1}^{(L)}\right) \tag{53}$$

are generated. In this case, if $l < t$, the recursion index l is increased by one and the initialization procedure for the next iteration is started; otherwise, if $l = t$, the turbo filtering procedure is over.

Initialization procedure - In principle, starting the message flow in the first iteration (corresponding to $l = 1$) requires the knowledge of the messages $\vec{m}_4^{(0)}\left(\mathbf{x}_l^{(L)}\right)$ (step 1 and 2), $\vec{m}_4^{(0)}\left(\mathbf{x}_l^{(N)}\right)$ (step 2), $\vec{m}_6^{(0)}\left(\mathbf{x}_{l+1}^{(L)}\right)$ (or, equivalently, $\vec{m}_5^{(0)}\left(\mathbf{x}_{l+1}^{(L)}\right)$; see step 2) and $\vec{m}_6^{(0)}\left(\mathbf{x}_{l+1}^{(N)}\right)$(or, equivalently, $\vec{m}_5^{(0)}\left(\mathbf{x}_{l+1}^{(N)}\right)$; see step 4). Note that, on the one hand, no information is available about $\mathbf{x}_{l+1}^{(L)}$ and $\mathbf{x}_{l+1}^{(N)}$ in the initialization; for this reason,

$$\vec{m}_6^{(0)}\left(\mathbf{x}_{l+1}^{(L)}\right) = \vec{m}_5^{(0)}\left(\mathbf{x}_{l+1}^{(L)}\right) = \vec{m}_6^{(0)}\left(\mathbf{x}_{l+1}^{(N)}\right) = \vec{m}_5^{(0)}\left(\mathbf{x}_{l+1}^{(N)}\right) = 1 \tag{54}$$

and

$$\vec{m}^{(1)}\left(\mathbf{z}_l^{(L)}\right) = \vec{m}^{(1)}\left(\mathbf{z}_l^{(N)}\right) = 1 \tag{55}$$

are selected. On the other hand, the only information available about $\mathbf{x}_l^{(L)}$ ($\mathbf{x}_l^{(N)}$) is expressed by $\vec{m}_l\left(\mathbf{x}^{(L)}\right)$ ($\vec{m}_l\left(\mathbf{x}^{(N)}\right)$) which, refers to the the prediction $\mathbf{x}_{l/l-1}^{(L)}$ ($\mathbf{x}_{l/l-1}^{(N)}$) of $\mathbf{x}_l^{(L)}$ ($\mathbf{x}_l^{(N)}$) based on past measurements (further details about this are provided in the following Paragraph), so that

$$\vec{m}_4^{(0)}\left(\mathbf{x}_l^{(L)}\right) = \vec{m}_l\left(\mathbf{x}^{(L)}\right) \tag{56}$$

and

$$\vec{m}_4^{(0)}\left(\mathbf{x}_l^{(N)}\right) = \vec{m}_l\left(\mathbf{x}^{(N)}\right) \tag{57}$$

are chosen.

0.4.2 Turbo filtering for linear Gaussian systems

Let us analyse now the application of the general procedure illustrated in the previous Paragraph to the specific class of *linear Gaussian systems*. For this reason, in this Paragraph we assume that: a) $\left\{\mathbf{w}_l^{(L)}\right\}$ ($\left\{\mathbf{w}_l^{(N)}\right\}$) is a Gaussian random process and all its elements have zero mean and covariance $\mathbf{C}_w^{(L)}$($\mathbf{C}_w^{(N)}$) for any l; b) $\{\mathbf{e}_l^{(L)}\}$, $\{\mathbf{e}_l^{(N)}\}$ and $\{\mathbf{e}_l^{(NL)}\}$ are Gaussian random processes having zero mean and covariance matrices $\mathbf{C}_e^{(L)}$, $\mathbf{C}_e^{(N)}$ and $\mathbf{C}_e^{(NL)}$, respectively, for any l. Under these assumptions, (22)-(25) can be rewritten as

$$f\left(\mathbf{y}_l^{(L)}\left|\mathbf{x}_l^{(L)}\right.\right) = \mathcal{N}\left(\mathbf{y}_l^{(L)}; \mathbf{C}_l^{(L)}\mathbf{x}_l^{(L)}, \mathbf{C}_e^{(L)}\right), \tag{58}$$

$$f\left(\tilde{\mathbf{y}}_l^{(L)}\left|\mathbf{x}_l^{(L)}\right.\right) = \mathcal{N}\left(\tilde{\mathbf{y}}_l^{(L)}; \mathbf{C}_l^{(NL)}\mathbf{x}_l^{(L)}, \mathbf{C}_e^{(NL)}\right), \tag{59}$$

$$f\left(\mathbf{z}_l^{(L)}\left|\mathbf{x}_l^{(L)}\right.\right) = \mathcal{N}\left(\mathbf{z}_l^{(L)}; \mathbf{A}_l^{(N)}\mathbf{x}_l^{(L)}, \mathbf{C}_w^{(N)}\right) \tag{60}$$

and

$$\begin{aligned} &f\left(\mathbf{x}_{l+1}^{(L)}\left|\mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right.\right) \\ &= \mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \mathbf{f}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right) + \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)}, \mathbf{C}_w^{(L)}\right), \end{aligned} \tag{61}$$

respectively. Similarly, (30)-(33) turn into

$$f\left(\mathbf{y}_l^{(N)}\left|\mathbf{x}_l^{(N)}\right.\right) = \mathcal{N}\left(\mathbf{y}_l^{(N)}; \mathbf{h}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right), \mathbf{C}_e^{(N)}\right), \tag{62}$$

$$f\left(\tilde{\mathbf{y}}_l^{(N)}\left|\mathbf{x}_l^{(N)}\right.\right)=\mathcal{N}\left(\tilde{\mathbf{y}}_l^{(N)};\mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right),\mathbf{C}_e^{(NL)}\right) \tag{63}$$

$$f\left(\mathbf{z}_l^{(N)}\left|\mathbf{x}_l^{(N)}\right.\right)=\mathcal{N}\left(\mathbf{z}_l^{(N)};\mathbf{f}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right),\mathbf{C}_w^{(L)}\right). \tag{64}$$

and

$$\begin{aligned} & f\left(\mathbf{x}_{l+1}^{(N)}\left|\mathbf{x}_l^{(N)},\mathbf{x}_l^{(L)}\right.\right) \\ & =\mathcal{N}\left(\mathbf{x}_{l+1}^{(N)};\mathbf{f}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right)+\mathbf{A}_l^{(N)}\mathbf{x}_l^{(L)},\mathbf{C}_w^{(N)}\right), \end{aligned} \tag{65}$$

respectively. As far as the representations of the messages $\vec{m}_l\left(\mathbf{x}^{(L)}\right)$ and $\vec{m}_l\left(\mathbf{x}^{(N)}\right)$ are concerned, the following choices have been made. The Gaussian model

$$\vec{m}_l\left(\mathbf{x}^{(L)}\right)=\mathcal{N}\left(\mathbf{x}_l^{(L)};\hat{\mathbf{x}}_{l/(l-1)}^{(L)},\mathbf{C}_{l/(l-1)}^{(L)}\right), \tag{66}$$

and the particle-based representation [7]

$$\vec{m}_l\left(\mathbf{x}^{(N)}\right)=\sum_{j=0}^{N_p-1}w_{l,j}\,\delta\left(\mathbf{x}_l^{(N)}-\mathbf{x}_{l,j}^{(N)}\right), \tag{67}$$

have been adopted; here, $\hat{\mathbf{x}}_{l/(l-1)}^{(L)}$ and $\mathbf{C}_{l/(l-1)}^{(L)}$ denote the mean and the covariance matrix, respectively, of the prediction $\mathbf{x}_{l/l-1}^{(L)}$ of $\mathbf{x}_l^{(L)}$ based on the sequence of vectors $\mathbf{y}_{1:(l-1)}$, $w_{l,j}$ denotes the weight associated with the j-th particle $\mathbf{x}_{l,j}^{(N)}$ (with $j=0,1,...,\ N_p-1$) at the beginning of the l-th recursion and N_p is the overall number of particles. It is also important to point out that:

- The turbo filtering procedure illustrated below is accomplished in a way that the functional form of $\vec{m}_l\left(\mathbf{x}^{(L)}\right)$ (66) and $\vec{m}_l\left(\mathbf{x}^{(N)}\right)$ (67) is *preserved* from recursion to recursion (in other words, the messages $\vec{m}_{l+1}\left(\mathbf{x}^{(L)}\right)$ and $\vec{m}_{l+1}\left(\mathbf{x}^{(N)}\right)$ generated at the end of the l-th recursion have the same functional form as $\vec{m}_l\left(\mathbf{x}^{(L)}\right)$ (66) and $\vec{m}_l\left(\mathbf{x}^{(N)}\right)$ (67), respectively).
- For $l=1$ (35) and (36) are replaced by its Gaussian projection

$$\vec{m}_1\left(\mathbf{x}^{(L)}\right)=\mathcal{N}\left(\mathbf{x}_l^{(L)};\hat{\mathbf{x}}_1^{(L)},\mathbf{C}_1^{(L)}\right) \tag{68}$$

  and its particle-based representation

$$\vec{m}_1\left(\mathbf{x}^{(N)}\right)=\sum_{j=0}^{N_p-1}w_{1,j}\,\delta\left(\mathbf{x}_1^{(N)}-\mathbf{x}_{1,j}^{(N)}\right); \tag{69}$$

  here, the mean $\hat{\mathbf{x}}_1^{(L)}$and the covariance $\mathbf{C}_1^{(L)}$ are evaluated on the basis of the pdf $f\left(\mathbf{x}_1^{(L)}\right)=\int f\left(\mathbf{x}_1\right)d\mathbf{x}_1^{(N)}$, whereas the particles $\left\{\mathbf{x}_{1,j}^{(N)}\right\}$ are drawn from $f\left(\mathbf{x}_1^{(N)}\right)=\int f\left(\mathbf{x}_1\right)d\mathbf{x}_1^{(L)}$ (the quantities $\{w_{1,j}\}$ represent the associated weights).

Under the above assumptions the message passing and initialization procedures for linear Gaussian systems can be formulated in an elegant way, as illustrated below for the l-th recursion.

Initialization procedure - All that has been illustrated in the previous Paragraph for the initialization in the most general case apply to this specific class of systems too. In addition, the following steps are carried out:

- Given (66), the quantities

$$\mathbf{W}_{l/(l-1)}^{(L)} \triangleq \left(\mathbf{C}_{l/(l-1)}^{(L)}\right)^{-1} \tag{70}$$

and

$$\hat{\mathbf{w}}_{l/(l-1)}^{(L)} \triangleq \mathbf{W}_{l/(l-1)}^{(L)} \hat{\mathbf{x}}_{l/(l-1)}^{(L)} \tag{71}$$

are evaluated.

- The matrix inverses $\mathbf{W}_e^{(N)} \triangleq \left(\mathbf{C}_e^{(N)}\right)^{-1}$, $\mathbf{W}_e^{(L)} \triangleq \left(\mathbf{C}_e^{(L)}\right)^{-1}$, $\mathbf{W}_e^{(NL)} \triangleq \left(\mathbf{C}_e^{(NL)}\right)^{-1}$, $\mathbf{W}_w^{(N)} \triangleq \left(\mathbf{C}_w^{(N)}\right)^{-1}$ and $\mathbf{W}_w^{(L)} \triangleq \left(\mathbf{C}_w^{(L)}\right)^{-1}$ are computed.

Message passing procedure - A description of the steps forming this procedure is provided below for the k-th iteration (with $k = 1, 2, ..., N_{it}$).

1. First *measurement update* for $\mathbf{x}_l^{(N)}$ - This update step requires the knowledge of $\vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(N)}\right)$, which is evaluated on the basis of $\tilde{\mathbf{y}}_l^{(NL)}$ (8) and the message $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right)$ (122). Since $\mathbf{y}_l^{(NL)}$ is a deterministic vector and $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right)$ is a Gaussian message, it is easy to show that

$$\vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(N)}\right) = \mathcal{N}\left(\tilde{\mathbf{y}}_l^{(N)}; \eta_{\tilde{\mathbf{y}}_{l,k}^{(N)}}, \mathbf{C}_{\tilde{\mathbf{y}}_{l,k}^{(N)}}\right), \tag{72}$$

where $\eta_{\tilde{\mathbf{y}}_{l,k}^{(N)}} = \mathbf{y}_l^{(NL)} - \mathbf{C}_l^{(NL)} \eta_{4,l,k-1}^{(L)}$ and $\mathbf{C}_{\tilde{\mathbf{y}}_{l,k}^{(N)}} = \mathbf{C}_l^{(NL)} \mathbf{C}_{4,l,k-1}^{(L)} \left(\mathbf{C}_l^{(NL)}\right)^T$; from these quantities

$$\mathbf{W}_{\tilde{\mathbf{y}}_{l,k}^{(N)}} \triangleq \mathbf{C}_{\tilde{\mathbf{y}}_{l,k}^{(N)}}^{-1} \tag{73}$$

and

$$\mathbf{w}_{\tilde{\mathbf{y}}_{l,k}^{(N)}} \triangleq \mathbf{W}_{\tilde{\mathbf{y}}_{l,k}^{(N)}} \eta_{\tilde{\mathbf{y}}_{l,k}^{(N)}} \tag{74}$$

can be evaluated. Then substituting (62), (63) and (72) in (38) produces, after some manipulation (see the Appendix for the evaluation of the integral appearing in the RHS of (38))

$$\begin{aligned} \vec{m}_1^{(k)}\left(\mathbf{x}_l^{(N)}\right) \propto \exp\Big[&-\left(\mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right) - \eta_{1,l,k}^{(N)}\right)^T \\ &\cdot \mathbf{W}_{1,l,k}^{(N)}\left(\mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right) - \eta_{1,l,k}^{(N)}\right)\Big] \\ &\cdot \mathcal{N}\left(\mathbf{y}_l^{(N)}; \mathbf{h}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right), \mathbf{C}_e^{(N)}\right), \end{aligned} \tag{75}$$

with

$$\mathbf{W}_{1,l,k}^{(N)} \triangleq \mathbf{W}_e^{(NL)}\left[\mathbf{I}_{D_N} - \left[\mathbf{W}_e^{(NL)} + \mathbf{W}_{\tilde{\mathbf{y}}_{l,k}^{(N)}}\right]^{-1} \mathbf{W}_e^{(NL)}\right] \tag{76}$$

and

$$\begin{aligned} \mathbf{w}_{1,l,k}^{(N)} &\triangleq \mathbf{W}_{1,l,k}^{(N)} \eta_{1,l,k}^{(N)} \\ &= \mathbf{W}_e^{(NL)}\left[\mathbf{W}_e^{(NL)} + \mathbf{W}_{\tilde{\mathbf{y}}_{l,k}^{(N)}}\right]^{-1} \mathbf{w}_{\tilde{\mathbf{y}}_{l,k}^{(N)}}. \end{aligned} \tag{77}$$

Then, substituting (67) and (75) in (39) yields

$$\vec{m}_2^{(k)}\left(\mathbf{x}_l^{(N)}\right) = K_2^{(k)} \sum_{j=0}^{N_p-1} \tilde{w}_{l,j}^{(k)}\, \delta\left(\mathbf{x}_l^{(N)} - \mathbf{x}_{l,j}^{(N)}\right), \tag{78}$$

where

$$\begin{aligned} \tilde{w}_{l,j}^{(k)} \triangleq w_{l,j} \exp\Big[& -\left(\mathbf{h}_l^{(NL)}\left(\mathbf{x}_{l,j}^{(N)}\right) - \eta_{1,l,k}^{(N)}\right)^T \\ & \cdot \mathbf{W}_{1,l,k}^{(N)}\left(\mathbf{h}_l^{(NL)}\left(\mathbf{x}_{l,j}^{(N)}\right) - \eta_{1,l,k}^{(N)}\right)\Big] \\ & \cdot \mathcal{N}\left(\mathbf{y}_{N,l}; \mathbf{h}_l^{(N)}\left(\mathbf{x}_{l,j}^{(N)}\right), \mathbf{C}_e^{(N)}\right) \end{aligned} \tag{79}$$

and $K_2^{(k)}$ is a proper normalization constant. Note that: a) the first measurement update modifies the weights of the particles, but not the particles themselves; b) in each iteration, the initial values of the particles representing the pdf of $\mathbf{x}_l^{(N)}$ are always the same, since they are specified by $\vec{m}_l\left(\mathbf{x}^{(N)}\right)$ (67).

2. Second *measurement update* for $\mathbf{x}_l^{(N)}$ - This update requires the knowledge of $\vec{m}^{(k)}\left(\mathbf{z}_l^{(N)}\right)$ (see (41)). Note that in the first iteration $\vec{m}^{(1)}\left(\mathbf{z}_l^{(N)}\right) = 1$ (see (55)), so that $\vec{m}_3^{(1)}\left(\mathbf{x}_l^{(N)}\right) = 1$ and $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right) = \vec{m}_2^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (see (41) and (42), respectively). In the following iterations (i.e., for $k > 1$) the assumption of a Gaussian model for the random variables $\mathbf{x}_{l+1}^{(L)}$ and $\mathbf{x}_l^{(L)}$ appearing in the definition (28) of $\mathbf{z}_l^{(N)}$ leads to adopting the Gaussian model

$$\vec{m}^{(k)}\left(\mathbf{z}_l^{(N)}\right) = \mathcal{N}\left(\mathbf{z}_l^{(N)}; \eta_{\mathbf{z}_{l,k}^{(N)}}, \mathbf{C}_{\mathbf{z}_{l,k}^{(N)}}\right), \tag{80}$$

for $\mathbf{z}_l^{(N)}$ too. In the Appendix it is shown that

$$\eta_{\mathbf{z}_{l,k}^{(N)}} = \eta_{5,l,k-1}^{(L)} - \mathbf{A}_l^{(L)} \eta_{4,l,k-1}^{(L)} \tag{81}$$

and

$$\mathbf{C}_{\mathbf{z}_{l,k}^{(N)}} = \mathbf{C}_{5,l,k-1}^{(L)} - \mathbf{A}_l^{(L)} \mathbf{C}_{4,l,k-1}^{(L)} \left(\mathbf{A}_l^{(L)}\right)^T, \tag{82}$$

where $\eta_{4,l,k-1}^{(L)}$ and $\eta_{5,l,k-1}^{(L)}$ ($\mathbf{C}_{4,l,k-1}^{(L)}$ and $\mathbf{C}_{5,l,k-1}^{(L)}$) are mean vectors (covariance matrices) defined at step 4 and at step 5, respectively (see (122) and (141)).

Given $\eta_{\mathbf{z}_{l,k}^{(N)}}$ (81) and $\mathbf{C}_{\mathbf{z}_{l,k}^{(N)}}$ (82), the quantities

$$\mathbf{W}_{\mathbf{z}_{l,k}^{(N)}} \triangleq \mathbf{C}_{\mathbf{z}_{l,k}^{(N)}}^{-1} \tag{83}$$

and

$$\mathbf{w}_{\mathbf{z}_{l,k}^{(N)}} \triangleq \mathbf{W}_{\mathbf{z}_{l,k}^{(N)}} \eta_{\mathbf{z}_{l,k}^{(N)}} \tag{84}$$

are evaluated. Then, substituting (64) and (80) in (41) yields (see the Appendix, point 2., and, in particular, (165) and (166))

$$\begin{aligned} \vec{m}_3^{(k)}\left(\mathbf{x}_l^{(N)}\right) &= \int \mathcal{N}\left(\mathbf{z}_l^{(N)}; \eta_{\mathbf{z}_{l,k}^{(N)}}, \mathbf{C}_{\mathbf{z}_{l,k}^{(N)}}\right) \\ &\mathcal{N}\left(\mathbf{z}_l^{(N)}; \mathbf{f}_l^{(L)}\left(\mathbf{x}_{N,l}\right), \mathbf{C}_w^{(L)}\right) d\mathbf{z}_l^{(N)} \\ &\propto \exp\left[-\left(\mathbf{f}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right) - \eta_{3,l,k}^{(N)}\right)^T \right. \\ &\left. \mathbf{W}_{3,l,k}^{(N)}\left(\mathbf{f}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right) - \eta_{3,l,k}^{(N)}\right)\right] \end{aligned} \tag{85}$$

with

$$\mathbf{W}_{3,l,k}^{(N)} \triangleq \mathbf{W}_w^{(L)}\left[\mathbf{I}_{D_L} - \left[\mathbf{W}_w^{(L)} + \mathbf{W}_{\mathbf{z}_{l,k}^{(N)}}\right]^{-1} \mathbf{W}_w^{(L)}\right] \tag{86}$$

and

$$\begin{aligned} \mathbf{w}_{3,l,k}^{(N)} &\triangleq \mathbf{W}_{3,l,k}^{(N)} \eta_{3,l,k}^{(N)} \\ &= \mathbf{W}_w^{(L)}\left[\mathbf{W}_w^{(L)} + \mathbf{W}_{\mathbf{z}_{l,k}^{(N)}}\right]^{-1} \mathbf{w}_{\mathbf{z}_{l,k}^{(N)}}. \end{aligned} \tag{87}$$

Finally, substituting (85) and (78) in (42) gives

$$\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right) = \sum_{j=0}^{N_p-1} \hat{w}_{l,j}^{(k)}\, \delta\left(\mathbf{x}_l^{(N)} - \mathbf{x}_{l,j}^{(N)}\right), \tag{88}$$

where

$$\begin{aligned} \hat{w}_{l,j}^{(k)} \triangleq K_3^{(k)} \tilde{w}_{l,j}^{(k)}\ \exp&\left[-\left(\mathbf{f}_l^{(L)}\left(\mathbf{x}_{l,j}^{(N)}\right) - \eta_{3,l,j}^{(N)}\right)^T \right. \\ &\left. \mathbf{W}_{3,l,k}^{(N)}\left(\mathbf{f}_l^{(L)}\left(\mathbf{x}_{l,j}^{(N)}\right) - \eta_{3,l,j}^{(N)}\right)\right] \end{aligned} \tag{89}$$

and $K_3^{(k)}$ is a proper normalization constant. Finally, it is important to point out that: a) the second measurement update, like the first one, modifies the weights of the particles, but not the particles themselves; b) since turbo filtering for linear Gaussian systems is devised in a way that the messages about $\mathbf{x}_l^{(L)}$ are always Gaussian and $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ is exploited in the evaluation of such messages, in this step $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ is also projected into the (mean and covariance preserving) Gaussian message

$$\vec{m}_{4,G}^{(k)}\left(\mathbf{x}_l^{(N)}\right) = \mathcal{N}\left(\mathbf{x}_l^{(N)}; \eta_{4,l,k}^{(N)}, \mathbf{C}_{4,l,k}^{(N)}\right), \tag{90}$$

where

$$\eta_{4,l,k}^{(N)} \triangleq \sum_{j=0}^{N_p-1} \hat{w}_{l,j}^{(k)}\, \mathbf{x}_{l,j}^{(N)} \tag{91}$$

and

$$\mathbf{C}_{4,l,k}^{(N)} \triangleq \sum_{j=0}^{N_p-1} \hat{w}_{l,j}^{(k)}\left(\mathbf{x}_{l,j}^{(N)} - \eta_{4,l,k}^{(N)}\right)\left(\mathbf{x}_{l,j}^{(N)} - \eta_{4,l,k}^{(N)}\right)^T \tag{92}$$

represent the mean and covariance matrix, respectively, of $\mathbf{x}_l^{(N)}$ evaluated on the basis of $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (88).

3. First *measurement update* for $\mathbf{x}_l^{(L)}$ - From (19) and $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (88), the message $\vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(L)}\right)$ can be easily generated in particle form as

$$\vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(L)}\right) = \sum_{j=0}^{N_p-1} \hat{w}_{l,j}^{(k)}\, \delta\left(\tilde{\mathbf{y}}_l^{(L)} - \tilde{\mathbf{y}}_{l,j}^{(L)}\right), \tag{93}$$

where

$$\tilde{\mathbf{y}}_{l,j}^{(L)} \triangleq \mathbf{y}_l^{(NL)} - \mathbf{h}_l^{(NL)}\left(\mathbf{x}_{l,j}^{(N)}\right), \tag{94}$$

for $j = 0, 1, ..., N_p-1$. However, as already explained above, message passing for $\mathbf{x}_l^{(L)}$involves Gaussian messages only, $\vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(L)}\right)$ is projected into the (mean and covariance preserving) Gaussian message

$$\vec{m}_G^{(k)}\left(\tilde{\mathbf{y}}_l^{(L)}\right) = \mathcal{N}\left(\tilde{\mathbf{y}}_l^{(L)}; \eta_{\tilde{\mathbf{y}}_{l,k}^{(L)}}, \mathbf{C}_{\tilde{\mathbf{y}}_{l,k}^{(L)}}\right), \tag{95}$$

where

$$\eta_{\tilde{\mathbf{y}}_{l,k}^{(L)}} \triangleq \sum_{j=0}^{N_p-1} \hat{w}_{l,j}^{(k)}\, \tilde{\mathbf{y}}_{l,j}^{(L)} \tag{96}$$

and

$$\mathbf{C}_{\tilde{\mathbf{y}}_{l,k}^{(L)}} \triangleq \sum_{j=0}^{N_p-1} \hat{w}_{l,j}^{(k)}\left(\tilde{\mathbf{y}}_{l,j}^{(L)} - \eta_{\tilde{\mathbf{y}}_{l,k}^{(L)}}\right)\left(\tilde{\mathbf{y}}_{l,j}^{(L)} - \eta_{\tilde{\mathbf{y}}_{l,k}^{(L)}}\right)^T. \tag{97}$$

denote the mean and the covariance matrix, respectively, of $\tilde{\mathbf{y}}_l^{(L)}$ evaluated on the basis of the pdf $\vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(L)}\right)$ (93). Then, the quantities

$$\mathbf{W}_{\tilde{\mathbf{y}}_{l,k}^{(L)}} \triangleq \mathbf{C}_{\tilde{\mathbf{y}}_{l,k}^{(L)}}^{-1} \tag{98}$$

and

$$\mathbf{w}_{\tilde{\mathbf{y}}_{l,k}^{(L)}} \triangleq \mathbf{W}_{\tilde{\mathbf{y}}_{l,k}^{(L)}} \eta_{\tilde{\mathbf{y}}_l^{(L)}} \tag{99}$$

are evaluated. It is interesting to note that a less computationally demanding (but also less accurate) alternative is also available for an approximate evaluation of the quantities $\eta_{\tilde{\mathbf{y}}_{l,k}^{(L)}}$ (96) and $\mathbf{C}_{\tilde{\mathbf{y}}_{l,k}^{(L)}}$ (97). This alternative is based on: a) rewriting $\tilde{\mathbf{y}}_l^{(L)}$ (19) as

$$\tilde{\mathbf{y}}_l^{(L)} = \tilde{\mathbf{y}}_l^{(NL)} - \mathbf{h}_l^{(NL)}\left(\eta_{4,l,k}^{(N)} + \epsilon_{l,k}^{(N)}\right), \tag{100}$$

where $\epsilon_{l,k}^{(N)} \triangleq \mathbf{x}_l^{(N)} - \eta_{4,l,k}^{(N)}$ is modelled as a D_N-dimensional Gaussian vector having zero mean and covariance $\mathbf{C}_{4,l,k}^{(N)}$ (92); b) adopting the first order Taylor approximation

$$\begin{aligned} &\mathbf{h}_l^{(NL)}\left(\eta_{4,l,k}^{(N)} + \epsilon_{l,k}^{(N)}\right) \\ &\cong \mathbf{h}_l^{(NL)}\left(\eta_{4,l,k}^{(N)}\right) + \mathbf{J}_{h^{(NL)},l,k}\, \epsilon_{l,k}^{(N)}, \end{aligned} \tag{101}$$

where $\mathbf{J}_{h^{(NL)},l,k}$ denotes the Jacobian of $\mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right)$ evaluated at $\mathbf{x}_l^{(N)} = \eta_{4,l,k}^{(N)}$. Then, substituting (101) in (100) yields the approximate expression

$$\tilde{\mathbf{y}}_l^{(L)} \cong \tilde{\mathbf{y}}_l^{(NL)} - \mathbf{h}_l^{(NL)}\left(\eta_{4,l,k}^{(N)}\right) - \mathbf{J}_{h^{(NL)},l,k}\epsilon_{l,k}^{(N)}, \tag{102}$$

from which the approximate expressions

$$\eta_{\tilde{\mathbf{y}}_{l,k}^{(L)}} \cong \tilde{\mathbf{y}}_l^{(NL)} - \mathbf{h}_l^{(NL)}\left(\eta_{4,l,k}^{(N)}\right) \tag{103}$$

and

$$\mathbf{C}_{\tilde{\mathbf{y}}_{l,k}^{(L)}} \cong \mathbf{J}_{h^{(NL)},l,k}\mathbf{C}_{4,l,k}^{(N)}\mathbf{J}_{h^{(NL)},l,k}^T \tag{104}$$

can be easily derived.

Given $\vec{m}_G^{(k)}\left(\tilde{\mathbf{y}}_l^{(L)}\right)$ (95), $\vec{m}_1^{(k)}\left(\mathbf{x}_l^{(L)}\right)$ (43) can be evaluated as follows. Substituting (95), and the pdfs (58) and (59) (referring to $\mathbf{y}_l^{(L)}|\mathbf{x}_l^{(L)}$ and $\tilde{\mathbf{y}}_l^{(L)}|\mathbf{x}_l^{(L)}$, respectively) in (43) and keeping into account that $f\left(\mathbf{y}_l^{(L)}|\mathbf{x}_l^{(L)}\right)$ can be rewritten as

$$\begin{aligned} & f\left(\mathbf{y}_l^{(L)}\middle|\mathbf{x}_l^{(L)}\right) \\ = \mathcal{N}\Bigg(\mathbf{x}_l^{(L)}; & \left(\left(\mathbf{C}_l^{(L)}\right)^T\mathbf{W}_e^{(L)}\mathbf{C}_l^{(L)}\right)^{-1}\left(\mathbf{C}_l^{(L)}\right)^T\mathbf{W}_e^{(L)}\mathbf{y}_l^{(L)}, \\ & \left(\left(\mathbf{C}_l^{(L)}\right)^T\mathbf{W}_e^{(L)}\mathbf{C}_l^{(L)}\right)^{-1}\Bigg) \end{aligned} \tag{105}$$

yields, after some manipulation (see the Appendix, point 1., and, in particular, (154) and (155)),

$$\vec{m}_1^{(k)}\left(\mathbf{x}_l^{(L)}\right) = \mathcal{N}\left(\mathbf{x}_l^{(L)}; \eta_{1,l,k}^{(L)}, \mathbf{C}_{1,l,k}^{(L)}\right) \tag{106}$$

with

$$\mathbf{W}_{1,l,k}^{(L)} \triangleq \left(\mathbf{C}_l^{(L)}\right)^T\mathbf{W}_e^{(L)}\mathbf{C}_l^{(L)} + \mathbf{W}_{0,l,k}^{(L)}, \tag{107}$$

and

$$\mathbf{w}_{1,l,k}^{(L)} \triangleq \mathbf{W}_{1,l,k}^{(L)}\eta_{1,l,k}^{(L)} = \left(\mathbf{C}_l^{(L)}\right)^T\mathbf{W}_e^{(L)}\mathbf{y}_l^{(L)} + \mathbf{w}_{0,l,k}^{(L)}. \tag{108}$$

Here, $\mathbf{W}_{0,l,k}^{(L)}$ and $\mathbf{w}_{0,l,k}^{(L)}$ are given by (see the Appendix, point 2., and, in particular, (165) and (166))

$$\begin{aligned} \mathbf{W}_{0,l,k}^{(L)} \triangleq & \left(\mathbf{C}_l^{(NL)}\right)^T\mathbf{W}_e^{(NL)} \\ \cdot & \left[\mathbf{I}_{P_{NL}} - \left[\mathbf{W}_{\tilde{\mathbf{y}}_{l,k}^{(L)}} + \mathbf{W}_e^{(NL)}\right]^{-1}\mathbf{W}_e^{(NL)}\right]\mathbf{C}_l^{(NL)} \end{aligned} \tag{109}$$

and

$$\begin{aligned} & \mathbf{w}_{0,l,k}^{(L)} \triangleq \mathbf{W}_{0,l,k}^{(L)}\eta_{0,l,k}^{(L)} \\ = & \left(\mathbf{C}_l^{(NL)}\right)^T\mathbf{W}_e^{(NL)}\left[\mathbf{W}_{\tilde{\mathbf{y}}_{l,k}^{(L)}} + \mathbf{W}_e^{(NL)}\right]^{-1}\mathbf{w}_{\tilde{\mathbf{y}}_{l,k}^{(L)}}, \end{aligned} \tag{110}$$

respectively. Finally, substituting (106) and (66) in (44) produces (see the Appendix, point 1., and, in particular, (154) and (155))

$$\vec{m}_2^{(k)}\left(\mathbf{x}_l^{(L)}\right)=\mathcal{N}\left(\mathbf{x}_{L,l},\eta_{2,l,k}^{(L)},\mathbf{C}_{2,l,k}^{(L)}\right), \tag{111}$$

with

$$\mathbf{W}_{2,l,k}^{(L)}\triangleq\left(\mathbf{C}_{2,l,k}^{(L)}\right)^{-1}=\mathbf{W}_{l/(l-1)}^{(L)}+\mathbf{W}_{1,l,k}^{(L)} \tag{112}$$

and

$$\mathbf{w}_{2,l,k}^{(L)}\triangleq\mathbf{W}_{2,l,k}^{(L)}\eta_{2,l,k}^{(L)}=\mathbf{w}_{l/(l-1)}^{(L)}+\mathbf{w}_{1,l,k}^{(L)}. \tag{113}$$

4. Second *measurement update* for $\mathbf{x}_l^{(L)}$ - This update requires the knowledge of $\vec{m}^{(k)}\left(\mathbf{z}_l^{(L)}\right)$ (see (46)). In the first iteration $\vec{m}^{(1)}\left(\mathbf{z}_l^{(L)}\right)=1$ (see (55)), so that $\vec{m}_3^{(1)}\left(\mathbf{x}_l^{(L)}\right)=1$ and $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(L)}\right)=\vec{m}_2^{(k)}\left(\mathbf{x}_l^{(L)}\right)$ (see (46) and (47), respectively). In the following iterations (i.e., for $k>1$) the Gaussian model

$$\vec{m}^{(k)}\left(\mathbf{z}_l^{(L)}\right)=\mathcal{N}(\mathbf{z}_l^{(L)};\eta_{\mathbf{z}_{l,k}^{(L)}},\mathbf{C}_{\mathbf{z}_{l,k}^{(L)}}) \tag{114}$$

is adopted. In the Appendix it is proved that the mean $\eta_{\mathbf{z}_{l,k}^{(L)}}$ and the covariance matrix $\mathbf{C}_{\mathbf{z}_{l,k}^{(L)}}$ of $\mathbf{z}_l^{(L)}$ are given by

$$\eta_{\mathbf{z}_{l,k}^{(L)}}=\eta_{5,l,k-1}^{(N)}-\mathbf{f}_l^{(N)}\left(\eta_{4,l,k}^{(N)}\right) \tag{115}$$

and

$$\mathbf{C}_{\mathbf{z}_{l,k}^{(L)}}=\mathbf{C}_{5,l,k-1}^{(N)}-\mathbf{J}_{f^{(N)},l,k}\mathbf{C}_{4,l,k}^{(N)}\left(\mathbf{J}_{f^{(N)},l,k}\right)^T \tag{116}$$

respectively, where $\eta_{4,l,k}^{(N)}$ and $\eta_{5,l,k-1}^{(N)}$ ($\mathbf{C}_{4,l,k}^{(N)}$ and $\mathbf{C}_{5,l,k-1}^{(N)}$) are mean vectors (covariance matrices) defined in this step and in the following step, respectively. From $\eta_{\mathbf{z}_{l,k}^{(L)}}$ (115) and $\mathbf{C}_{\mathbf{z}_{l,k}^{(L)}}$ (116) the quantities

$$\mathbf{W}_{\mathbf{z}_{l,k}^{(L)}}\triangleq\mathbf{C}_{\mathbf{z}_{l,k}^{(L)}}^{-1} \tag{117}$$

and

$$\mathbf{w}_{\mathbf{z}_{l,k}^{(L)}}\triangleq\mathbf{W}_{\mathbf{z}_{l,k}^{(L)}}\eta_{\mathbf{z}_{l,k}^{(L)}} \tag{118}$$

are evaluated. Then, substituting (114) and (60) in (46) yields (see the Appendix, point 2., and, in particular, (165) and (166))

$$\vec{m}_3^{(k)}\left(\mathbf{x}_l^{(L)}\right)=\mathcal{N}\left(\mathbf{x}_l^{(L)};\eta_{3,l,k}^{(L)},\mathbf{C}_{3,l,k}^{(L)}\right), \tag{119}$$

with

$$\begin{aligned}\mathbf{W}_{3,l,k}^{(L)}\triangleq&\left(\mathbf{A}_l^{(N)}\right)^T\mathbf{W}_w^{(N)}\\ &\cdot\left[\mathbf{I}_{D_N}-\left[\mathbf{W}_{\mathbf{z}_{l,k}^{(L)}}+\mathbf{W}_w^{(N)}\right]^{-1}\mathbf{W}_w^{(N)}\right]\mathbf{A}_l^{(N)}\end{aligned} \tag{120}$$

and

$$
\begin{aligned}
\mathbf{w}_{3,l,k}^{(L)} &\triangleq \mathbf{W}_{3,l,k}^{(L)} \eta_{3,l,k}^{(L)} \\
= \left(\mathbf{A}_l^{(N)}\right)^T & \mathbf{W}_w^{(N)} \left[\mathbf{W}_{\mathbf{z}_{l,k}^{(L)}} + \mathbf{W}_w^{(N)}\right]^{-1} \mathbf{w}_{\mathbf{z}_{l,k}^{(L)}}.
\end{aligned}
\tag{121}
$$

Finally, from (47), (111), and (119) it is easily inferred that (see the Appendix, point 1., and, in particular, (154) and (155))

$$
\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(L)}\right) = \mathcal{N}\left(\mathbf{x}_l^{(L)}; \eta_{4,l,k}^{(L)}, \mathbf{C}_{4,l,k}^{(L)}\right), \tag{122}
$$

with

$$
\mathbf{W}_{4,l,k}^{(L)} \triangleq \left(\mathbf{C}_{4,l,k}^{(L)}\right)^{-1} = \mathbf{W}_{2,l,k}^{(L)} + \mathbf{W}_{3,l,k}^{(L)} \tag{123}
$$

and

$$
\mathbf{w}_{4,l,k}^{(L)} \triangleq \mathbf{W}_{4,l,k}^{(L)} \eta_{4,l,k}^{(L)} = \mathbf{w}_{2,l,k}^{(L)} + \mathbf{w}_{3,l,k}^{(L)}. \tag{124}
$$

5. *Time update* for $\mathbf{x}_l^{(L)}$ and $\mathbf{x}_l^{(N)}$ - This update generates the messages $\vec{m}_6^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right)$ and $\vec{m}_6^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right)$. The first message is evaluated on the basis of (48), which requires a double integration. We first focus on the integral

$$
\int f\left(\mathbf{x}_{l+1}^{(N)} \middle| \mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) \vec{m}_4^{(k)}\left(\mathbf{x}_l^{(L)}\right) d\mathbf{x}_l^{(L)} \tag{125}
$$

and substitute (65) and (122) in it. This produces the function (see the Appendix, point 3., and, in particular, (167), (168) and (169))

$$
\begin{aligned}
g_{l,k}\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right) &= \int \mathcal{N}\left(\mathbf{x}_l^{(L)}; \eta_{4,l,k}^{(L)}, \mathbf{C}_{4,l,k}^{(L)}\right) \\
\cdot \mathcal{N}\left(\mathbf{x}_{l+1}^{(N)}; \mathbf{f}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right) + \mathbf{A}_l^{(N)} \mathbf{x}_l^{(L)}, \mathbf{C}_w^{(N)}\right) & d\mathbf{x}_l^{(L)} \\
\propto \exp\Bigg[-\left(\mathbf{x}_{l+1}^{(N)} - \eta_{l,k}^{(g)}\left(\mathbf{x}_l^{(N)}\right)\right)^T & \\
\cdot \left(\mathbf{C}_{l,k}^{(g)}\right)^{-1} \left(\mathbf{x}_{l+1}^{(N)} - \eta_{l,k}^{(g)}\left(\mathbf{x}_l^{(N)}\right)\right)\Bigg]&,
\end{aligned}
\tag{126}
$$

where

$$
\mathbf{C}_{l,k}^{(g)} \triangleq \mathbf{C}_w^{(N)} + \mathbf{A}_l^{(N)} \mathbf{C}_{4,l,k}^{(L)} \left(\mathbf{A}_l^{(N)}\right)^T \tag{127}
$$

and

$$
\eta_{l,k}^{(g)}\left(\mathbf{x}_l^{(N)}\right) \triangleq \mathbf{A}_l^{(N)} \eta_{4,l,k}^{(L)} + \mathbf{f}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right). \tag{128}
$$

Then, the RHS of (48) can be rewritten as

$$
\begin{aligned}
\vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right) &= \int g_{l,k}\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right) \\
&\cdot \vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right) d\mathbf{x}_l^{(N)}.
\end{aligned}
\tag{129}
$$

The integration appearing in the RHS of the last expression can be easily carried out, thanks to the structure of $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (88). Actually, as already mentioned above turbo filtering is expected to preserve the functional form of the output messages generated by each recursion; as it will become clearer later, this result can be achieved for $\mathbf{x}_l^{(N)}$ expressing $\vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right)$ (129) in particle form. A particle list together with its list of weights for the last message can be generated using a *weighted sampling* procedure (see [4, Par. III.A]), which consists of the following steps: a) for $p = 0, 1, ..., N_p - 1$ select the point $\mathbf{x}_{l,p}^{(N)}$ in the particle list contained in $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (88); b) draw a sample[6] $\mathbf{x}_{l+1,k,p}^{(N)}$ from $g_{l,k}\left(\mathbf{x}_{l,p}^{(N)}, \mathbf{x}_{l+1}^{(N)}\right)$ (note that, for a given $\mathbf{x}_l^{(N)}$, $g_{l,k}(\cdot,\cdot)$ is a Gaussian function and that the set of particles referring to $\mathbf{x}_{l+1}^{(N)}$ change from iteration to iteration) assign to $\mathbf{x}_{l+1,k,p}^{(N)}$ a probability proportional to $\hat{w}_{l,p}^{(k)}$. This procedure generates the message

$$\begin{aligned}\vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right) &= \sum_{p=0}^{N_p-1} \hat{w}_{l,p}^{(k)}\, \delta\left(\mathbf{x}_{l+1}^{(N)} - \mathbf{x}_{l+1,k,p}^{(N)}\right) \\ &= \vec{m}_6^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right).\end{aligned} \tag{130}$$

Finally, $\vec{m}_5^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ is projected into the (mean and covariance preserving) Gaussian message

$$\vec{m}_{5,G}^{(k)}\left(\mathbf{x}_l^{(N)}\right) = \mathcal{N}\left(\mathbf{x}_{l+1}^{(N)}; \eta_{5,l,k}^{(N)}, \mathbf{C}_{5,l,k}^{(N)}\right) = \vec{m}_{6,G}^{(k)}\left(\mathbf{x}_{l+1}^{(N)}\right), \tag{131}$$

where

$$\eta_{5,l,k}^{(N)} \triangleq \sum_{p=0}^{N_p-1} \hat{w}_{l,j}^{(k)}\, \mathbf{x}_{l+1,k,p}^{(N)} \tag{132}$$

and

$$\mathbf{C}_{5,l,k}^{(N)} \triangleq \sum_{p=0}^{N_p-1} \hat{w}_{l,j}^{(k)} \left(\mathbf{x}_{l+1,k,p}^{(N)} - \eta_{5,l,k}^{(N)}\right)\left(\mathbf{x}_{l+1,k,p}^{(N)} - \eta_{5,l,k}^{(N)}\right)^T \tag{133}$$

represent the mean and covariance matrix, respectively, of $\mathbf{x}_{l+1}^{(N)}$ evaluated on the basis of $\vec{m}_5^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (130).

The evaluation of $\vec{m}_6^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right)$ is based on (50), which, similarly as (48), requires a double integration. We first take into consideration the integral

$$\int f\left(\mathbf{x}_{l+1}^{(L)} \middle| \mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) \vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right) d\mathbf{x}_l^{(N)} \tag{134}$$

and substitute (88) and (61) in it. This produces

$$\begin{aligned}&\int f\left(\mathbf{x}_{l+1}^{(L)} \middle| \mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) \vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right) d\mathbf{x}_l^{(N)} \\ &= \sum_{j=0}^{N_p-1} \hat{w}_{l,j}^{(k)} \mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \mathbf{f}_l^{(L)}\left(\mathbf{x}_{l,j}^{(N)}\right) + \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)}, \mathbf{C}_w^{(L)}\right).\end{aligned} \tag{135}$$

Then, substituting the last result and (122) in (50) yields

$$\begin{aligned}\vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right) &= \sum_{j=0}^{N_p-1} \hat{w}_{l,j}^{(k)} \int \mathcal{N}\left(\mathbf{x}_l^{(L)}; \eta_{4,l,k}^{(L)}, \mathbf{C}_{4,l,k}^{(L)}\right) \\ &\cdot \mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \mathbf{f}_l^{(L)}\left(\mathbf{x}_{l,j}^{(N)}\right) + \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)}, \mathbf{C}_w^{(L)}\right) d\mathbf{x}_l^{(L)}.\end{aligned} \tag{136}$$

[6]Note that *resampling* can be accomplished after this step in order to mitigate the effects of the so called *degeneracy problem* [7, 31].

Then, it is not difficult to show that (see the Appendix, point 3., and, in particular, (168) and (169))

$$\begin{aligned} \int \mathcal{N}\left(\mathbf{x}_l^{(L)}; \eta_{4,l,k}^{(L)}, \mathbf{C}_{4,l,k}^{(L)}\right) \cdot \mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \mathbf{f}_l^{(L)}\left(\mathbf{x}_{l,j}^{(N)}\right) + \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)}, \mathbf{C}_w^{(L)}\right) d\mathbf{x}_l^{(L)} \\ = \mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \eta_{1,l}^{(k)}\left(\mathbf{x}_{l,j}^{(N)}\right), \mathbf{C}_{1,l}^{(k)}\right), \end{aligned} \tag{137}$$

where

$$\mathbf{C}_{1,l}^{(k)} \triangleq \mathbf{C}_w^{(L)} + \mathbf{A}_l^{(L)}\mathbf{C}_{4,l,k}^{(L)}\left(\mathbf{A}_l^{(L)}\right)^T \tag{138}$$

and

$$\eta_{1,l}^{(k)}\left(\mathbf{x}_l^{(N)}\right) \triangleq \mathbf{A}_l^{(L)}\eta_{4,l,k}^{(L)} + \mathbf{f}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right). \tag{139}$$

Finally, substituting (137) in (136) produces

$$\begin{aligned} \vec{m}_5^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right) = \textstyle\sum_{j=0}^{N_p-1} \hat{w}_{l,j}^{(k)} \mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \eta_{1,l}^{(k)}\left(\mathbf{x}_{l,j}^{(N)}\right), \mathbf{C}_{1,l}^{(k)}\right) \\ = \vec{m}_6^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right), \end{aligned} \tag{140}$$

which, unluckily, is a Gaussian mixture. However, in our filtering technique this Gaussian mixture is replaced with a Gaussian pdf preserving mean and covariance [16], so that the Gaussianity of $\vec{m}_{l+1}^{(k)}\left(\mathbf{x}^{(L)}\right)$ is preserved. This leads to the new message

$$\begin{aligned} \vec{m}_{6,G}^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right) = \vec{m}_{5,G}^{(k)}\left(\mathbf{x}_{l+1}^{(L)}\right) \\ = \mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \eta_{5,l,k}^{(L)}, \mathbf{C}_{5,l,k}^{(L)}\right), \end{aligned} \tag{141}$$

where

$$\eta_{5,l,k}^{(L)} \triangleq \sum_{j=0}^{N_p-1} \hat{w}_{l,k}^{(j)}\, \eta_{1,l}^{(k)}\left(\mathbf{x}_{l,j}^{(N)}\right) \tag{142}$$

and

$$\begin{aligned} \mathbf{C}_{5,l,k}^{(L)} = \mathbf{C}_{1,l}^{(k)} \\ + \textstyle\sum_{j=0}^{N_p-1} \hat{w}_{l,k}^{(j)}\left(\eta_{1,l}^{(k)}\left(\mathbf{x}_{l,j}^{(N)}\right) - \eta_{5,l,k}^{(L)}\right)\left(\eta_{1,l}^{(k)}\left(\mathbf{x}_{l,j}^{(N)}\right) - \eta_{5,l,k}^{(L)}\right)^T. \end{aligned} \tag{143}$$

6. *Stop or get ready for the new recursion* - This step is the same as that described in the previous Paragraph.

Finally, it is important to point out that:

- The processing accomplished in the measurement and time update for $\mathbf{x}_l^{(L)}$ can be interpreted as a form of 'soft' Kalman filtering, since, unlike standard Kalman filtering, a portion of the available measurements (and, in particular, the information referring to $\mathbf{z}_l^{(L)}$ (20) and $\tilde{\mathbf{y}}_l^{(L)}$ (19) is not deterministic, but of probabilistic nature.
- The TF algorithm derived above is denoted TF #1 in the following, since other two TF techniques, based on the same line of reasoning illustrated above, are developed in the following Paragraph.

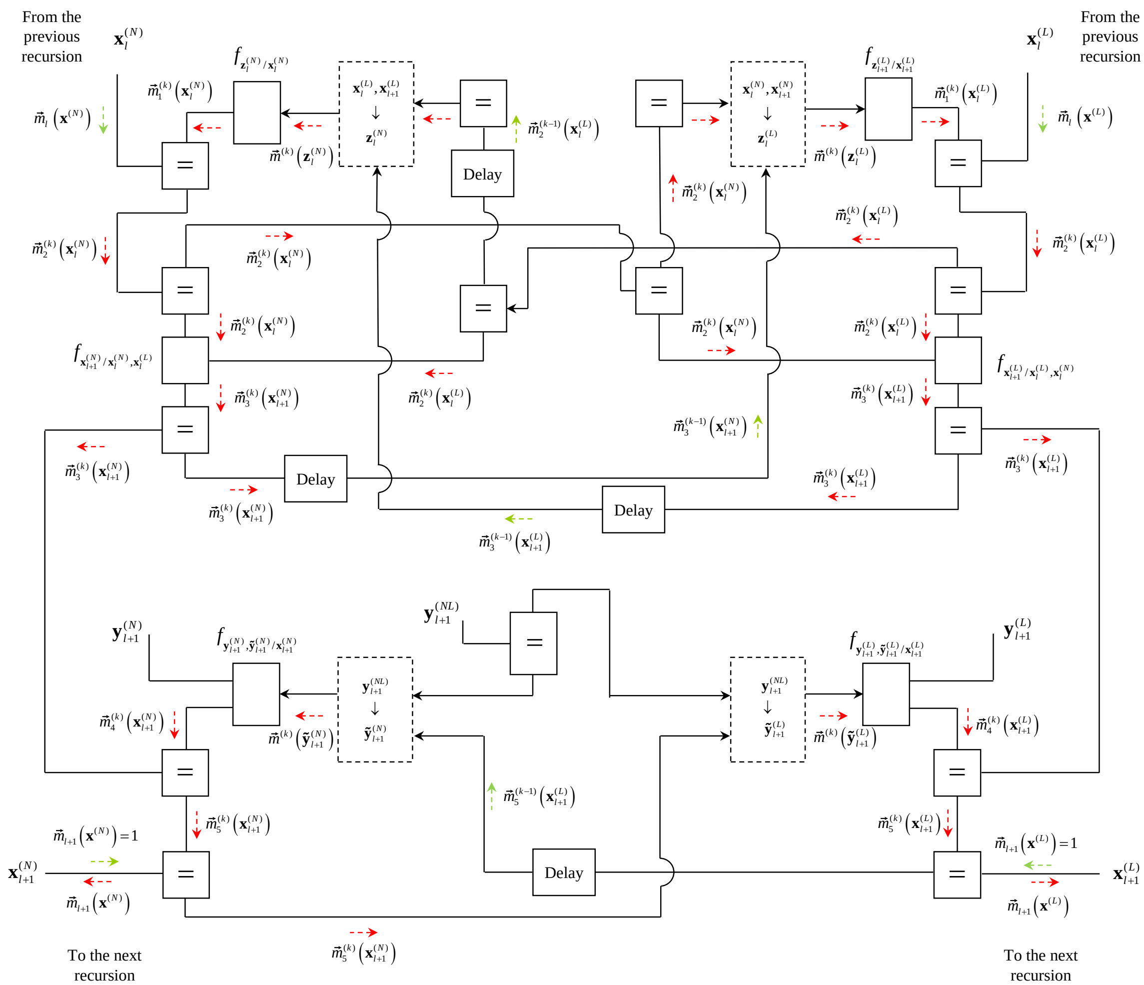

Figure 8: Message passing for a TF algorithm (called TF #2) developed on the basis of the FG shown in Fig. 5. All the quantities appearing in this figure refer to the k-th iteration of the l-th recursion; the messages available at the beginning of the considered iteration are indicated by green arrows.

0.4.3 Other filtering technique

As already mentioned at the end of Section 0.3, the FG shown in Fig. 5 can be employed in place of that shown in Fig. 4; for this FG different scheduling procedures, leading to distinct TF algorithms, can be considered. It is important to note that

- The FG of Fig. 5 can be partitioned in two subgraphs. On the one hand, the first subgraph consisting of upper part of the given graph and message passing over it allows to process the pseudo-measurement $\mathbf{z}_l^{(L)}$ and $\mathbf{z}_l^{(N)}$ to generate probabilistic information about $\mathbf{x}_{l+1}^{(L)}$ and $\mathbf{x}_{l+1}^{(N)}$. On the other hand, the second subgraph consists of the remaining part (i.e., of the lower part) of the same graph and message passing over it can be employed to refine our probabilistic knowledge of $\mathbf{x}_{l+1}^{(L)}$ and $\mathbf{x}_{l+1}^{(N)}$ on the basis of $\mathbf{y}_{l+1}^{(L)}$, $\mathbf{y}_{l+1}^{(N)}$ and $\mathbf{y}_{l+1}^{(NL)}$.
- In developing novel filtering techniques multiple passes can be accomplished within the first subgraph before passing messages to the second subgraph; similar considerations hold for message passing from the second subgraph to the first one.

On the basis of these considerations, two different scheduling procedures have been considered in the following for the message flow in the FG of 4. In the first procedure a single pass is accomplished within each subgraph but, generally speaking, N_{it} passes accomplished within the overall graph; the corresponding message flow for the k-th iteration of the l-th recursion is shown in Fig. 8. On the contrary, in the second procedure $N_{it,1}$ iterations accomplished within the upper subgraph are followed by $N_{it,2}$ iterations carried out within the lower subgraph; then, the resulting statistical information are passed to the next recursion. The resulting filtering techniques are denoted TF #2 and TF #3, respectively, in the following. Note that, unlike TF #1, mathematical expressions are not provided below for the messages evaluated in these additional filtering techniques, since they can be easily developed from those already derived for TF #1. In Section 0.5 the three TF options are compared in terms of performance in specific cases.

0.4.4 Computational complexity of turbo filtering for linear Gaussian systems

In this Paragraph the computational complexity of TF #1 is analysed[7]. Following [17], in assessing the overall computational load of a single iteration of our filtering technique, only the messages and the procedures influenced by the number of particles N_p have been taken into consideration, since provide the dominant contribution to the load itself. For this reason, the number of sums, products and other operations have been evaluated for the following tasks (see Table 1):

1. Computation of the message $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (88) - In particular, the complexity required for the evaluation of $\tilde{w}_{l,j}^{(k)}$ (79) (contribution # 1.a), $\hat{w}_{l,j}^{(k)}$ (89) (contribution # 1.b), and $\eta_{4,l,k}^{(N)}$ (91) and $\mathbf{C}_{4,l,k}^{(N)}$ (92) (contribution # 1.c) have been assessed.
2. Computation of the message $\vec{m}_5^{(k)}\left(\mathbf{x}_l^{(L)}\right)$ (140) - In particular, the complexity required for the evaluation of $\eta_{5,l,k}^{(L)}$ (142) and $\mathbf{C}_{5,l,k}^{(L)}$ (143) has been assessed (contribution # 2).

[7]The computational complexity of TF #2 is similar to that of TF #1, whereas that of TF #3 can be easily evaluated following the same line of reasoning as TF #1.

Contribution #	Sums	Products	Other
1.a	$2P_{NL}^2 N_p +$ $2D_N^2 N_p + 2N_p$	$2P_{NL}^2 N_p +$ $2D_N^2 N_p + 2N_p$	$2N_p + N_p \mathfrak{F}_{\mathbf{h}_{\mathrm{NL}}}$
1.b	$3D_L^2 N_p +$ $N_p D_L + N_p$	$2D_L^2 N_p + 3N_p$	$N_p + N_p \mathfrak{F}_{\mathbf{f}_{\mathrm{L}}}$
1.c	$N_p D_N + 2N_p D_N^2$	$N_p D_N + N_p D_N^2$	
2	$2N_p D_L^2 +$ $2D_L^2 + N_p D_L$	$2N_P D_L^2 +$ $2D_L^2 + N_p D_L$	$N_p \mathfrak{F}_{\mathbf{f}_{\mathrm{L}}}$
3	$D_N D_L^2 + D_N^2 D_L +$ $N_p D_N D_L + N_p D_N^2 +$ $2D_N^2 + 2N_p D_N$	$D_N D_L^2 + D_N^2 D_L +$ $N_p D_N D_L +$ $N_p D_N^2$	$c_{res} +$ $\mathfrak{F}_{\mathfrak{chol}} + N_p \mathfrak{F}_{\mathbf{f}_{\mathrm{N}}}$
4	$N_p D_N + 2N_p D_N^2$	$N_p D_N + N_p D_N^2$	

Table 1: Computational complexity of different procedures contained in TF #1 for linear Gaussian systems; the main tasks carried out in a single iteration are considered.

3. Particle generation and resampling procedure accomplished at step 5. (contribution # 3).
4. Computation of the message $\vec{m}_5^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (130) - In particular, the complexity required for the evaluation of $\eta_{5,l,k}^{(N)}$ (132) and $\mathbf{C}_{5,l,k}^{(N)}$ (133) has been assessed (contribution # 4).

Note that in Table 1 $\mathfrak{F}_{\mathbf{h}_{\mathrm{NL}}}$,$\mathfrak{F}_{\mathbf{f}_{\mathrm{L}}}$and $\mathfrak{F}_{\mathbf{f}_{\mathrm{N}}}$denote the computational complexity associated with the evaluation of the functions $\mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right)$, $\mathbf{f}_t^{(L)}\left(\mathbf{x}_t^{(N)}\right)$ and $\mathbf{f}_t^{(N)}\left(\mathbf{x}_t^{(N)}\right)$, respectively, whereas $\mathfrak{F}_{\mathfrak{chol}}$ represents the complexity of the Cholesky factorisations of the matrix $\mathbf{C}_{l,k}^{(g)}$ (127) (this is required at step. 5, when generating a new set of particles $\left\{\mathbf{x}_{l+1,k,p}^{(N)}\right\}$).

If we now assign a unit weight to all the operations considered in Table 1, conventionally assign the complexities P_{NL}, D_L and D_N to $\mathfrak{F}_{\mathbf{h}_{\mathrm{NL}}}$, $\mathfrak{F}_{\mathbf{f}_{\mathrm{L}}}$ and $\mathfrak{F}_{\mathbf{f}_{\mathrm{N}}}$, respectively (in other words, the complexity of each of these functions is deemed to be proportional to its size), and neglect $\mathfrak{F}_{\mathfrak{chol}}$ (since this is not influenced by N_p), the estimate

$$\begin{gathered} C_{TF}\left(D_L, D_N, N_p, P_L, P_N, P_{NL}\right) = \\ \left(4P_{NL}^2 + 12D_N^2 + 8 + 9D_L^2 + 7D_L + 8D_N + 2D_N D_L + 2P_{NL} + c_{res}\right) N_p \end{gathered} \tag{144}$$

can be easily obtained for the computational complexity, in terms of *floating point operations* (flops), of a single iteration in turbo filtering; here, c_{res} represents the contribution of resampling to flop count [17]. The corresponding estimate for marginalized particle filtering, evaluated on the basis of [17, Table I, p. 4409], is

$$\begin{gathered} C_{MPF}\left(D_L, D_N, N_p\right) = \\ \left(D_L D_N + 6D_N^2 + 2D_L^2 + D_N - D_L + D_N c_3 + c_1 + \right. \\ \left. c_2 + 4D_N D_L^2 + 8D_L D_N^2 + \tfrac{4}{3}D_N^3 + 6D_L^3\right) N_p, \end{gathered} \tag{145}$$

where c_1, c_2 and c_3 refer to the computation of the Gaussian likelihood, of resampling and of random numbers, respectively. Note that C_{MPF} is dominated by various cubic terms appearing in the brackets, which do not appear in C_{TF}; this evidences that the latter complexity should be expected to be substantially larger than the former one when the size of the state vector is large.

	TF #1	TF #1	MPF
	$N_{it}=2$	$N_{it}=3$	
$\mathbf{x}_t^{(L)}$	0.0990	0.0985	0.0875
$x_t^{(N)}$	0.1356	0.1345	0.1311

Table 2: RMSE performance provided by TF #1 and MPF for the considered system.

0.5 Numerical Results

In this Section TF is compared with MPF in terms of accuracy and complexity for two different linear Gaussian systems.

0.5.1 System A

The first system we consider (denoted *system A* in the following) is characterized by: a) the state model

$$\mathbf{x}_{t+1}^{(L)} = \begin{pmatrix} \cos\left(x_t^{(N)}\right) \\ -\sin\left(x_t^{(N)}\right) \end{pmatrix} + \begin{pmatrix} 1 & 0.3 \\ 0 & 0.92 \end{pmatrix} \mathbf{x}_t^{(L)} + \mathbf{w}_t^{(L)}, \tag{146}$$

$$x_{t+1}^{(N)} = \arctan x_t^{(N)} + (1 \quad 0.5)\,\mathbf{x}_t^{(L)} + w_t^{(N)}, \tag{147}$$

where $\mathbf{w}_t^{(L)} \sim \mathcal{N}(0, \mathbf{C}_w^{(L)})$ and $w_t^{(N)} \sim \mathcal{N}(0, C_w^{(N)})$ (consequently, $D = 3$, $D_L = 2$ and $D_N = 1$); b) the measurement model

$$\mathbf{y}_t = \begin{pmatrix} 0.1\left(x_t^{(N)}\right)^2 \cdot \operatorname{sgn}\left(x_t^{(N)}\right) \\ 0 \\ 1 + \cos\left(x_t^{(N)}\right) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 1 & -1 \\ 1 & 0 \end{pmatrix} \mathbf{x}_t^{(L)} + \mathbf{e}_t, \tag{148}$$

where $\mathbf{e}_t = [e_t^{(L)}, e_t^{(N)}, e_t^{(NL)}]^T$, with $e_t^{(L)} \sim \mathcal{N}(0, C_e^{(L)})$, $e_t^{(N)} \sim \mathcal{N}(0, C_e^{(N)})$, and $e_t^{(NL)} \sim \mathcal{N}(0, C_e^{(NL)})$ (consequently, $P = 3$ and $P_L = P_N = P_{NL} = 1$). In our simulations the *root mean square error* (RMSE) performance provided by TF and MPF for the considered system has been assessed under the following assumptions: a) $\mathbf{C}_w^{(L)} = 0.1\mathbf{I}_2$, $\mathbf{C}_w^{(N)} = 0.2$; b) $C_e^{(L)} = C_e^{(N)} = C_e^{(NL)} = 0.01$; c) $N_p = 200$ for both the considered filtering techniques (a minor improvement has been found with larger values of N_p); d) the "jittering" technique [2] has been employed in TF to mitigate the so called *depletion problem* in the generation of new particles (in practice, a value larger than the real one is taken for $C_w^{(N)}$ when evaluating $\mathbf{C}_{l,k}^{(g)}$ (127)). Some numerical results are listed in Table 2, which shows the RMSE referring to the linear and the nonlinear portions of the system state when TF #1 or MPF are employed. These results lead to the conclusion that TF #1 performance marginally improves after two iterations and is close to that achieved by MPF. The good accuracy provided by TF #1 is also evidenced by Fig. 9, which shows a realization of the state evolution for the considered dynamic model over 50 consecutive intervals (black curves) and the state estimates evaluated by the TF with $N_{it} = 3$ (red curves). As far as the computational load is concerned, from (144) and (145) it is easily inferred that $N_{it} \cdot C_{TF\#1} = 2.3 \cdot 10^5$ with $N_{it} = 2$ and $C_{MPF} = 2.306 \cdot 10^5$, so that a marginal gap is found (this gap widens substantially as D increases, as evidenced below in Section 0.5.3).

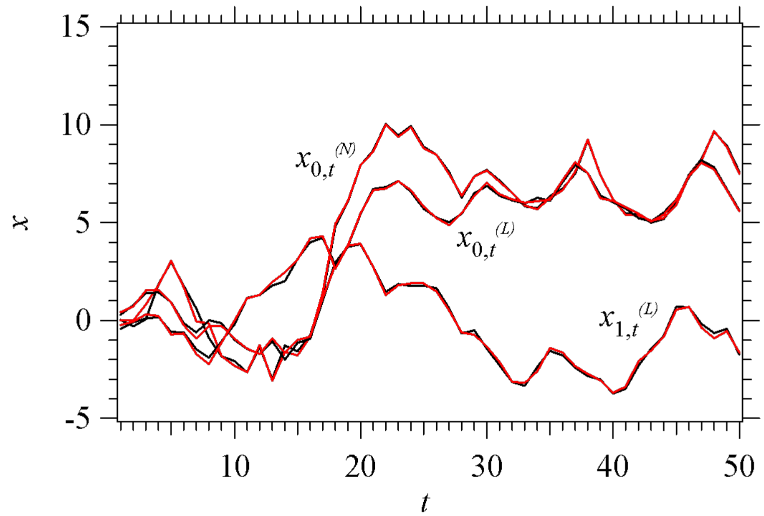

Figure 9: Example of state evolution for the dynamic model (146)-(147) (black curves); the corresponding state estimates evaluated by TF #1 with $N_{it} = 3$ are also shown (red curves).

0.5.2 System B

The second considered system (denoted *system B* in the following) is characterized by: a) the state model

$$x_{t+1}^{(L)} = \cos\left(x_t^{(N)}\right) + 0.5 x_t^{(L)} + w_t^{(L)}, \tag{149}$$

$$x_{t+1}^{(N)} = \sin\left(x_t^{(N)}\right) + 0.5 x_t^{(L)} + w_t^{(N)}, \tag{150}$$

where $w_t^{(L)} \sim \mathcal{N}(0, \mathbf{C}_w^{(L)})$ and $w_t^{(N)} \sim \mathcal{N}(0, C_w^{(N)})$ (consequently, $D = 2$, $D_L = 1$ and $D_N = 1$); b) the measurement model

$$\mathbf{y}_t = \begin{pmatrix} 0.1\left(x_t^{(N)}\right)^2 \cdot \operatorname{sgn}\left(x_t^{(N)}\right) \\ 0 \\ 1+\cos\left(x_t^{(N)}\right) \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} x_t^{(L)} + \mathbf{e}_t, \tag{151}$$

where $\mathbf{e}_t = [e_t^{(L)}, e_t^{(N)}, e_t^{(NL)}]^T$, with $e_t^{(L)} \sim \mathcal{N}(0, C_e^{(L)})$, $e_t^{(N)} \sim \mathcal{N}(0, C_e^{(N)})$, and $e_t^{(NL)} \sim \mathcal{N}(0, C_e^{(NL)})$ (consequently, $P = 3$ and $P_L = P_N = P_{NL} = 1$). In our simulations the *root mean square error* (RMSE) performance provided by TF and MPF for the considered system has been assessed, similarly to the System A (Section 0.5.1), under the following assumptions: a) $C_w^{(L)} = C_w^{(N)} = 10^{-2}$; b) $C_e^{(L)} = C_e^{(N)} = C_e^{(NL)} = 10^{-3}$; c) $N_p = 200$ for both the considered filtering techniques; d) the "jittering" technique [2] has been employed in TF to mitigate the so called *depletion problem* in the generation of new particles (in practice, a value larger than the real one is taken for $C_w^{(N)}$ when evaluating $\mathbf{C}_{l,k}^{(g)}$ (127)). Some numerical results are listed in Tab 3, which shows the RMSE referring to the linear and the nonlinear portions of the system state; MPF and all the developed TF algorithms

	TF #1	TF #1	MPF
	$N_{it} = 2$	$N_{it} = 3$	
$x_t^{(L)}$	0.0455	0.0420	0.0398
$x_t^{(N)}$	0.0274	0.0273	0.0273

(a) TF #1 and MPF.

	TF #3	TF #3	TF #2	TF #2
	$N_{it,1} = 2$	$N_{it,1} = 3$	$N_{it} = 2$	$N_{it} = 3$
	$N_{it,2} = 2$	$N_{it,2} = 2$		
$x_t^{(L)}$	0.0402	0.0395	0.0391	0.0391
$x_t^{(N)}$	0.0273	0.0273	0.0289	0.0289

(b) TF #2 and TF #3.

Table 3: RMSE performance.

have been considered in this case. These results lead to the following conclusions: a) the performance of TF #1 (TF #2) marginally improves (does not improve at all) after two iterations and is very close to that achieved by MPF; b) the TF #3 performance with $N_{it,1} = 3$ and $N_{it,s} = 2$ is even better than provided by MPF. As far as the computational load is concerned, from (144) and (145) it is easily inferred that $N_{it} \cdot C_{TF\#1} = 2.15 \cdot 10^5$ with $N_{it} = 2$ and $C_{MPF} = 2.17 \cdot 10^5$, so that, once again, a marginal gap is found (similar comments hold if TF #2 or TF #3 are considered in place of TF #1).

0.5.3 Other results

Comparing TF # 1 with MPF, the gap previously found in the computational load for the two specific system taken into consideration widens substantially as D increases, as evidenced by Fig. 10. This figure allows us to compare the trend of TF complexity ($N_{it} \cdot C_{TF\#1}$ with $N_{it} = 2$) with that of MPF (C_{MPF}) as D_N increases for three different values of the ratio $R \triangleq D_N/D_L$; in this case $P_{NL} = 1$ has been selected in (144) for simplicity and $N_p = 200$ has been adopted for both algorithms.

0.6 Conclusions

In this manuscript a FG approach to the filtering problem for mixed linear/nonlinear systems has been employed. This has resulted in a novel recursive filtering method, dubbed *turbo filtering*, whose application to the class of linear Gaussian systems has been analysed in detail. Our preliminary results evidence that the performance achieved by this method is close to that of MPF; its complexity, however, can be significantly smaller if the size of the state vector representing the considered system is large. Our ongoing research work concerns the development of turbo filtering algorithms for other classes of systems and its application to specific state estimation problems.

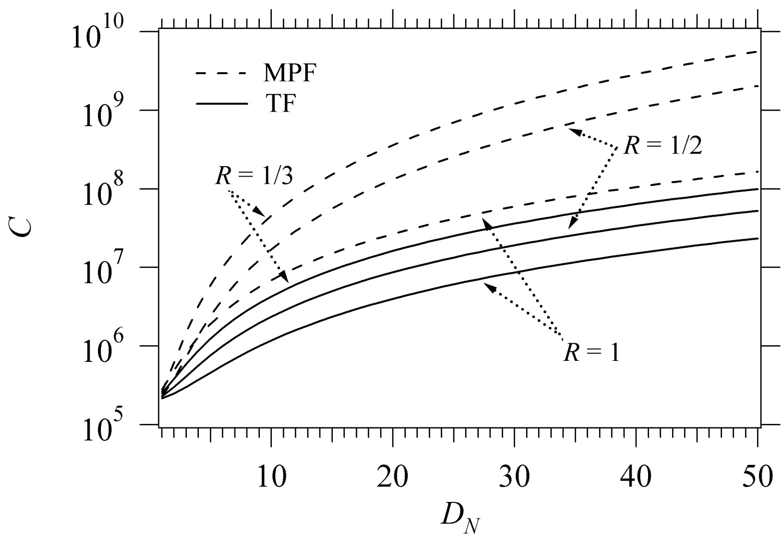

Figure 10: Overall computational complexity of TF #1 (with $N_{it} = 2$) and MPF versus D_N for $R = 1$, 1/2 and 1/3; $N_p = 200$ has been selected for both techniques.

0.7 Appendix A

In this Appendix the derivations of some expressions employed for the evaluation of turbo filtering messages are sketched. Such derivations and those mentioned in the previous Sections are based on the mathematical results summarized in the following four points.

1. If the Gaussian messages $\vec{m}_1(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \eta_1, \mathbf{C}_1)$ and $\vec{m}_2(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \eta_2, \mathbf{C}_2)$ enter an equality node in a FG, the resulting message is $\vec{m}_3(\mathbf{x}) \triangleq \vec{m}_1(\mathbf{x}) \cdot \vec{m}_2(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \eta, \mathbf{C})$, with (see [5, p. 1303, Table II])
$$\mathbf{C}^{-1} = \mathbf{C}_1^{-1} + \mathbf{C}_2^{-1} \tag{152}$$
and
$$\mathbf{C}^{-1}\eta = \mathbf{C}_1^{-1}\eta_1 + \mathbf{C}_2^{-1}\eta_2. \tag{153}$$
Note that: a) (152) and (153) can be rewritten as
$$\mathbf{W} = \mathbf{W}_1 + \mathbf{W}_2 \tag{154}$$
and
$$\mathbf{w} = \mathbf{w}_1 + \mathbf{w}_2, \tag{155}$$
respectively, where $\mathbf{W}_1 \triangleq \mathbf{C}_1^{-1}$, $\mathbf{W}_2 \triangleq \mathbf{C}_2^{-1}$, $\mathbf{W} \triangleq \mathbf{C}^{-1}$, $\mathbf{w}_1 \triangleq \mathbf{W}_1\eta_1$, $\mathbf{w}_2 \triangleq \mathbf{W}_2\eta_2$ and $\mathbf{w} \triangleq \mathbf{W}\eta$.

2. Given the pdf $f(\mathbf{y}) \triangleq \mathcal{N}(\mathbf{y}; \eta_1, \mathbf{C}_1)$ and the conditional pdf $f(\mathbf{y}|\mathbf{x}) \triangleq \mathcal{N}(\mathbf{y}; \mathbf{g}(\mathbf{x}), \mathbf{C}_2)$ for the N-dimensional vector $\mathbf{y}$, where $\mathbf{g}(\mathbf{x})$ is a real valued function of the M-dimensional vector $\mathbf{x}$, it can be proved that
$$\begin{aligned} \int f(\mathbf{y}) \cdot f(\mathbf{y}|\mathbf{x})\, d\mathbf{y} \propto \exp\Big[& -(\mathbf{g}(\mathbf{x}) - \eta_g)^T \\ & \cdot \mathbf{C}_g^{-1} (\mathbf{g}(\mathbf{x}) - \eta_g)\Big] \end{aligned} \tag{156}$$

where

$$\mathbf{C}_g^{-1} = \mathbf{C}_2^{-1}\left[\mathbf{I}_N - \left[\mathbf{C}_2^{-1} + \mathbf{C}_1^{-1}\right]^{-1}\mathbf{C}_2^{-1}\right] \tag{157}$$

and

$$\mathbf{C}_g^{-1}\eta_g = \mathbf{C}_2^{-1}\left[\mathbf{C}_1^{-1} + \mathbf{C}_2^{-1}\right]^{-1}\mathbf{C}_1^{-1}\eta_1. \tag{158}$$

The last two equations can be rewritten as

$$\mathbf{W}_g = \mathbf{W}_2\left[\mathbf{I}_N - [\mathbf{W}_1 + \mathbf{W}_2]^{-1}\mathbf{W}_2\right] \tag{159}$$

and

$$\mathbf{w}_g = \mathbf{W}_2[\mathbf{W}_1 + \mathbf{W}_2]^{-1}\mathbf{w}_1, \tag{160}$$

respectively, where $\mathbf{W}_1 \triangleq \mathbf{C}_1^{-1}$, $\mathbf{W}_2 \triangleq \mathbf{C}_2^{-1}$, $\mathbf{W}_g \triangleq \mathbf{C}_g^{-1}$, $\mathbf{w}_g \triangleq \mathbf{C}_g^{-1}\eta_g$ and $\mathbf{w}_1 \triangleq \mathbf{W}_1\eta_1$. If the function $\mathbf{g}(\mathbf{x})$ exhibits a *linear* dependence on $\mathbf{x}$, i.e.

$$\mathbf{g}(\mathbf{x}) \triangleq \mathbf{g}_0 + \mathbf{G}\mathbf{x}, \tag{161}$$

where $\mathbf{g_0}$ is an N-dimensional vector and $\mathbf{G}$ is an $N \times M$ matrix, it can also be proved that

$$\int f(\mathbf{y}) \cdot f(\mathbf{y}\,|\mathbf{x})\, d\mathbf{y} \propto \mathcal{N}(\mathbf{x}; \eta_x, \mathbf{C}_x) \tag{162}$$

with

$$\mathbf{C}_x^{-1} = \mathbf{G}^T\mathbf{C}_g^{-1}\mathbf{G} \tag{163}$$

and

$$\mathbf{C}_x^{-1}\eta_x = \mathbf{G}^T\mathbf{C}_g^{-1}(\eta_g - \mathbf{g}_0). \tag{164}$$

Similarly as the previous case, the last two equations can be rewritten as (see (157)-(160))

$$\begin{aligned}\mathbf{W}_x &= \mathbf{G}^T\mathbf{W}_g\mathbf{G} \\ &= \mathbf{G}^T\mathbf{W}_2\left[\mathbf{I}_N - [\mathbf{W}_1 + \mathbf{W}_2]^{-1}\mathbf{W}_2\right]\mathbf{G}\end{aligned} \tag{165}$$

and

$$\begin{aligned}\mathbf{w}_x &= \mathbf{G}^T\mathbf{W}_g(\eta_g - \mathbf{g}_0). \\ &= \mathbf{G}^T\mathbf{W}_2[\mathbf{W}_1 + \mathbf{W}_2]^{-1}\mathbf{w}_1 \\ &-\mathbf{G}^T\mathbf{W}_2\left[\mathbf{I}_N - [\mathbf{W}_1 + \mathbf{W}_2]^{-1}\mathbf{W}_2\right]\mathbf{g}_0,\end{aligned} \tag{166}$$

respectively, where $\mathbf{W}_1 \triangleq \mathbf{C}_1^{-1}$, $\mathbf{W}_2 \triangleq \mathbf{C}_2^{-1}$, $\mathbf{W}_x \triangleq \mathbf{C}_x^{-1}$, $\mathbf{w}_x \triangleq \mathbf{C}_x^{-1}\eta_x$ and $\mathbf{w}_1 \triangleq \mathbf{W}_1\eta_1$.

3. Given the pdf $f(\mathbf{x}) \triangleq \mathcal{N}(\mathbf{x}; \eta_1, \mathbf{C}_1)$ for the M-dimensional vector $\mathbf{x}$ and the conditional pdf $f(\mathbf{y}\,|\mathbf{x}) \triangleq \mathcal{N}(\mathbf{y}; \mathbf{g}(\mathbf{x}), \mathbf{C}_2)$ for the N-dimensional vector $\mathbf{y}$, where $\mathbf{g}(\mathbf{x})$ is expressed by (161), it can be proved that (e.g., see [14, Par. 2.3.3]):

$$\int f(\mathbf{x}) \cdot f(\mathbf{y}\,|\mathbf{x})\, d\mathbf{x} = \mathcal{N}(\mathbf{y}; \eta_y, \mathbf{C}_y) \tag{167}$$

with

$$\mathbf{C}_y = \mathbf{C}_2 + \mathbf{G}\mathbf{C}_1\mathbf{G}^T \tag{168}$$

and

$$\eta_y = \mathbf{G}\eta_1 + \mathbf{g}_0. \tag{169}$$

4. Given the pdf $f(\mathbf{x}) \triangleq \mathcal{N}(\mathbf{x}; \eta_1, \mathbf{C}_1)$ for the M-dimensional vector $\mathbf{x}$ and the conditional pdf $f(\mathbf{y}|\mathbf{x}) \triangleq \mathcal{N}(\mathbf{y}; \mathbf{g}(\mathbf{x}), \mathbf{C}_2)$ for the N-dimensional vector $\mathbf{y}$, where $\mathbf{g}(\mathbf{x})$ is expressed by (161), it can be proved that (e.g., see [14, Par. 2.3.3]) the covariance matrix of the joint distribution is:

$$\mathbf{C}_{xy} = \begin{pmatrix} \mathbf{C}_1 & \mathbf{C}_1 \mathbf{G}^T \\ \mathbf{G}\mathbf{C}_1 & \mathbf{C}_2 + \mathbf{G}\mathbf{C}_1\mathbf{G}^T \end{pmatrix}$$

Let us now show how some of these results can be exploited to derive some results not proved in the previous Sections. In particular, we first take into consideration the derivation of (75), (76) and (77); this requires the evaluation of the integral (see (38), (63) and (72))

$$\begin{aligned} &\int \vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(N)}\right) f\left(\tilde{\mathbf{y}}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) d\tilde{\mathbf{y}}_l^{(N)} \\ &= \int \mathcal{N}\left(\tilde{\mathbf{y}}_l^{(N)}; \eta_{\tilde{\mathbf{y}}_{l,k}^{(N)}}, \mathbf{C}_{\tilde{\mathbf{y}}_{l,k}^{(N)}}\right) \\ &\cdot \mathcal{N}\left(\tilde{\mathbf{y}}_l^{(N)}; \mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right), \mathbf{C}_e^{(NL)}\right) d\tilde{\mathbf{y}}_l^{(N)}. \end{aligned} \tag{170}$$

Then, from (156)-(158) it can be easily inferred that

$$\begin{aligned} &\int \vec{m}^{(k)}\left(\tilde{\mathbf{y}}_l^{(N)}\right) f\left(\tilde{\mathbf{y}}_l^{(N)} \middle| \mathbf{x}_l^{(N)}\right) d\tilde{\mathbf{y}}_{N,l} \\ &\propto \exp\left[-\left(\mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right) - \eta_{1,l,k}^{(N)}\right)^T \right. \\ &\left. \cdot \mathbf{W}_{1,l,k}^{(N)}\left(\mathbf{h}_l^{(NL)}\left(\mathbf{x}_l^{(N)}\right) - \eta_{1,l,k}^{(N)}\right)\right], \end{aligned} \tag{171}$$

where $\mathbf{W}_{1,l,k}^{(N)}$ and $\eta_{1,l,k}^{(N)}$ are given by (76) and (77), respectively. Finally, substituting (62) and (171) in (38) yields (75). Note that a similar procedure followed for deriving (75) can be also adopted for the evaluation of the integrals appearing in the RHSs of (85) and (126) (for which, however, the results illustrated at point 3., instead of point 2., are used).

Let us takes into consideration now the derivation of (80), (81) and (82). To begin, we note that (see (80))

$$\mathbf{z}_l^{(N)} \triangleq \mathbf{x}_{l+1}^{(L)} - \mathbf{A}_l^{(L)} \mathbf{x}_l^{(L)}, \tag{172}$$

so that averaging with respect to the the messages $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right)$ and $\vec{m}_{6,G}^{(k-1)}\left(\mathbf{x}_{l+1}^{(L)}\right)$ (see (122), (141) and Fig. 7) produces (81). Consequently, the covariance matrix $\mathbf{C}_{\mathbf{z}_{l,k}^{(N)}}$ of $\mathbf{z}_l^{(N)}$ in the same iteration is given by

$$\begin{aligned} \mathbf{C}_{\mathbf{z}_{l,k}^{(N)}} &= \mathrm{E}\left\{\left[\left(\mathbf{x}_{l+1}^{(L)} - \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)}\right) - \left(\eta_{5,l,k-1}^{(L)} - \mathbf{A}_l^{(L)}\eta_{4,l,k-1}^{(L)}\right)\right] \right. \\ &\left. \cdot \left[\left(\mathbf{x}_{l+1}^{(L)} - \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)}\right) - \left(\eta_{5,l,k-1}^{(L)} - \mathbf{A}_l^{(L)}\eta_{4,l,k-1}^{(L)}\right)\right]^T\right\} \\ &= \mathrm{E}\left\{\left(\mathbf{x}_{l+1}^{(L)} - \eta_{5,l,k-1}^{(L)}\right) - \mathbf{A}_l^{(L)}\left(\mathbf{x}_l^{(L)} - \eta_{4,l,k-1}^{(L)}\right) \right. \\ &\left. \cdot \left[\left(\mathbf{x}_{l+1}^{(L)} - \eta_{5,l,k-1}^{(L)}\right) - \mathbf{A}_l^{(L)}\left(\mathbf{x}_l^{(L)} - \eta_{4,l,k-1}^{(L)}\right)\right]^T\right\} \\ &= \mathbf{C}_{5,l,k-1}^{(L)} + \mathbf{A}_l^{(L)}\mathbf{C}_{4,l,k-1}^{(L)}\left(\mathbf{A}_l^{(L)}\right)^T \\ &- \mathbf{A}_l^{(L)}\left(\mathbf{C}_{l,l+1,k}^{(L)}\right)^T - \mathbf{C}_{l,l+1,k}^{(L)}\left(\mathbf{A}_l^{(L)}\right)^T, \end{aligned} \tag{173}$$

where

$$\mathbf{C}_{l,l+1,k}^{(L)} \triangleq \mathrm{E}\left\{ \left[\mathbf{x}_{l+1}^{(L)} - \eta_{5,l,k-1}^{(L)} \right] \left[\mathbf{x}_{l}^{(L)} - \eta_{4,l,k-1}^{(L)} \right]^T \right\}. \tag{174}$$

Unluckily, the evaluation of $\mathbf{C}_{l,l+1,k}^{(L)}$ (174) requires the knowledge of the joint pdf of the random variables $\mathbf{x}_{l+1}^{(L)}$ and $\mathbf{x}_{l+1}^{(L)}$; note that this pdf cannot inferred from $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right)$ and $\vec{m}_6^{(k-1)}\left(\mathbf{x}_{l+1}^{(L)}\right)$, unless the random variables $\mathbf{x}_l^{(L)}$ and $\mathbf{x}_{l+1}^{(L)}$ are assumed to be independent. A more refined alternative to making the last assumptions is represented by approximating the joint pdf of the random variables $\mathbf{x}_{l+1}^{(L)}$ and $\mathbf{x}_{l+1}^{(L)}$ with the function $f_{l,k}\left(\mathbf{x}_l^{(L)}, \mathbf{x}_{l+1}^{(L)}\right)$ (40), which involves $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(N)}\right)$ (whose flow into the dashed block generating $\mathbf{z}_l^{(N)}$ is not visible in the graph), but is not influenced by $\vec{m}_6^{(k-1)}\left(\mathbf{x}_{l+1}^{(L)}\right)$. In our derivation, the Gaussian projection $\vec{m}_{4,G}^{(k-1)}\left(\mathbf{x}_l^{(N)}\right)$ of $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(N)}\right)$ is employed (see (90)), so that the integral appearing in the RHS of (40) is approximated as

$$\begin{gathered}\int f\left(\mathbf{x}_{l+1}^{(L)} \middle| \mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) \cdot \vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(N)}\right) d\mathbf{x}_l^{(N)} \\ \cong \int \mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \mathbf{f}_l^{(L)}\left(\mathbf{x}_l^{(N)}\right) + \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)}, \mathbf{C}_w^{(L)}\right) \cdot \mathcal{N}\left(\mathbf{x}_l^{(N)}; \eta_{4,l,k-1}^{(N)}, \mathbf{C}_{4,l,k-1}^{(N)}\right) d\mathbf{x}_l^{(N)}.\end{gathered} \tag{175}$$

In the last integral we adopt the further approximation

$$\mathbf{f}_L\left(\mathbf{x}_{N,l}\right) \cong \mathbf{f}_l^{(L)}\left(\hat{\mathbf{x}}_{l,k-1}^{(N)}\right) + \mathbf{J}_{f^{(L)},l,k-1}\left(\mathbf{x}_l^{(N)} - \eta_{4,l,k-1}^{(N)}\right) \tag{176}$$

over the whole integration domain, where $\mathbf{J}_{f^{(L)},l,k-1}$ denotes the Jacobian of $\mathbf{f}_l^{(L)}\left(\mathbf{x}_{N,l}\right)$ evaluated at $\mathbf{x}_l^{(N)} = \eta_{4,l,k-1}^{(N)}$. Exploiting this approximation, the mathematical results illustrated at point 3. (see, in particular, (168) and (169)) in the evaluation of (175) leads to the expression

$$\begin{gathered}\int f\left(\mathbf{x}_{l+1}^{(L)} \middle| \mathbf{x}_l^{(L)}, \mathbf{x}_l^{(N)}\right) \cdot \vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(N)}\right) d\mathbf{x}_l^{(N)} \\ \cong \mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \tilde{\eta}_k\left(\mathbf{x}_{L,l}\right), \tilde{\mathbf{C}}_k^{(L)}\right),\end{gathered} \tag{177}$$

where

$$\begin{aligned}\tilde{\eta}^{(k)}\left(\mathbf{x}_l^{(L)}\right) &= \mathbf{J}_{f^{(L)},l,k-1}\eta_{4,l,k-1}^{(N)} + \mathbf{f}_l^{(L)}\left(\eta_{4,l,k-1}^{(N)}\right) - \mathbf{J}_{f^{(L)},l,k-1}\eta_{4,l,k-1}^{(N)} + \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)} \\ &= \mathbf{f}_l^{(L)}\left(\eta_{4,l,k-1}^{(N)}\right) + \mathbf{A}_l^{(L)}\mathbf{x}_l^{(L)}\end{aligned} \tag{178}$$

and

$$\tilde{\mathbf{C}}_k^{(L)} = \mathbf{C}_w^{(L)} + \mathbf{J}_{f^{(L)},l,k-1}\mathbf{C}_{l,k-1}^{(N)}\left(\mathbf{J}_{f^{(L)},l,k-1}\right)^T. \tag{179}$$

Then, substituting (122) and (177) in the RHS of (40) yields

$$f_{l,k}\left(\mathbf{x}_l^{(L)}, \mathbf{x}_{l+1}^{(L)}\right) \cong \mathcal{N}\left(\mathbf{x}_{L,l}; \eta_{4,L,l}^{(k)}, \mathbf{C}_{4,l,k-1}^{(L)}\right)\mathcal{N}\left(\mathbf{x}_{l+1}^{(L)}; \tilde{\eta}^{(k)}\left(\mathbf{x}_{L,l}\right), \tilde{\mathbf{C}}_k^{(L)}\right) \tag{180}$$

On the basis of the last expression, (178) and the mathematical results illustrated at point 4., it is easy to show that $\mathbf{C}_{l,l+1,k}^{(L)}$ (174) can be approximated as

$$\mathbf{C}_{l,l+1,k}^{(L)} \cong \mathbf{A}_l^{(L)} \mathbf{C}_{4,l,k-1}^{(L)}, \tag{181}$$

so that substituting the last result in (174) produces (82). Note that this result is independent of $\vec{m}_{4,G}^{(k-1)}\left(\mathbf{x}_l^{(N)}\right)$, which, consequently, is not required in the evaluation of $\vec{m}^{(k)}\left(\mathbf{z}_l^{(N)}\right)$ (80) (as shown in Fig. 7).

A similar line of reasoning can be followed for the message $\vec{m}^{(k)}\left(\mathbf{z}_l^{(L)}\right)$ (114), whose evaluation in the k-th iteration involves $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (88) and $\vec{m}_6^{(k-1)}\left(\mathbf{x}_{l+1}^{(N)}\right)$ (130). To simply the derivation of this message, the first order Taylor expansion

$$\mathbf{f}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right) \cong \mathbf{f}_l^{(N)}\left(\eta_{4,l,k}^{(N)}\right) + \mathbf{J}_{f^{(N)},l,k}\left(\mathbf{x}_l^{(N)} - \eta_{4,l,k}^{(N)}\right) \tag{182}$$

is employed, where $\mathbf{J}_{f^{(N)},l,k}$ denotes the Jacobian of $\mathbf{f}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right)$ evaluated at $\mathbf{x}_l^{(N)} = \eta_{4,l,k}^{(N)}$. This allows us to approximate $\mathbf{z}_l^{(L)}$ (20) as

$$\mathbf{z}_l^{(L)} \triangleq \mathbf{x}_{l+1}^{(N)} - \mathbf{f}_{N,l}\left(\mathbf{x}_l^{(N)}\right) \cong \mathbf{x}_{l+1}^{(N)} - \mathbf{f}_l^{(N)}\left(\eta_{4,l,k}^{(N)}\right) - \mathbf{J}_{f^{(N)},l,k}\left(\mathbf{x}_l^{(N)} - \eta_{4,l,k}^{(N)}\right) \tag{183}$$

From the last equation the expression (115) is easily derived for the mean $\eta_{\mathbf{z}_{l,k}^{(L)}}$ of $\mathbf{z}_l^{(L)}$ in the k-th iteration. Moreover, the covariance matrix $\mathbf{C}_{\mathbf{z}_{l,k}^{(L)}}$ of $\mathbf{z}_l^{(L)}$ in the same iteration is given by

$$\begin{aligned}
\mathbf{C}_{\mathbf{z}_{l,k}^{(L)}} &= \mathrm{E}\left\{\left[\left(\mathbf{x}_{l+1}^{(N)} - \eta_{5,l,k-1}^{(N)}\right) - \mathbf{J}_{f^{(N)},l,k}\left(\mathbf{x}_l^{(N)} - \eta_{4,l,k}^{(N)}\right)\right]\right. \\
&\quad \left.\cdot\left[\left(\mathbf{x}_{l+1}^{(N)} - \eta_{5,l,k-1}^{(N)}\right) - \mathbf{J}_{f^{(N)},l,k}\left(\mathbf{x}_l^{(N)} - \eta_{4,l,k}^{(N)}\right)\right]^T\right\} \\
&= \mathbf{C}_{5,l,k-1}^{(N)} + \mathbf{J}_{f^{(N)},l,k}\mathbf{C}_{4,l,k}^{(N)}\left(\mathbf{J}_{f^{(N)},l,k}\right)^T \\
&\quad -\mathbf{C}_{l,l+1,k}^{(N)}\left(\mathbf{J}_{f^{(N)},l,k}\right)^T - \mathbf{J}_{f^{(N)},l,k}\left(\mathbf{C}_{l,l+1,k}^{(N)}\right)^T,
\end{aligned} \tag{184}$$

where

$$\mathbf{C}_{l,l+1,k}^{(N)} \triangleq \mathrm{E}\left\{\left[\mathbf{x}_{l+1}^{(N)} - \eta_{5,l,k-1}^{(N)}\right]\left[\mathbf{x}_l^{(N)} - \eta_{4,l,k}^{(N)}\right]^T\right\} \tag{185}$$

Similarly as $\mathbf{C}_{l,l+1,k}^{(L)}$ (174), the evaluation of $\mathbf{C}_{l,l+1,k}^{(N)}$ (185) requires the knowledge of the joint pdf of the random variables $\mathbf{x}_{l+1}^{(N)}$ and $\mathbf{x}_{l+1}^{(N)}$, which cannot inferred from $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ and $\vec{m}_6^{(k-1)}\left(\mathbf{x}_{l+1}^{(N)}\right)$, unless the random variables $\mathbf{x}_l^{(N)}$ and $\mathbf{x}_{l+1}^{(N)}$ are assumed to be independent. For this reason this joint pdf is approximated with the function $\tilde{f}_{l,k}\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right)$ (45), which involves $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right)$ (whose flow into the dashed block generating $\mathbf{z}_l^{(L)}$ is not visible in the graph), but is not influenced by $\vec{m}_6^{(k-1)}\left(\mathbf{x}_{l+1}^{(N)}\right)$. Let us focus now on the integral

$$\int f\left(\mathbf{x}_{l+1}^{(N)} \middle| \mathbf{x}_l^{(N)}, \mathbf{x}_l^{(L)}\right) \cdot \vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right) d\mathbf{x}_l^{(L)}, \tag{186}$$

appearing in the RHS of (45). Substituting (65) and (122) in this integral and exploiting the results illustrated at point 3. yields

$$\int \mathcal{N}\left(\mathbf{x}_{l+1}^{(N)}; \mathbf{f}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right) + \mathbf{A}_l^{(N)}\mathbf{x}_l^{(L)}, \mathbf{C}_w^{(N)}\right) \cdot \mathcal{N}\left(\mathbf{x}_l^{(L)}; \eta_{4,l,k-1}^{(L)}, \mathbf{C}_{4,l,k-1}^{(L)}\right) d\mathbf{x}_l^{(L)} \\ = \mathcal{N}\left(\mathbf{x}_{l+1}^{(N)}; \tilde{\eta}^{(k)}\left(\mathbf{x}_l^{(N)}\right), \tilde{\mathbf{C}}_k^{(N)}\right) \tag{187}$$

where

$$\tilde{\eta}^{(k)}\left(\mathbf{x}_l^{(N)}\right) = \mathbf{A}_l^{(N)}\eta_{4,l,k-1}^{(L)} + \mathbf{f}_l^{(N)}\left(\mathbf{x}_l^{(N)}\right) \tag{188}$$

and

$$\tilde{\mathbf{C}}_k^{(N)} = \mathbf{C}_w^{(N)} + \mathbf{A}_l^{(N)}\mathbf{C}_{4,l,k-1}^{(L)}\left(\mathbf{A}_l^{(N)}\right)^T. \tag{189}$$

Since a Gaussian form is desired for $\vec{m}^{(k)}\left(\mathbf{z}_l^{(L)}\right)$, the Gaussian projection $\vec{m}_{4,G}^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (90) of $\vec{m}_4^{(k)}\left(\mathbf{x}_l^{(N)}\right)$ (88) is employed in the evaluation of $\tilde{f}_{l,k}\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right)$ (45). Then, substituting (90) and (187) in (45) yields

$$\tilde{f}_{l,k}\left(\mathbf{x}_l^{(N)}, \mathbf{x}_{l+1}^{(N)}\right) \cong \mathcal{N}\left(\mathbf{x}_l^{(N)}; \eta_{4,l,k}^{(N)}, \mathbf{C}_{4,l,k}^{(N)}\right) \mathcal{N}\left(\mathbf{x}_{l+1}^{(N)}; \tilde{\eta}^{(k)}\left(\mathbf{x}_l^{(N)}\right), \tilde{\mathbf{C}}_k^{(N)}\right), \tag{190}$$

Finally, based on the last expression, (188) and the mathematical results illustrated at point 4., it is easy to show that $\mathbf{C}_{l,l+1,k}^{(N)}$ (174) can be approximated as

$$\mathbf{C}_{l,l+1,k}^{(N)} \cong \mathbf{J}_{f^{(N)},l,k}\mathbf{C}_{4,l,k}^{(N)} \tag{191}$$

Finally, substituting the last result in (184) yields (116); note that this expression is independent of $\vec{m}_4^{(k-1)}\left(\mathbf{x}_l^{(L)}\right)$, which, consequently, is not required in the evaluation of $\vec{m}^{(k)}\left(\mathbf{z}_l^{(L)}\right)$ (114) (as shown in Fig. 7).

Bibliography

[1] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson and P. Nordlund, "Particle filters for positioning, navigation, and tracking", *IEEE Trans. Sig. Proc.*, vol. 50, 425–435, 2002.

[2] F. Gustafsson, "Particle filter theory and practice with positioning applications", *IEEE Aerospace and Electronic Syst. Mag.*, vol. 25, no. 7, pp. 53-82, July 2010.

[3] F. Daum and J. Huang, "Curse of dimensionality and particle filters", Proc. of the IEEE Aerospace Conference, vol.4, pp. 1979-1993, March 2003.

[4] J. Dauwels, S. Korl and H.-A. Loeliger, "Particle Methods as Message Passing", *Proc. of the 2006 IEEE Int. Symp. on Inf. Theory*, pp.2 052-2056, 9-14 July 2006.

[5] H.-A. Loeliger, J. Dauwels, Junli Hu, S. Korl, Li Ping, F. R. Kschischang, "The Factor Graph Approach to Model-Based Signal Processing", *IEEE Proc.*, vol. 95, no. 6, pp. 1295-1322, June 2007.

[6] T. Schon, F. Gustafsson, P.-J. Nordlund, "Marginalized particle filters for mixed linear/nonlinear state-space models", *IEEE Trans. Sig. Proc.*, vol.53, no.7, pp. 2279-2289, July 2005.

[7] M. S. Arulampalam, S. Maskell, N. Gordon and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking", *IEEE Trans. Sig. Proc.*, vol. 50, no. 2, pp. 174-188, Feb. 2002.

[8] F. Lindsten, P. Bunch, S. J. Godsill, T. B. Schon, "Rao-Blackwellized particle smoothers for mixed linear/nonlinear state-space models", *Proc. of the 2013 IEEE Int. Conf. on Ac., Speech and Sig. Proc.* (ICASSP 2013), pp.6288-6292, 26-31 May 2013.

[9] F. R. Kschischang, B. Frey, and H. Loeliger, "Factor graphs and the sum-product algorithm", *IEEE Trans. Inf. Theory*, vol. 41, no. 2, pp. 498–519, Feb. 2001.

[10] F. R. Kschischang and B. J. Frey, "Iterative decoding of compound codes by probability propagation in graphical models", *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 219–230, Feb. 1998.

[11] J. Hagenauer, "The turbo principle: Tutorial introduction & state of the art", *Proc. Int. Symp. Turbo Codes & Related Topics*, Brest, France, Sep. 1997, pp. 1–11.

[12] A. P. Worthen and W. E. Stark, "Unified design of iterative receivers using factor graphs," *IEEE Trans. Veh. Tech.*, vol. 47, no. 2, pp. 843–849, Feb. 2001.

[13] S. Benedetto, D. Divsalar, G. Montorsi and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design, and iterative decoding", *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 909-926, May 1998.

[14] C. M. Bishop, **Pattern Recognition and Machine Learning**, Springer, 2006.

[15] S. Mazuelas, Y. Shen and M. Z. Win, "Belief Condensation Filtering", *IEEE Trans. Sig. Proc.*, vol. 61, no. 18, pp. 4403-4415, Sept. 2013.

[16] A. R. Runnalls, "Kullback-Leibler Approach to Gaussian Mixture Reduction", *IEEE Trans. on Aerospace and Electronic Syst.*, vol. 43, no. 3, pp. 989-999, July 2007.

[17] R. Karlsson, T. Schon, F. Gustafsson, "Complexity analysis of the marginalized particle filter", *IEEE Trans. Sig. Proc.*, vol. 53, no. 11, pp. 4408-4411, Nov. 2005.

[18] F. E. Daum, "Exact finite-dimensional nonlinear filters", *IEEE Tran. Aut. Contr.*, vol. 31, no. 7, pp. 616-622, July 1986.

[19] A. Doucet, J. F. G. de Freitas and N. J. Gordon, "An introduction to sequential Monte Carlo methods," in **Sequential Monte Carlo Methods in Practice**, A. Doucet, J. F. G. de Freitas, and N. J. Gordon, Eds. New York: Springer-Verlag, 2001.

[20] A. Doucet, S. Godsill and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering", *Statist. Comput.*, vol. 10, no. 3, pp. 197–208, 2000.

[21] V. Smidl and A. Quinn, "Variational Bayesian Filtering", *IEEE Trans. Sig. Proc.*, vol. 56, no. 10, pp.5020-5030, Oct. 2008.

[22] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation", *IEEE Proc.*, vol. 92, no. 3, pp. 401-422, Mar. 2004.

[23] B. Anderson and J. Moore, **Optimal Filtering**, Englewood Cliffs, NJ, Prentice-Hall, 1979.

[24] T. P. Minka, "Expectation propagation for approximate Bayesian inference", *Proc. 17th Annual Conf. Uncertainty in Artif. Intell.*, pp. 362–369, Seattle, WA, USA, Aug. 2001.

[25] R. Koetter, A. C. Singer and M. Tüchler, "Turbo equalization", *IEEE Signal Processing Magazine*, vol. 21, no. 1, pp. 67-80, Jan. 2004.

[26] M. Šimandl, J. Královeca and T. Söderströmc, "Advanced point-mass method for nonlinear state estimation", Automatica, vol. 42, pp. 1133 – 1145, 2006.

[27] R. Bucy and K. Senne, "Digital synthesis of non-linear filters", Automatica, vol. 7, no. 3, pp. 287–298, 1971.

[28] F. Mustiere, M. Bolic and M. Bouchard, "A Modified Rao-Blackwellised Particle Filter", Proc. of the 2006 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2006), vol. 3, 14-19 May 2006.

[29] V. Smídl and A. Quinn, **The Variational Bayes Method in Signal Processing**, Berlin, Germany, Springer, 2005.

[30] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes", IEEE Transactions on Communications, vol. 44, no. 10, pp. 1261 - 1271, Oct. 1996.

[31] T. Li, M. Bolic, P. Djuric, "Resampling Methods for Particle Filtering: Classification, implementation, and strategies", *IEEE Signal Processing Magazine*, vol.32, no.3, pp.70-86, May 2015.