

# Direct Tracking of a Wireless Transmitter based on Rao-Blackwellized Point Mass Filter

Evert Pocoma, *ULB* and Laurent Storrer, *ULB* and François Quitin, *ULB* and Luc Vandendorpe, *UCL* and Philippe De Doncker, *ULB* and François Horlin, *ULB*

**Abstract**—In this paper, we address the problem of direct tracking of a wireless transmitter. That is, the inputs given to the Bayesian filter are the received baseband signals instead of pre-computed ranges or angles. We first propose to use the Rao-Blackwellized Point Mass Filter (RBPMF) to solve such a tracking problem. As such, the resulting tracking solution is still computationally expensive. Therefore, we propose an approach for reducing the computational cost of the RBPMF. More precisely, we replace the prediction step by the one of the Linear Kalman Filter (LKF). This combination helps to avoid expensive operations such as the weight convolution in the prediction step. In addition, it also allows complexity reductions in the correction step. As a result, the complexity is reduced by one order of magnitude compared to the original RBPMF. We compare our approach to representative direct-tracking methods, based on Iterative Extended Kalman Filter (IEKF) and Particle Filter (PF). The proposed solution has lower and comparable localization error compared to IEKF and PF, respectively. In addition, the proposed solution is of slightly less complexity than PF. However, the complexity reduction is significant compared to the conventional RBPMF.

**Index Terms**—Tracking, Direct positioning, Rao-Blackwellized Point Mass Filter, Bayesian Filter.

## I. INTRODUCTION

WHEN a Mobile Terminal (MT) transmits a signal, the corresponding received signal at the base station (BS) contains information that can be used to localize the MT. Such a localization challenge has been thoroughly studied for many decades and used in many applications. As a result, device localization has become an essential functionality of 5G/6G networks [1], [2]. For instance, a positioning reference signal (PRS) is included in the protocols to support localization based on the estimation of time-of-arrival (ToA) [3]. Tracking the evolution of the MT position over time can be used afterwards for different applications, such as smart cities [4], traffic flow control, autonomous driving [5], [6], etc. Therefore, the tracking problem of a MT is a well motivated topic and the focus herein.

On the one hand, the received signals are characterized by the signal model which accounts mainly for the signal propagation through the wireless channel. On the other

hand, some intermediate parameters (IPs) can be estimated from the received signals with the goal of squeezing the useful information into a reduced number of parameters. Nevertheless, part of the useful information is lost when performing such estimation at each individual BS [7]. Either the received signals or the estimated IPs can be passed as measurements to the Bayesian filter. Consequently, the tracking problem of a wireless transmitter can be categorized as direct or indirect.

Direct approaches perform tracking using the received signals; whereas, indirect approaches use IPs, such as ToA or angle of arrival (AoA). Consequently, indirect approaches result in a simpler sensor model compared to the direct case. For instance, the sensor model for ToA involves quadratic and squared-root functions; whereas in the direct case, it also involves complex exponential functions. For such a reason, indirect tracking is commonly chosen and even silently assumed in several studies [8]. However, the estimation of IPs becomes unreliable as the signal-to-noise-ratio (SNR) decreases [9], [10]. Moreover as pointed out in [9], the measurement noise is no longer Gaussian distributed which leads to a greater degradation of tracking performance when using indirect approaches. In summary, direct approaches outperform indirect ones as shown in [11]–[13], and proven in [7]. Nevertheless, there are just a few studies addressing direct tracking in the literature from which we consider [9], [11] and [14] as the most representative.

The approach in [9] is based on the Iterative Extended Kalman Filter (KF) (KF) that relies on the iterative Gauss-Newton method. First, it linearizes the system around the current state vector at every single iteration. Second, it numerically maximizes the linearized log-posterior distribution, whose solution defines the new state vector estimate. As seen in [10], the log-posterior distribution presents several local maxima and minima, specially at low SNR values. Therefore, not-sampled approaches, such as IEKF, can lead to incorrect solutions causing the filter to diverge. For this reason, we prefer to study sampled over not-sampled filtering approaches.

The two main sampled-filtering methods are Point Mass Filter (PMF) and Particle Filter (PF). PF relies on stochastic sampling and PMF does it on deterministic sampling [15]. PMF requires a much higher computational effort than PF due to the weight convolution problem, which is present at

Manuscript received March 17, 2022; revised November 29, 2022, and April 23, 2023; accepted June 15, 2023. (Corresponding author: E. Pocoma)

E. Pocoma, L. Storrer, F. Quitin, P. De Doncker, and F. Horlin are with the Université Libre de Bruxelles, Brussels 1050, Belgium (e-mail:epocomac@ulb.be; lstorrer@ulb.be; fquitin@ulb.ac.be; philippe.dedoncker@ulb.be; francois.horlin@ulb.be).

L. Vandendorpe is with the Université Catholique de Louvain, 1348 Ottignies-Louvain-la-Neuve, Belgium (e-mail:luc.vandendorpe@uclouvain.be)

the prediction step<sup>1</sup> [16]. Therefore, PF has been preferred in the direct-tracking literature, e.g., [11] and [14]. In [11], PF is directly applied to track a wireless emitter based on the delay and Doppler information contained in the received signal. In [14], PF is used to track multiple targets in a scene based directly on the signals from multiple radars.

Setting aside the computational complexity, PF presents some shortcomings like the degeneracy problem, sample impoverishment, laborious tuning, etc., which requires special care as shown in [17]; whereas PMF does not suffer from those shortcomings due to the use of deterministic sampling. Moreover, PMF has been well adopted for tracking applications with highly nonlinear systems in the context of Terrain Aided Navigation (TAN) [18]. Therefore, we choose to study PMF with special focus on the reduction of complexity. As first measure, we decide to study Rao-Blackwellized PMF (RBPMPF), which is derived from PMF. It reduces the sampling space by using a special factorization denoted as Rao-blackwellization [19]. Since RBPMPF requires less samples, it also reduces the complexity compared to PMF [20]. RBPMPF has been also broadly used in the context of TAN, e.g., [16], [20], [21]. However, it has not yet been explored in the context of direct tracking of a wireless transmitter.

To further motivate our choice, we mention two possible improvements that are applicable to RBPMPF but not to PF. Both possible improvements are related to the fact that RBPMPF can work with a fixed grid of points (due to deterministic sampling); whereas PF obtains new grid-points at every iteration. First, the complexity can be further reduced by considering the possibility of pre-computing certain operations. For instance, the received signal model must be computed at all grid points; and thus for a fixed grid, such an operation could be done only once. Second, a possible straightforward extension to the distributed scenario is possible by using the method introduced in [22], such a method makes it possible to gather the whole measurement likelihood information at each BS. However, it requires a fixed grid of samples. From the two possible improvements, only the first one is discussed in this paper, since the second one deviates significantly from the paper scope.

The contributions of this paper are threefold and can be summarized as follows:

- We propose to use RBPMPF for the tracking problem of a wireless transmitter. The resulting tracking approach is direct, i.e., based on the received baseband signals. As a result, the tracking solution has better localization performance compared to indirect approaches. The localization gain is more evident specially at low SNR values. In addition, it is more accurate than not-sampled approaches such as IEKF and yields a comparable performance to the PF.
- RBPMPF as such is computationally expensive. Therefore, we propose to replace the prediction step of RBPMPF, since it is the most costly step of the algorithm. More precisely, we combine the prediction step of the

Linear KF (KF) with the correction step of RBPMPF. The resulting approach, denoted as Reduced-Complexity (RC)-RBPMPF, reduces the computational complexity of the original RBPMPF in such a way that its computational complexity is comparable/competitive to the one of PF.

- Lastly, we compare the performance of our approach against two representative direct-tracking solutions, IEKF and PF. The comparison is done in terms of localization error and complexity using Monte-Carlo simulations for 1000 realizations.

The remainder of the paper is organized as follows. Section II describes the preliminaries and system model. Section III details RBPMPF in the context of tracking of a wireless transmitter. Section IV describes the complexity reduction approach. Section V first presents simulation results for the localization performance. Later, it discusses complexity reduction results. Finally, Section VI concludes the paper.

*Notation:* Vectors and matrices are shown in bold lowercase and uppercase letters, respectively.  $\delta(\cdot)$  is the Dirac delta function.  $\mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}, \Sigma)$  denotes a Normal distribution, with mean  $\hat{\mathbf{x}}$  and covariance  $\Sigma$ , evaluated at  $\mathbf{x}$ . Superscript  $(\cdot)^{(j)}$  denotes a grid-point- $j$ .  $\mathbf{I}_P$  and  $\mathbf{0}_P$  are the  $P \times P$ -identity and zero matrices, respectively.  $(\|\cdot\|_2)$  denotes the L-2 norm of a vector.

## II. SYSTEM MODEL AND PRELIMINARIES

For the sake of clarity, we start by formally defining the system model considered in our study, i.e., the measurement model followed by the process model. We also describe shortly the Bayesian filtering that makes use of such models, as well as the Rao-Blackwellized (RB) factorization.

### A. System Model

As usually done in the RBPMPF literature [16], [19], [20], [23], the considered system model is defined at time instant- $k$  as:

$$\begin{bmatrix} \mathbf{x}_{k+1}^n \\ \mathbf{x}_{k+1}^\ell \end{bmatrix} = \begin{bmatrix} \mathbf{F}_k^n & \mathbf{F}_k^\ell \\ \mathbf{F}_k^n & \mathbf{F}_k^\ell \end{bmatrix} \begin{bmatrix} \mathbf{x}_k^n \\ \mathbf{x}_k^\ell \end{bmatrix} + \begin{bmatrix} \mathbf{w}_k^n \\ \mathbf{w}_k^\ell \end{bmatrix} \quad (1)$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k^n) + \mathbf{H}_k \mathbf{x}_k^\ell + \mathbf{v}_k \quad (2)$$

where  $\mathbf{F}_k$  is the transition matrix,  $\mathbf{x}_k$  and  $\mathbf{z}_k$  are, respectively, the state and measurement vectors, and  $\mathbf{w}_k$  and  $\mathbf{v}_k$  are zero-mean additive noises with respective covariance matrices  $\mathbf{Q}_w$  and  $\mathbf{Q}_v$ . The measurement linear matrix and nonlinear function are denoted by  $\mathbf{H}_k$  and  $\mathbf{h}(\cdot)$ , respectively.

The state vector  $\mathbf{x}_k$  is split in two sub-vectors  $\mathbf{x}_k^n$  and  $\mathbf{x}_k^\ell$ , that are respectively denoted as nonlinear and linear state variables. Notice that  $(\cdot)^n$  and  $(\cdot)^\ell$  denote the vectors and matrices associated to nonlinear and linear parts, respectively. Since such a model has been used in the TAN literature, we proceed to detail the considerations for its use in the context of tracking of a wireless transmitter.

<sup>1</sup>PF avoids the weight convolution by tracking the state trajectories rather than the state itself [15]-Section-III-A.

### Measurement Model

At time instant- $k$ , the location of a wireless transmitter in Cartesian coordinates is  $(x_k, y_k)$ . Such a transmitter communicates with a fixed infrastructure of  $N$  single-antenna BSs. The communication takes place using orthogonal frequency-division multiplexing (OFDM) modulation which divides the communication bandwidth in  $P$  equispaced pilots. Consequently, BS- $i$  receives the baseband signal  $z_{k,i,p}$  defined at pilot- $p$  and time instant- $k$ :

$$z_{k,i,p} = h_{i,p}(\mathbf{x}_k) + v_{k,i,p}, \quad (3)$$

where  $v_{k,i,p}$  is the corrupting noise assumed to be independent zero mean circular symmetric complex Gaussian of variance  $\sigma_{v_{k,i}}^2$ . Function  $h_{i,p}(\cdot)$  performs a mapping from state space to measurement space which is defined as:

$$h_{i,p}(\mathbf{x}_k) = a_i e^{\phi_i} s_p e^{-j p \zeta d_i(x_k, y_k)}, \quad (4)$$

where  $s_p$  is the symbol transmitted on pilot- $p$  and  $d_i(x_k, y_k)$  is the transmitter-to-BS distance. Notice that the transmitter is assumed to be time synchronized with the infrastructure.  $\zeta = \frac{2\pi \Delta f}{c}$  is a constant with  $\Delta f$  and  $c$  being respectively the pilot spacing and the propagation velocity.  $a_i$  and  $\phi_i$  are respectively the amplitude and remaining carrier frequency offset (CFO). However, as shown in [24] and [13], one can marginalize out the nuisance parameters  $(a_i, \phi_i)$  from the likelihood of the received signal; therefore, we choose not to take them into account to avoid over-complicating the analysis herein.

The vectorized measurement model at time instant- $k$  and BS- $i$  is:

$$\mathbf{z}_{k,i} = \mathbf{h}_i(\mathbf{x}_k) + \mathbf{v}_{k,i}, \quad (5)$$

with:  $\mathbf{z}_{k,i} = [z_{k,i,1}, \dots, z_{k,i,P}]^T$ ;  $\mathbf{v}_{k,i} = [v_{k,i,1}, \dots, v_{k,i,P}]^T$ ,  
 $\mathbf{h}_i(\mathbf{x}_k) = [h_{i,1}(\mathbf{x}_k), \dots, h_{i,P}(\mathbf{x}_k)]^T$ .

Since CFO is assumed to be removed/compensated before any processing, the received signals only carry position information as shown in (4). Hence, the final measurement model results from gathering the received signals of all BSs as:

$$\mathbf{z}_k = \mathbf{h}(x_k, y_k) + \mathbf{v}_k, \quad (6)$$

with  $\mathbf{z}_k = [z_{k,1}, \dots, z_{k,N}]^T$ ,  $\mathbf{v}_k = [v_{k,1}, \dots, v_{k,N}]^T$  and  $\mathbf{h}(x_k, y_k) = [h_1(x_k, y_k), \dots, h_N(x_k, y_k)]^T$ . By comparing (6) to (2), we choose to define the nonlinear part as  $\mathbf{x}_k^n$  as the transmitter position, i.e.,  $\mathbf{x}_k^n = [x_k, y_k]^T$  and  $\mathbf{H}_k = \mathbf{0}$ . This also means that  $p(\mathbf{z}_k | \mathbf{x}_k) = p(\mathbf{z}_k | \mathbf{x}_k^n, \mathbf{x}_k^\ell) = p(\mathbf{z}_k | \mathbf{x}_k^n)$

Lastly, the received signals at different BSs are assumed to be uncorrelated. Therefore, the measurement likelihood is:

$$p(\mathbf{z}_k | \mathbf{x}_k^n) = \prod_{i=1}^N \mathcal{N}(\mathbf{z}_k; \mathbf{h}_i(\mathbf{x}_k^n), \sigma_{v_{k,i}}^2 \mathbf{I}_P), \quad (7)$$

Notice that the measurement function  $\mathbf{h}_i(\cdot)$  is highly nonlinear; thus, the likelihood  $p(\mathbf{z}_k | \mathbf{x}_k)$  presents multiple local maxima and minima that are more pronounced as the SNR decreases [10].

### Process Model

We focus on motion models that can be expressed in the form shown in (1) such as linear and curvilinear models<sup>2</sup>. Linear models, such as Constant Velocity (CV) and Constant Acceleration (CA), are overly optimistic yet used broadly in the tracking literature [8], [26]. Similarly, Curvilinear Models, such as Constant Turn Rate and Velocity (CTRV), are important for vehicle tracking as shown in [6], [25].

In the previous section, we defined the nonlinear part  $\mathbf{x}_k^n$ . However, the definition of  $\mathbf{x}_k^\ell$ ,  $\mathbf{F}_k$  and  $\mathbf{Q}_w$  depends on the chosen motion model. For instance, we present such vectors and matrices in Appendix-A for the CV and CTRV model. Lastly, it is possible to define the transition probability density function (PDF) with (1) as:

$$p(\mathbf{x}_{k+1} | \mathbf{x}_k) = \mathcal{N}(\mathbf{x}_{k+1}; \mathbf{F}\mathbf{x}_k, \mathbf{Q}_w) \quad (8)$$

### B. Bayesian Filtering

In short, the Bayesian filtering approach estimates the state vector  $\mathbf{x}_k$  by recursively iterating between:

$$p(\mathbf{x}_k | \mathbb{Z}_{k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbb{Z}_{k-1}) d\mathbf{x}_{k-1} \quad (9)$$

$$p(\mathbf{x}_k | \mathbb{Z}_k) = \frac{p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbb{Z}_{k-1})}{p(\mathbf{z}_k | \mathbb{Z}_{k-1})}, \quad (10)$$

where  $\mathbb{Z}_k = \{\mathbf{z}_0, \dots, \mathbf{z}_k\}$  is the set of measurements up to time instant  $k$ . Such expressions are respectively denoted as the prediction and correction steps. Notice that the transition PDF  $p(\mathbf{x}_k | \mathbf{x}_{k-1})$  and measurement PDF  $p(\mathbf{z}_k | \mathbf{x}_k)$  are defined by (8) and (7), respectively. Besides, Fig. 1(a) illustrates the Bayesian recursion.

Sampling methods are one of the approaches used to solve (9) and (10). Such methods numerically compute the desired PDFs through the use of sampling. For instance, PF uses stochastic sampling, whereas PMF uses deterministic sampling. Both approaches perform sampling in the state-vector space. However, in some cases the measurement model might depend on a subset of the state variables. In such cases, one could use RBPMF that reduces the sampling space by using a special factorization described in the next section.

### C. Rao Blackwellization

It requires to split the state vector  $\mathbf{x}_k$  into a nonlinear part  $\mathbf{x}_k^n$  and a linear part  $\mathbf{x}_k^\ell$ . Intuitively, such a splitting allows one to reduce the sampling space to the one of  $\mathbf{x}_k^n$ , which is of lower dimension than the state vector  $\mathbf{x}_k$ . Such reduction not only decreases the computational cost, but also increases the filter accuracy compared to PF or PMF [23].

We derive the RB factorization of a generic distribution  $p(\mathbf{x}_k)$  that can be factorized as:

$$p(\mathbf{x}_k) = p(\mathbf{x}_k^\ell, \mathbf{x}_k^n) = p(\mathbf{x}_k^\ell | \mathbf{x}_k^n) p(\mathbf{x}_k^n). \quad (11)$$

<sup>2</sup>Since Curvilinear Models are nonlinear, they need to be first linearized around the current state vector as shown in [25].

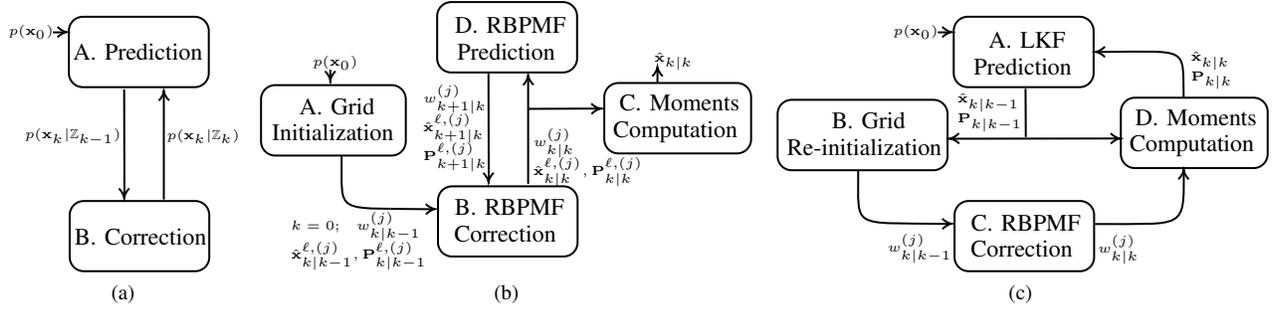


Fig. 1. Block Diagrams of the considered Filters for the problem of tracking of a single transmitter a) Bayesian Filtering b)RBPMF c) RC-RBPMF.

The marginal PDF  $p(\mathbf{x}_k^n)$  is approximated through deterministic sampling as:

$$p(\mathbf{x}_k^n) \approx \sum_{j=1}^{N_s} w_k^{(j)} \delta(\mathbf{x}_k^n - \mathbf{x}_k^{n,(j)}), \quad (12)$$

where  $N_s$  is the number of grid points located at  $\mathbf{x}_k^{n,(j)}$ . Notice that the sampling is done only on the nonlinear part. In addition,  $w_k^{(j)} = p(\mathbf{x}_k^{n,(j)})$ . Replacing (12) into (11) results in (13) which herein will be referred to as RB factorization.

$$p(\mathbf{x}_k) = \sum_{j=1}^{N_s} p(\mathbf{x}_k^\ell | \mathbf{x}_k^{n,(j)}) w_k^{(j)} \delta(\mathbf{x}_k^n - \mathbf{x}_k^{n,(j)}). \quad (13)$$

Second, the distribution of the linear part associated to grid point- $j$  is usually assumed Gaussian distributed [20], i.e.,

$$p(\mathbf{x}_k^\ell | \mathbf{x}_k^{n,(j)}) = \mathcal{N}(\mathbf{x}_k^\ell; \hat{\mathbf{x}}_k^{\ell,(j)}, \mathbf{P}_k^{\ell,(j)}), \quad (14)$$

where  $\hat{\mathbf{x}}_k^{\ell,(j)}$  and  $\mathbf{P}_k^{\ell,(j)}$  are, respectively, the mean vector and covariance matrix of the linear part. In summary, the RB factorization of  $p(\mathbf{x}_k)$ , shown in (13), is completely defined at each grid point- $j$  by the three parameters  $(w_k^{(j)}, \hat{\mathbf{x}}_k^{\ell,(j)}, \mathbf{P}_k^{\ell,(j)})$ .

Consequently, the procedure explained in (11)-(14) can be applied to any distribution dependent of  $\mathbf{x}_k$ . Moreover, notice that the predicted and posterior PDFs used in the following sections are expressed in the RB factorization.

### III. RBPMF FOR DIRECT TRACKING OF A WIRELESS TRANSMITTER

With the definition of the process and measurement models in (8) and (7), RBPMF can be applied directly to our study case<sup>3</sup>. Therefore, rather than presenting an RBPMF derivation, we provide a short and intuitive explanation for each step highlighting the necessary considerations regarding our system model. RBPMF expresses all PDFs using RB factorization. Therefore it is enough to define  $(w_{k|k-1}^{(j)}, \hat{\mathbf{x}}_{k|k-1}^{\ell,(j)}, \mathbf{P}_{k|k-1}^{\ell,(j)})$  and  $(w_{k|k}^{(j)}, \hat{\mathbf{x}}_{k|k}^{\ell,(j)}, \mathbf{P}_{k|k}^{\ell,(j)})$  to represent, respectively, the predicted and corrected PDFs. An illustration of the RBPMF recursion is shown in Fig. 1(b) and summarized in Algorithm-1. Notice that the letters in Fig. 1 indicate the execution order being (A) the first step to be executed.

<sup>3</sup>RBPMF was derived in the context of TAN systems considering general nonlinear process and measurement models in [16], [20].

#### Algorithm 1 : RBPMF for Direct Tracking

##### A. Initialization:

with  $k=0$  and  $p(\mathbf{x}_k | \mathbb{Z}_{k-1}) = \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_0, \mathbf{P}_0)$ , with  $(\hat{\mathbf{x}}_0, \mathbf{P}_0)$  defined in (27), initialize:

*Non-linear part:*

$$w_{0|-1}^{(j)} = \mathcal{N}(\mathbf{x}_0^{n,(j)}; \hat{\mathbf{x}}_0^n, \mathbf{P}_0^n) \quad (15)$$

*Linear part:*

$$\hat{\mathbf{x}}_{0|-1}^{\ell,(j)} = \hat{\mathbf{x}}_0^\ell - \mathbf{P}_0^{\ell n} (\mathbf{P}_0^n)^{-1} (\mathbf{x}_0^{n,(j)} - \hat{\mathbf{x}}_0^n) \quad (16a)$$

$$\mathbf{P}_{0|-1}^{\ell,(j)} = \mathbf{P}_0^\ell - \mathbf{P}_0^{\ell n} (\mathbf{P}_0^n)^{-1} \mathbf{P}_0^{\ell n}. \quad (16b)$$

##### B. Correction Step: with the new measurement $\mathbf{z}_k$ .

*Non-linear part:*

$$w_{k|k}^{(j)} = \frac{w_{k|k-1}^{(j)} \xi_k^{(j)}}{\sum_i w_{k|k-1}^{(i)} \xi_k^{(i)}}; \quad \xi_k^{(i)} = p(\mathbf{z}_k | \mathbf{x}_k^n = \mathbf{x}_k^{n,(i)}) \quad (17)$$

*Linear part:*

$$\hat{\mathbf{x}}_{k|k}^{\ell,(j)} = \hat{\mathbf{x}}_{k|k-1}^{\ell,(j)} \quad ; \quad \mathbf{P}_{k|k}^{\ell,(j)} = \mathbf{P}_{k|k-1}^{\ell,(j)} \quad (18)$$

##### C. Moment Computation:

*First order moments:*

$$\hat{\mathbf{x}}_{k|k}^n = \sum_j w_{k|k}^{(j)} \mathbf{x}_k^{n,(j)} \quad (19a)$$

$$\hat{\mathbf{x}}_{k|k}^\ell = \sum_j w_{k|k}^{(j)} \hat{\mathbf{x}}_{k|k}^{\ell,(j)}, \quad (19b)$$

*Second order moments:*

$$\mathbf{P}_{k|k}^n = \sum_j w_{k|k}^{(j)} \Delta_{k|k}^{n,(j)} (\Delta_{k|k}^{n,(j)})^T \quad (20a)$$

$$\mathbf{P}_{k|k}^\ell = \sum_j w_{k|k}^{(j)} \left( \mathbf{P}_{k|k}^{\ell,(j)} + \Delta_{k|k}^{\ell,(j)} (\Delta_{k|k}^{\ell,(j)})^T \right) \quad (20b)$$

$$\mathbf{P}_{k|k}^{\ell n} = \mathbf{P}_{k|k}^{\ell n} = \sum_j w_{k|k}^{(j)} (\Delta_{k|k}^{\ell,(j)}) (\Delta_{k|k}^{n,(j)})^T, \quad (20c)$$

with:  $\Delta_{k|k}^{n,(j)} = \mathbf{x}_k^{n,(j)} - \hat{\mathbf{x}}_{k|k}^n$ ;  $\Delta_{k|k}^{\ell,(j)} = \hat{\mathbf{x}}_{k|k}^{\ell,(j)} - \hat{\mathbf{x}}_{k|k}^\ell$ .

**D. Prediction Step:** with the process model defined in (1).

*Non-linear part:*

$$w_{k+1|k}^{(m,j)} = w_{k|k}^{(j)} \mathcal{N}(\mathbf{x}_{k+1}^{n,(m)}; \bar{\mathbf{x}}_{k+1}^{n,(j)}, \mathbf{N}_k^{(j)}) \quad (21)$$

$$\text{with: } \bar{\mathbf{x}}_{k+1}^{n,(j)} = \mathbf{F}_k^n \mathbf{x}_k^{n,(j)} + \mathbf{F}_k^{n\ell} \hat{\mathbf{x}}_{k|k}^{\ell,(j)} \quad (22a)$$

$$\mathbf{N}_k^{(j)} = \mathbf{F}_k^n \mathbf{P}_k^{\ell,(j)} (\mathbf{F}_k^n)^T + \mathbf{Q}_w^n \quad (22b)$$

*Linear part:*

$$\hat{\mathbf{x}}_{k+1|k}^{\ell,(m,j)} = \mathbf{F}_k^{\ell n} \mathbf{x}_k^{n,(j)} + \mathbf{F}_k^\ell \hat{\mathbf{x}}_{k|k}^{\ell,(m,j)} \quad (23a)$$

$$\mathbf{P}_{k+1|k}^{\ell,(m,j)} = \mathbf{F}_k^\ell \mathbf{P}_k^{\ell,(m,j)} (\mathbf{F}_k^\ell)^T + \mathbf{Q}_w^\ell \quad (23b)$$

$$\text{with: } \hat{\mathbf{x}}_{k|k}^{\ell,(m,j)} = \hat{\mathbf{x}}_{k|k}^{\ell,(j)} + \mathbf{L}_k^{(j)} (\mathbf{x}_{k+1}^{n,(m)} - \bar{\mathbf{x}}_{k+1}^{n,(j)}) \quad (24a)$$

$$\mathbf{P}_{k|k}^{\ell,(m,j)} = \mathbf{P}_{k|k}^{\ell,(j)} - \mathbf{L}_k^{(j)} \mathbf{N}_k^{(j)} (\mathbf{L}_k^{(j)})^T \quad (24b)$$

$$\mathbf{L}_k^{(j)} = \mathbf{P}_{k|k}^{\ell,(j)} (\mathbf{F}_k^{n\ell})^T (\mathbf{N}_k^{(j)})^{-1} \quad (24c)$$

*Gaussian Mixture (GM) reduction:*

$$w_{k+1|k}^{(m)} = \sum_j w_{k+1|k}^{(m,j)} \quad (25a)$$

$$\hat{\mathbf{x}}_{k+1|k}^{\ell,(m)} = \sum_j \alpha_{k+1|k}^{(m,j)} \hat{\mathbf{x}}_{k+1|k}^{\ell,(m,j)} \quad (25b)$$

$$\mathbf{P}_{k+1|k}^{\ell,(m)} = \sum_j \alpha_{k+1|k}^{(m,j)} (\mathbf{P}_{k+1|k}^{\ell,(m,j)} + \bar{\mathbf{P}}_{k+1|k}^{\ell,(m,j)}) \quad (25c)$$

$$\text{with: } \alpha_{k+1|k}^{(m,j)} = \frac{w_{k+1|k}^{(m,j)}}{w_{k+1|k}^{(m)}} \quad (26a)$$

$$\bar{\mathbf{P}}_{k+1|k}^{\ell,(m,j)} = \Delta_{k+1|k}^{\ell,(m,j)} (\Delta_{k+1|k}^{\ell,(m,j)})^T \quad (26b)$$

$$\Delta_{k+1|k}^{\ell,(m,j)} = \hat{\mathbf{x}}_{k+1|k}^{\ell,(m,j)} - \hat{\mathbf{x}}_{k+1|k}^{\ell,(m)} \quad (26c)$$

### A. Initialization

This step initializes the predicted PDF  $p(\mathbf{x}_k | \mathbb{Z}_{k-1})$  based on an initially known Gaussian distribution  $p(\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0; \hat{\mathbf{x}}_0, \mathbf{P}_0)$ .  $\hat{\mathbf{x}}_0$  and  $\mathbf{P}_0$  can be expressed in terms of the linear and nonlinear parts as:

$$\hat{\mathbf{x}}_0 = \begin{bmatrix} \hat{\mathbf{x}}_0^n \\ \hat{\mathbf{x}}_0^\ell \end{bmatrix}; \quad \mathbf{P}_0 = \begin{bmatrix} \mathbf{P}_0^n & \mathbf{P}_0^{n\ell} \\ \mathbf{P}_0^{\ell n} & \mathbf{P}_0^\ell \end{bmatrix}. \quad (27)$$

Notice that  $\mathbf{P}_0^{n\ell} = (\mathbf{P}_0^{\ell n})^T$ , since they are the initial cross-covariance matrices between the linear and nonlinear parts. The expressions to initialize the predicted-PDF parameters at grid point- $j$ , i.e.,  $(w_{k|k-1}^{(j)}, \hat{\mathbf{x}}_{k|k-1}^{\ell,(j)}, \mathbf{P}_{k|k-1}^{\ell,(j)})$ , from the terms in (27) are given in (15)-(16) of Algorithm-1. Notice that  $w_{k|k-1}^{(j)}$  are associated to the marginal distribution of  $\mathbf{x}_0^n$ . Whereas,  $(\hat{\mathbf{x}}_{k|k-1}^{\ell,(j)}, \mathbf{P}_{k|k-1}^{\ell,(j)})$  are associated to the conditional distribution of  $\mathbf{x}_0^\ell$  given  $\mathbf{x}_0^n$ ; hence  $\mathbf{x}_0^\ell$  parameters depend on  $\mathbf{x}_0^n$  as shown in (16). The derivation of such expressions is detailed in [20]-Section-IV-B.

### B. Correction

This step computes the corrected-PDF based on both the predicted-PDF and the new available measurement. Therefore, the corrected-PDF parameters at grid point- $j$ , i.e.,  $(w_{k|k}^{(j)}, \hat{\mathbf{x}}_{k|k}^{\ell,(j)}, \mathbf{P}_{k|k}^{\ell,(j)})$ , are computed using (17) and (18) of Algorithm-1. Those expressions are slightly different from the ones derived in [20]-Section-IV-C. However, it is easy to prove that both are equivalent considering that the measurement model only depends on  $\mathbf{x}^n$ . Such a consideration is true for the considered system model described in Section II-A.

On the one hand, in Bayesian filtering, the covariance of the state-vector is increased at the prediction step, and reduced at the correction step. Such a reduction is a consequence of adding the new measurement information. On the other hand, as shown in (18), the linear part is not modified at the correction step. Consequently, one could incorrectly assume that the variance of the linear part increases indefinitely, which is not the case. As shown in (11), the nonlinear and linear parts are jointly distributed. Moreover,  $\hat{\mathbf{x}}_{k|k}^{\ell,(j)}$  and  $\mathbf{P}_{k|k}^{\ell,(j)}$  are parameters of the conditional distribution  $p(\mathbf{x}_{k|k}^\ell | \mathbf{x}_k^n)$  at grid point- $j$ . Therefore, the nonlinear part will incorporate measurement information to the linear part at the prediction step, thus reducing its covariance. This effect will be better explained in Section III-D and III-C.

### C. Moment Computation

As shown in Figure 1(b), the final RBPMF estimates of the state-vector are computed as the first two order moments of the corrected PDF. As detailed in [20]-Section IV-C-3, the first and second order moments can be computed as shown in (19) and (20). Notice that each of those expressions require a sum over all grid points.

### D. Prediction

This step computes the predicted-PDF based on the corrected-PDF and the process model shown in (1). Such an operation requires to work with two grids of points<sup>4</sup>, i.e., a first grid at time instant  $k$  with associated index  $j$  and a second grid at time instant  $k+1$  with associated index  $m$ . The predicted-PDF is defined on the second grid; thus, the final expressions for its parameters at grid point- $m$ , i.e.,  $(w_{k+1|k}^{(m)}, \hat{\mathbf{x}}_{k+1|k}^{\ell,(m)}, \mathbf{P}_{k+1|k}^{\ell,(m)})$  are given in (25). As shown in Algorithm-1, the prediction step is composed of three sub-steps as:

1) *Prediction of nonlinear part:* Intuitively, only  $\mathbf{x}_k^n$  is considered as state vector; whereas  $\mathbf{x}_k^\ell$  is taken as an additional source of noise [16]. As a result, for a grid point- $j$ , the process model (1) associated to the non-linear part is rearranged as:

$$\mathbf{x}_{k+1}^{n,(j)} = \mathbf{F}_k^n \mathbf{x}_k^{n,(j)} + \bar{\mathbf{w}}_k^n, \quad (28)$$

where  $\bar{\mathbf{w}}_k^n$  is an artificial noise with mean  $\mathbf{F}_k^n \hat{\mathbf{x}}_k^{\ell,(j)}$  and covariance matrix  $\mathbf{N}_k^{(j)}$  defined in (22b).

<sup>4</sup>Notice that the number of new grid points  $\mathbf{x}_{k+1}^{n,(m)}$  could be different from the grid points  $\mathbf{x}_k^{n,(j)}$ , and located at different locations [16]. Nevertheless in this paper we consider fixed locations and constant number of grid points.

Following the procedure in [20]-Section-IV-D, the nonlinear marginal PDF  $p(\mathbf{x}_{k+1}^n | \mathbb{Z}_k)$  can be computed using such a modified process model, resulting in:

$$p(\mathbf{x}_{k+1}^n | \mathbb{Z}_k) = \sum_{m=1}^{N_s} \sum_{j=1}^{N_s} w_{k+1|k}^{(m,j)} \delta(\mathbf{x}_{k+1}^n - \mathbf{x}_{k+1}^{n,(m)}), \quad (29)$$

where  $w_{k+1|k}^{(m,j)}$  is defined in (21). We would like to highlight three important characteristics of such a distribution.

- The internal summation means that the PDF is in fact a mixture distribution. As can be easily seen in (21), the PDF has one Gaussian component per grid point- $j$ .
- Due to the sampling operation, the computation of the weight at grid point- $m$  will require to evaluate  $N_s$  Gaussian mixture (GM) components. Such computation is repeated  $N_s$  times, one per grid point- $m$ . Hence, the complexity of the prediction step is quadratic on the number of grid points. This drawback is known in the literature as weight convolution that has huge impact over the computational complexity [16].
- The number of GM components grows after each iteration, which is solved using moment matching as explained in Section III-D3.

2) *Prediction of linear part:* Similarly to the previous step, only  $\mathbf{x}_k^\ell$  is considered as state vector; whereas  $\mathbf{x}_{k+1}^n$  and  $\mathbf{x}_k^n$  are considered fixed values. Therefore, the process model (1) is rearranged to update  $\mathbf{x}_k^\ell$  at grid point- $j$  as:

$$\mathbf{x}_{k+1}^{\ell,(j)} = \mathbf{F}_k^\ell \mathbf{x}_k^{\ell,(j)} + \bar{\mathbf{w}}_k^\ell, \quad (30a)$$

$$\mathbf{y}_k^{(m,j)} = \mathbf{F}_k^{n\ell} \mathbf{x}_k^{\ell,(j)} + \mathbf{w}_k^n, \quad (30b)$$

where  $\bar{\mathbf{w}}_k^\ell$  is an artificial noise with mean  $\mathbf{F}_k^{n\ell} \mathbf{x}_k^{n,(j)}$  and covariance matrix  $\mathbf{Q}_w^\ell$ . Additionally,  $\mathbf{y}_k^{(m,j)} = \mathbf{x}_{k+1}^{n,(m)} - \mathbf{F}_k^{n\ell} \mathbf{x}_k^{n,(j)}$  is referred to as virtual measurement. With such a system<sup>5</sup>, the final linear estimates are computed as in (23) of Algorithm-1, which were derived in [20]-Section-IV-D. It is important to notice that (30) has a linear structure, based on which we highlight the following key points:

- Intuitively, (23) is found in two stages. First, a correction step is done using the virtual measurement  $\mathbf{y}_k^{(m,j)}$ . This can be seen clearly in (24), where  $\mathbf{L}_k^{(j)}$  plays the role of Kalman gain. Second, the corrected  $(\hat{\mathbf{x}}_{k|k}^{\ell,(m,j)}, \mathbf{P}_{k|k}^{\ell,(m,j)})$  are propagated in time using the process model (30a). Such an operation can be seen clearly in (23).
- The virtual measurement  $\mathbf{y}_k^{(m,j)}$  depends on the future and current grids. Therefore, for each new grid point- $m$  the operations described by (23) and (24) are performed  $N_s$  times, i.e., one operation per grid point- $j$ . Thus, similar to the weight convolution problem, it also has an impact over the computational complexity.
- It is at this step where the nonlinear part incorporates virtual measurement information to the linear part. Moreover, it can be seen in (24b) how the matrix covariance of the linear part is reduced by the nonlinear part.

<sup>5</sup>Such a system silently assumes that  $\mathbf{w}^n$  and  $\mathbf{w}^\ell$  are not correlated. When the system does not satisfy such a condition, the Gram-Schmidt procedure can be used to fulfill such a requirement [19].

3) *Gaussian Mixture components reduction:* The resulting distribution is GM with  $N_s$  components represented by  $(w_{k+1|k}^{(m,j)}, \hat{\mathbf{x}}_{k+1|k}^{\ell,(m,j)}, \mathbf{P}_{k+1|k}^{\ell,(m,j)})$ . Therefore, Moment Matching is used to reduce the number of GM components as shown in (25). Notice that the virtual measurements used in the previous step were not linked directly to the real measurement  $\mathbf{z}_k$ . However, such information is carried in the associated weights  $w_{k+1|k}^{(m,j)}$ , as defined in (21). Consequently, the weights are used to select the appropriate virtual-measurement set that represents the true measurement information, as it is seen in (25).

#### IV. REDUCED COMPLEXITY - RBPMF

A heuristic approach to reduce the complexity of the conventional RBPMF is described in this section. On the one hand, as discussed in Section-III-D, the conventional RBPMF is computationally very expensive. The most costly operation is the weight convolution which takes place at the prediction step [16], [19], [20]. On the other hand, we restrict the posterior-PDF to be Gaussian distributed. Such a condition will be justified in Sections IV-A and IV-D. Therefore, we propose to replace the RBPMF prediction step by the LKF prediction step.

Such replacement also helps reducing the complexity at the correction step, as detailed in Section IV-B. The resulting tracking algorithm is presented in Algorithm-2 and shown in Fig. 1(c). Similar to the conventional RBPMF, the resulting approach is also composed of four functional steps. However, the interaction between them is different. Each functional block is explained as follows.

##### A. Prediction

In general, the prediction step makes use of the Chapman-Kolmogorov equation, given in (9). This equation has a closed form solution whenever two requirements are met: 1. The process model is linear, and 2. the posterior PDF is Gaussian distributed [27].

We fulfill the first requirement by considering a low-level maneuvering target, i.e., whose motion is constrained to a reduced set of trajectories. A clear example of such a target is a car moving in an urban environment. Cars maneuvers are reduced, since they are physically constrained to follow the road, roundabouts, etc. [28]. Consequently, it is reasonable to consider that a linear process model (1) captures the target motion well enough.

The second requirement is met by ensuring the corrected PDF to be Gaussian distributed, which we do as follows:

- At  $k=0$ , the corrected PDF is initialized with a Gaussian distribution [20], i.e.,  $p(\mathbf{x}_0 | \mathbb{Z}_0) = \mathcal{N}(\mathbf{x}_0; \hat{\mathbf{x}}_0, \mathbf{P}_0)$ . This initialization is different from the one of the conventional RBPMF, in which rather the grid parameters of the predicted PDF are initialized.
- Since we track a single target, it is reasonable to think that the filter will concentrate the distribution around a single location over time. Therefore at  $k > 0$ , we approximate the

corrected PDF as a Gaussian distribution using moment matching, i.e.,

$$p(\mathbf{x}_k|\mathbb{Z}_k) \approx \mathcal{N}(\mathbf{x}_k|k; \hat{\mathbf{x}}_k|k, \mathbf{P}_k|k), \quad (38)$$

where  $\hat{\mathbf{x}}_k|k$  and  $\mathbf{P}_k|k$  are computed in Section IV-D. We are aware that this approximation reduces the filter use to specific scenarios. However, it leads to a considerable reduction in complexity as seen in Section-V-C2. We discuss the impact of this approximation in the same Section-V-C1.

With both requirements being met, we propose to replace completely the prediction step detailed in Section III-D, by the closed-form solution given by the LKF prediction step. As a result, the predicted PDF is defined as:

$$p(\mathbf{x}_k|\mathbb{Z}_{k-1}) = \mathcal{N}(\mathbf{x}_k|k-1; \hat{\mathbf{x}}_k|k-1, \mathbf{P}_k|k-1), \quad (39)$$

where  $(\hat{\mathbf{x}}_k|k-1, \mathbf{P}_k|k-1)$  are defined in (31). Such parameters are computed based on  $(\hat{\mathbf{x}}_{k-1}|k-1, \mathbf{P}_{k-1}|k-1)$ , which are the parameters of the approximated corrected-PDF at time instant  $k-1$ . Notice that on the one hand, the conventional RBPMF-prediction step has a quadratic complexity on the

number of grid points. On the other hand, the LKF-prediction step does not depend on the grid points (See Table-I). Therefore, by replacing the RBPMF-prediction step by the one of the LKF, we avoid such a quadratic complexity.

### B. Grid Re-initialization

Up to this point, the predicted PDF is given by  $(\hat{\mathbf{x}}_k|k-1, \mathbf{P}_k|k-1)$  computed in the previous step. However, in order to use RBPMF in the correction step, we need to express such PDF into its RB factorization form, i.e., we need  $(w_{k|k-1}^{(j)}, \hat{\mathbf{x}}_k^{\ell,(j)}|k-1, \mathbf{P}_k^{\ell,(j)}|k-1)$  in terms of  $(\hat{\mathbf{x}}_k|k-1, \mathbf{P}_k|k-1)$ . Therefore, we reinitialize the grid at every iteration using the method described in Section III-A applied to the predicted PDF. Hence, the predicted mean vector and covariance matrix must be divided into nonlinear and linear parts as

$$\hat{\mathbf{x}}_k|k-1 = \begin{bmatrix} \hat{\mathbf{x}}_k^n|k-1 \\ \hat{\mathbf{x}}_k^\ell|k-1 \end{bmatrix}; \mathbf{P}_k|k-1 = \begin{bmatrix} \mathbf{P}_k^n|k-1 & \mathbf{P}_k^{\ell n}|k-1 \\ \mathbf{P}_k^{\ell n}|k-1 & \mathbf{P}_k^\ell|k-1 \end{bmatrix}. \quad (40)$$

The set of parameters  $(w_{k|k-1}^{(j)}, \hat{\mathbf{x}}_k^{\ell,(j)}|k-1, \mathbf{P}_k^{\ell,(j)}|k-1)$  are initialized at every iteration as:

$$w_{k|k-1}^{(j)} = \mathcal{N}(\mathbf{x}_k^{n,(j)}; \hat{\mathbf{x}}_k^n|k-1, \mathbf{P}_k^n|k-1) \quad (41a)$$

$$\hat{\mathbf{x}}_k^{\ell,(j)}|k-1 = \hat{\mathbf{x}}_k^\ell|k-1 - \mathbf{K}_{k|k-1}(\mathbf{x}_k^{n,(j)} - \hat{\mathbf{x}}_k^n|k-1) \quad (41b)$$

$$\mathbf{P}_k^{\ell,(j)}|k-1 = \mathbf{P}_k^\ell|k-1 - \mathbf{K}_{k|k-1}\mathbf{P}_k^{\ell n}|k-1, \quad (41c)$$

with  $\mathbf{K}_{k|k-1} = \mathbf{P}_k^{\ell n}|k-1(\mathbf{P}_k^n|k-1)^{-1}$ . Finally, notice that (41b) and (41c) will no longer be needed in the final algorithm, due to the complexity reduction explained in Section IV-D.

### C. Correction

There are no changes applied to this step compared to the conventional RBPMF. That is, only the nonlinear part is directly updated from the measurement, as can be shown in (33).

### D. Moment Computation

One could use the same Moment-computation-step described in Section III-C using (19) and (20). However, it is possible to simplify the computation of the linear moments. In summary, the final linear moments shown in (36) imply that the complexity reduction of this step is twofold. First, we can compute the linear moments easily based on the current and previous nonlinear moments. Second, we can avoid to compute  $(\hat{\mathbf{x}}_k^{\ell,(j)}, \mathbf{P}_k^{\ell,(j)}|k-1)$ , since they are no longer used by the final algorithm. We reduce the computation of the linear moments as follows.

1) *Mean vector*  $\hat{\mathbf{x}}_k^\ell|k$ : Replacing (18) and (41b) into (19b) yields:

$$\hat{\mathbf{x}}_k^\ell|k = \sum_{j=1}^{N_s} w_{k|k}^{(j)} (\hat{\mathbf{x}}_k^\ell|k-1 - \mathbf{K}_{k|k-1}(\mathbf{x}_k^{n,(j)} - \hat{\mathbf{x}}_k^n|k-1)). \quad (42)$$

Such an expression can be finally reduced to (36a) in three stages. First, distribute the summation to all the terms. Second, replace the nonlinear covariance matrix using (19a); and third, notice that  $\sum_j w_{k|k}^{(j)}=1$  as shown in (17). Notice that the final expression (36a) does not involve the use of grid points.

---

## Algorithm 2 :Reduced Complexity RBPMF for Direct Tracking

---

**Initialization:**  $p(\mathbf{x}_{k-1}|\mathbb{Z}_{k-1})=\mathcal{N}(\mathbf{x}_0; \hat{\mathbf{x}}_0, \mathbf{P}_0)$ .

**A. Prediction:** Compute  $p(\mathbf{x}_k|\mathbb{Z}_{k-1})$  using:

$$\hat{\mathbf{x}}_k|k-1 = \mathbf{F}_k \hat{\mathbf{x}}_{k-1}|k-1 \quad (31a)$$

$$\mathbf{P}_k|k-1 = \mathbf{F}_k \mathbf{P}_{k-1}|k-1 \mathbf{F}_k^T + \mathbf{Q}_w \quad (31b)$$

**B. Grid Re-initialization:** Expressing  $(\hat{\mathbf{x}}_k|k-1, \mathbf{P}_k|k-1)$  as in (40), compute the weights:

$$w_{k|k-1}^{(j)} = \mathcal{N}(\mathbf{x}_k^{n,(j)}; \hat{\mathbf{x}}_k^n|k-1, \mathbf{P}_k^n|k-1) \quad (32)$$

**C. Correction:** nonlinear part only

$$w_{k|k}^{(j)} = \frac{w_{k|k-1}^{(j)} \xi_k^{(j)}}{\sum_i w_{k|k-1}^{(i)} \xi_k^{(i)}}; \quad \xi_k^{(i)} = p(\mathbf{z}_k|\mathbf{x}_k^n = \mathbf{x}_k^{n,(i)}) \quad (33)$$

**D. Moment Computation:**

*Nonlinear moments:*

$$\hat{\mathbf{x}}_k^n|k = \sum_j w_{k|k}^{(j)} \mathbf{x}_k^{n,(j)} \quad (34)$$

$$\mathbf{P}_k^n|k = \sum_j w_{k|k}^{(j)} \Delta_{k|k}^{n,(j)} (\Delta_{k|k}^{n,(j)})^T \quad (35)$$

*Linear moments:*

$$\hat{\mathbf{x}}_k^\ell|k = \hat{\mathbf{x}}_k^\ell|k-1 - \mathbf{K}_{k|k-1}(\hat{\mathbf{x}}_k^n|k - \hat{\mathbf{x}}_k^n|k-1) \quad (36a)$$

$$\mathbf{P}_k^\ell|k = \mathbf{P}_k^\ell|k-1 - \mathbf{K}_{k|k-1}(\mathbf{P}_k^{\ell n}|k-1 - \mathbf{P}_k^n|k \mathbf{K}_{k|k-1}^T) \quad (36b)$$

$$\mathbf{P}_k^{\ell n}|k = \mathbf{K}_{k|k-1} \mathbf{P}_k^n|k \quad (36c)$$

$$\text{with: } \mathbf{K}_{k|k-1} = \mathbf{P}_k^{\ell n}|k-1 (\mathbf{P}_k^n|k-1)^{-1}. \quad (37)$$


---

2) *Covariance matrix*  $\mathbf{P}_{k|k}^\ell$ : We start by computing the difference  $\Delta \hat{\mathbf{x}}_{k|k}^{\ell,(j)} = \hat{\mathbf{x}}_{k|k}^{\ell,(j)} - \hat{\mathbf{x}}_{k|k}^\ell$ ; thus subtracting (36a) from (41b) results in:

$$\Delta \hat{\mathbf{x}}_{k|k}^{\ell,(j)} = \mathbf{K}_{k|k-1} \Delta \hat{\mathbf{x}}_{k|k}^{\text{n},(j)}, \quad (43)$$

where  $\Delta \hat{\mathbf{x}}_{k|k}^{\text{n},(j)} = \hat{\mathbf{x}}_{k|k}^{\text{n},(j)} - \hat{\mathbf{x}}_{k|k}^{\text{n}}$ . Using (43), the linear covariance matrix, defined in (20b), is rewritten as:

$$\mathbf{P}_{k|k}^\ell = \sum_j^{N_s} w_{k|k}^{(j)} \left( \mathbf{P}_{k|k}^{\ell,(j)} + \mathbf{K}_{k|k-1} \Delta \hat{\mathbf{x}}_{k|k}^{\text{n},(j)} \Delta \hat{\mathbf{x}}_{k|k}^{\text{n},(j)T} \mathbf{K}_{k|k-1}^T \right). \quad (44)$$

Finally, such an expression can be reduced to (36b) in three stages. First, distribute the summation. Second, replace the nonlinear covariance matrix using (20a). Third, notice that  $\mathbf{P}_{k|k}^{\ell,(j)}$  does not depend on  $j$ , as shown in (41c).

3) *Crosscovariance matrix*  $\mathbf{P}_{k|k}^{\ell n}$ : Using (43), (20c) is rewritten as:

$$\mathbf{P}_{k|k}^{\ell n} = \mathbf{K}_{k|k-1} \sum_j^{N_s} w_{k|k}^{(j)} \Delta \hat{\mathbf{x}}_{k|k}^{\text{n},(j)} \Delta \hat{\mathbf{x}}_{k|k}^{\text{n},(j)T}. \quad (45)$$

The final expression given in (36c) is easily obtained by replacing the nonlinear covariance matrix using (20a). Finally, we use moment matching to represent the corrected PDF as Gaussian distribution, i.e.,

$$p(\mathbf{x}_k | \mathbb{Z}_k) \approx \mathcal{N} \left( \mathbf{x}_k; \begin{bmatrix} \hat{\mathbf{x}}_{k|k}^{\text{n}} \\ \hat{\hat{\mathbf{x}}}_{k|k}^{\text{n}} \end{bmatrix}, \begin{bmatrix} \mathbf{P}_{k|k}^{\text{n}} & \mathbf{P}_{k|k}^{\text{n}\ell} \\ \mathbf{P}_{k|k}^{\ell n} & \mathbf{P}_{k|k}^\ell \end{bmatrix} \right). \quad (46)$$

Such distribution is passed to the prediction-step of the next RC-RBPMF iteration.

## V. PERFORMANCE AND COMPLEXITY ANALYSIS

### A. Considered Scenario and performance metric

We consider a scene consisting of  $N=4$  time-synchronized BSs. Such BSs are located on the corners of a 50m-sided square that encloses a roundabout. We assume a target, such as an automobile, equipped with a radio transmitter which is also time-synchronized with the BSs. The true target trajectory reassembles a car making a left turn following the roundabout. A graphical representation of the target trajectory, as well as of the scenario is presented in Fig. 5.

For convenience, we use a CV model, since the target trajectory is constrained by the shape of the roundabout. Hence,  $\mathbf{x}_k^{\text{n}}$  and  $\dot{\mathbf{x}}_k^{\ell}$  are respectively, the position and velocities in Cartesian coordinates. Moreover, the matrices of the process model are given in Appendix-A. The target is initially located at  $x_0=37$  m and  $y_0=3$  m with initial velocities  $\dot{x}=0$  m/s and  $\dot{y}=15$  m/s. The total target velocity at any point of the trajectory is constant, i.e.,  $\|\dot{\mathbf{x}}_k^{\ell}\|_2=15$  m/s. We choose  $\Delta v_{\text{max}}=2$  m/s as maximum change in velocity expected during one time interval  $T=100$  ms. Consequently, we choose  $\sigma_w=\Delta v_{\text{max}}/T$ , i.e.,  $\sigma_w=20$  m/s<sup>2</sup>.

Regarding the target-to-infrastructure communication, OFDM modulation is used over a bandwidth of 40 MHz. Each OFDM symbol contains  $P=64$  equispaced pilots with  $\Delta f=625$  kHz. It is assumed that the BSs communicate

the received pilot symbols to a localization center where the processing takes place.

Similarly to [12], the SNR is defined as  $\text{SNR}=\frac{1}{P\sigma_w^2} \sum_p |s_p|^2$ . For convenience, we fix a  $\text{SNR}_{\text{ref}}$  value at a particular reference distance  $d_{\text{ref}}=20$  m. In addition, we assume that the higher or lower SNR at the BSs is only due to the increment or reduction of signal power due to free-space propagation. Hence, similar to [29] the SNR values at each BS- $i$  can be computed as:

$$\text{SNR}_{k,i} = \text{SNR}_{\text{ref}} \left( \frac{d_{\text{ref}}}{d_i(\mathbf{x}_k^{\text{n}})} \right)^\alpha, \quad (47)$$

where  $\alpha=2$  is the path loss exponent. Notice that the resulting SNR is different at each BS. Moreover, the closer the target is to BS- $i$  the higher the  $\text{SNR}_{k,i}$ .

In this paper, the localization performance is considered as the first performance metric. Similar to [9], it is defined as the localization-Root Mean Squared Error (RMSE) for the whole trajectory as:

$$\text{RMSE} = \sqrt{\frac{1}{N_r N_k} \sum_{r=1}^{N_r} \sum_{k=1}^{N_k} \|\mathbf{x}_k^{\text{n}} - \hat{\mathbf{x}}_{k,r}^{\text{n}}\|_2^2}, \quad (48)$$

where  $N_r$  and  $N_k$  are, respectively, the number of realizations and filtering iterations. The vectors  $\mathbf{x}_k^{\text{n}}$  and  $\hat{\mathbf{x}}_{k,r}^{\text{n}}$  are the true and estimated transmitter positions respectively for realization  $r$  at time instant  $k$ .

The second performance metric is the computational complexity. Similar to [30], we choose to take the number of Floating point operation (Flop)s required for a single filtering iteration. Notice that a Flop is defined as one addition, subtraction, multiplication or division of two floating-point numbers. This metric is not dependent on SNR; hence, it is analyzed in terms of number grid points.

### B. Direct vs Indirect Tracking

To better show the advantage of direct tracking compared to indirect ones, we analyze the localization performance of sampled approaches. As a reminder, direct-tracking considers the received signals as input to the filter. That is, they consider the measurement model detailed in (4). Conversely, indirect-tracking take some IPs, which are estimated from the received signals. In this paper, the considered IPs are the ToA-estimates. Consequently, the mean ToA  $\tau$  and its associated variance  $\sigma_\tau^2$  are estimated from the received signals as explained in [12]. Since the IPs are used as input for tracking, the filter employs a different measurement model. Similar to [8], such a model can be defined for BSs- $i$  as:

$$z_{k,i} = \tau_{k,i} = \frac{1}{c} d_i(\mathbf{x}_k^{\text{n}}) + v_i, \quad (49)$$

where  $v_i$  is a zero-mean Gaussian additive noise with variance  $\sigma_\tau^2$ .

We start by comparing the sampled approaches, i.e., PF and RBPMF. Fig. 2 shows the localization-RMSE for both approaches applied to direct and indirect tracking. The IPs-estimation is reliable for high SNR-values ( $\text{SNR} > -3$  dB), and thus both direct and indirect tracking approaches have

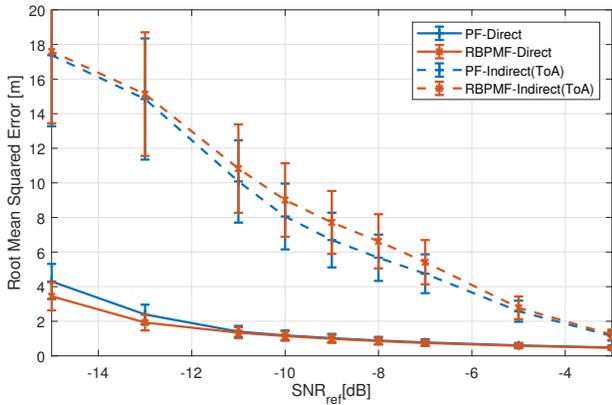


Fig. 2. Localization error for the whole trajectory for Direct and Indirect sampled methods. The bars express 95-percent confidence intervals.

similar performance. As the SNR decreases, the IPs estimation becomes unreliable. As a result, Direct PF and RBPMF have lower localization error compared to their indirect counterparts. Notice that both PF and RBPMF show similar behaviors for the whole range of SNR-values. Due to the reduction of the sampling space, the same number of samples yield a better PDF-approximation when using RB-factorization; however, the difference in localization error is small in this case. Lastly, even though the localization errors are very similar for both PF and RBPMF, there is still a huge computational complexity gap between them which is the topic addressed in the next section.

### C. Reduced Complexity-RBPMF

Before assessing the performance of the proposed RC-RBPMF, we discuss the validity of the approximation done in Section-IV-A. That is, we analyze, qualitatively, if the matched Gaussian distribution is a good approximation or not for the posterior distribution. To measure the difference between a given distribution and its matched Gaussian distribution, we define the mean Kullback–Leibler (KL) divergence as

$$D_{KL}(\hat{p}||p) = \frac{1}{N_r N_k} \sum_{r=1}^{N_r} \sum_{k=1}^{N_k} \left( \sum_S \hat{p}_{r,k} \log \left( \frac{\hat{p}_{r,k}}{p_{r,k}} \right) \right), \quad (50)$$

where  $S$  is the support of the distributions, and  $p_{r,k}$  and  $\hat{p}_{r,k}$  are respectively the given distribution and its moment-matched Gaussian approximation at time instant  $k$  and realization  $r$ . Since the KL divergence is not bounded, we normalize it as

$$\bar{D}_{KL} = 1 - e^{-|D_{KL}|}. \quad (51)$$

As a result,  $\bar{D}_{KL}$  is zero when there is no difference at all, and one for a huge difference between the considered distributions.

The approximation of the posterior-PDF takes place at the output of the correction step (10). Therefore in Fig 3, we show the  $\bar{D}_{KL}$  values for the posterior  $p(\mathbf{x}_k|\mathbb{Z}_k)$  and measurement likelihood  $p(\mathbf{z}_k|\mathbf{x}_k)$ . The measurement likelihood is more different from its matched normal distribution, as the SNR decreases. Such behavior agrees with the fact that the

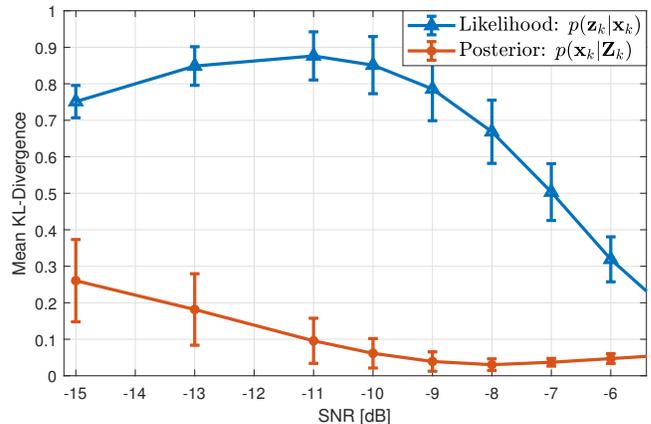


Fig. 3. KL-divergence for the Posterior and Likelihood PDFs compared to their moment-matched Gaussian distribution, respectively.

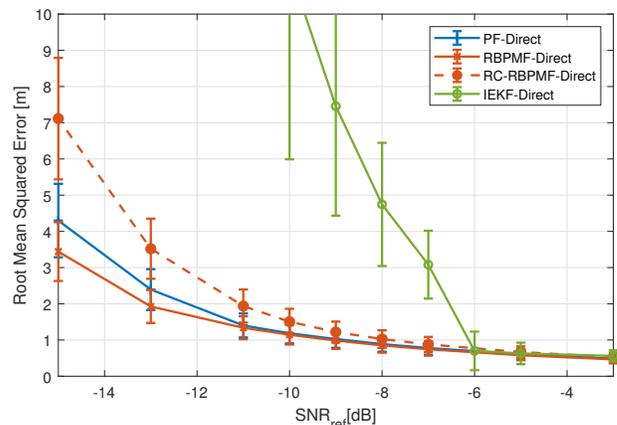


Fig. 4. Localization error for the whole trajectory for the considered Direct-tracking methods. The bars express the 95-percent confidence intervals.

measurement likelihood presents multiple local maximas at low SNR-values [10]-Section-IV.

The approximation error for the posterior-PDF,  $p(\mathbf{x}_k|\mathbb{Z}_k)$ , is low at high SNR values. That is, the posterior-PDF is well represented by the matched-distribution (46). Such behavior supports our choice of replacing the posterior distribution by a moment-matched Gaussian distribution as stated in Section IV-A. Notice that mean KL-divergence of the approximated distribution also increases as the SNR decreases. However, it increases at a lower degree compared to the measurement likelihood.

We now proceed to compare the RC-RBPMF, proposed in Section-IV, to the conventional RBPMF. We discuss first the localization performance, followed by the reduction in computational complexity.

#### 1) Localization Performance:

As it will be seen in the next sub-Section, the reduction of complexity is substantial. Nonetheless, it comes with a cost. As shown in Fig. 4, we compare the RC-RBPMF localization-error to the other direct-approaches.

At high-SNR values, the RC and all other considered direct-tracking approaches present very similar localization performance. Intuitively, this means that the approximation

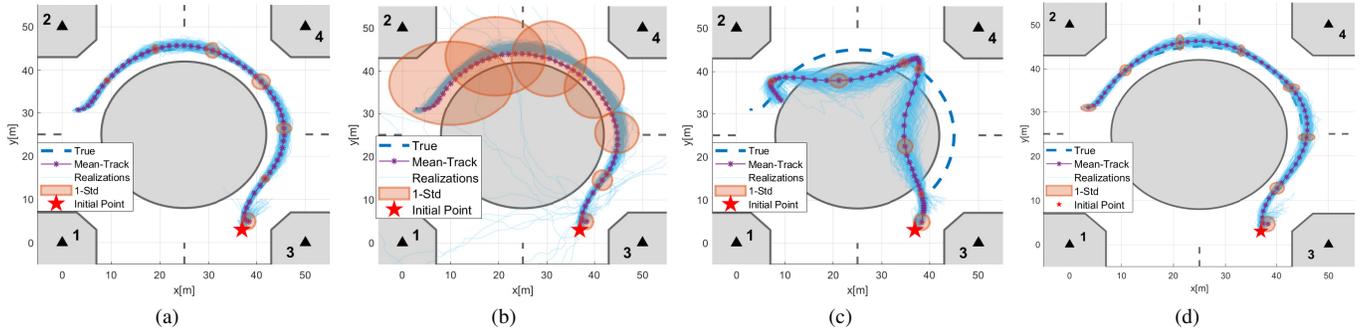


Fig. 5. Track realizations at  $\text{SNR}_{ref} = -10$  dB for: a) Direct RBPMF b) Direct IEKF c) Indirect (ToA) RBPMF d) Direct PF. The ellipses represent the 1-Standard deviation at the specific time across the realizations

to the posterior-PDF does not introduce significant errors. In addition, this is also supported by the results shown in Fig. 3, which shows a low mean KL-divergence for the posterior distribution. Conversely, the localization-RMSE of the proposed RC-RBPMF increases as the SNR decreases. Moreover, the localization-RMSE is slightly larger than the ones for PF and conventional-RBPMF at low SNR. Such larger error implies that the matched-Gaussian does not represent the posterior-PDF completely well, as it can be seen in Fig. 3. Although, the resulting direct RC-RBPMF has larger localization-RMSE than the conventional RBPMF, it presents a significant reduction in complexity as it will be detailed in the Section-V-C2.

Aside from PF and RBPMF, we show in Fig-4 the direct-IEKF proposed in [9]. It is interesting to see that for  $\text{SNR} < -6$ dB, IEKF has a very high localization error. The reason for that is twofold. First, direct-IEKF solves the correction step by maximizing, iteratively, the linearized version of the received-signal log-likelihood. Second, as explained in [10]-Section-IV, the received-signal log-likelihood presents several local maxima and minima, which becomes more problematic as the SNR decreases. Therefore, the iterative maximization used by the direct-IEKF might choose a local maxima different from the right solution. This wrong decision leads the filter to diverge, and thus to high localization errors. To have a better understanding of such phenomenon, we present all track realizations in Fig. 5 for  $\text{SNR}_{ref} = -10$  dB. We show the mean trajectory as well as the 1-Standard Deviation (SD) ellipses every 800 ms. Such mean trajectory and SD ellipses are computed across the realizations at the same time instant  $k$ . Fig. 5b shows clearly how the direct-IEKF diverges from the right trajectory at unpredictable times. Conversely, Fig. 5a and Fig. 5d shows, respectively, the track realizations for the RBPMF and PF, which do not present such an issue. Moreover, as interpreted from Fig.-2, PF and RBPMF have the similar performance at  $\text{SNR}_{ref} = -10$  dB.

## 2) Computational Complexity:

For the sake of clarity, we provide the final complexity expressions for each step in Table-I and we provide the expressions grouped by multiplications and additions in Appendix B. Such expressions were derived following the procedure detailed in [30].

Notice that in Table-I,  $p$  and  $n$  are, respectively, the

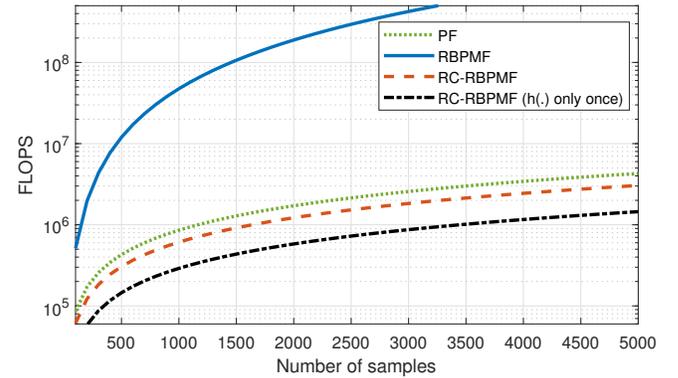


Fig. 6. Computational Complexity as a function of number of samples.

TABLE I  
COMPLEXITY PER STEP

STEP	COMPLEXITY
RBPMF-Grid Init.	$(4n^3 + 3n^2 + 2n + k_e)N_s + n^3$
RBPMF Correction	$(3 + k_m + k_a + k_e + h_m + h_a)N_s - 1$
RBPMF Moment Comp.	$(9n^2 + 6n)N_s - (3n^2 + 2n)$
RBPMF Prediction	$a_1 N_s^2 + a_2 N_s + a_3$ with: $a_1 = 8n^2 + 3n + 1$ $a_2 = 19n^3 - n^2 - 2n + 2 + k_e$ $a_3 = 12n^3 - 3n^2 - 1$
PF Prediction	$(4p^3 + p^2 - p + k_r)N_s$
PF Correction	$(2p + 4 + k_z + h_m + h_a)N_s$ with: $k_z = k_m + k_a + k_e + k_r$
LKF Prediction	$(16n^2 + 2n - 1)2n$

dimension of  $\mathbf{x}_k$  and  $\mathbf{x}_k^n$ ,  $N_s$  is the number of samples,  $k_m$  and  $k_a$  are the flops related to multiplication and addition, required to evaluate the log-likelihood defined in (7) for one grid point. Notice that  $k_m$  and  $k_a$  do not consider the evaluation of  $h_i(\cdot)$ , and thus we define  $h_m$  and  $h_a$  as the flops needed to evaluate  $h_i(\cdot)$  for one grid point. Lastly,  $k_e > 0$  and  $k_r > 0$  are, respectively, the average values of effective flops in the exponentiation and sampling operations.

Figure 6 shows the total complexity as a function of the number of samples  $N_s$ . It is considered that  $k_m = 2P$  and  $k_a = 2P - 1$  are required as described in [13]; as well as  $h_m = 5P$  and  $h_a = P$  for the oversimplified measurement

model. Two main remarks can be done based on Figure 6. First, the complexity gain between the conventional RBPMF and RC-RBPMF is noticeable. Second, the RC-RBPMF is slightly less complex than PF, due to the fact that RC-RBPMF does not make the prediction per sample, as the PF does.

It should be mentioned that this complexity reduction strategy can be also applied to the PF. However, the reduction in complexity would not be as noticeable as in the RBPMF case, since the PF prediction step is of much less complexity than the one of the conventional RBPMF. Lastly, the complexity reduction strategy can be pushed further by considering a fixed grid of samples (as the one considered in this paper). In such a scenario, some operations can be removed from the iterative process and computed only once. For instance, the measurement function  $\mathbf{h}_i(\mathbf{x}_k)$  could be evaluated in advance, since it requires the value of the sample  $\mathbf{x}_k^{(j)}$  that is fixed though out the iterations. As a result, the RC-RBPMF reaches even lower computational complexity as shown in Fig. 6.

## VI. CONCLUSION

In this work, we have addressed the tracking problem of a wireless transmitter, considering a system model composed of a linear process model and a highly nonlinear measurement model. First, we used a low complexity version of PMF, such as RBPMF. Second, we reduced the complexity even further by replacing the prediction-step of RBPMF by the one of LKF. As a result, the proposed approach is able to track the target for a broader range of SNR values as well as having a reduced complexity compared to the common sampling approaches. To illustrate the benefits of our approach, we have compared the proposed tracking algorithm to two other representative direct-tracking approaches, the direct-PF and direct-IEKF. Further studies will follow, for example, in extending the proposed algorithm to a distributed scenario.

### APPENDIX A

#### LINEAR AND CURVILINEAR MOTION MODELS

##### A. CV model

Since this model is trivial, we directly provide the final vectors and matrices as:

$$\mathbf{x}_k^n = [x_k, y_k]^T \quad ; \quad \mathbf{x}_k^\ell = [\dot{x}_k, \dot{y}_k]^T \quad (52a)$$

$$\mathbf{Q}_w^n = \begin{bmatrix} \mathbf{Q}_w^n & \mathbf{Q}_w^{n\ell} \\ \mathbf{Q}_w^{\ell n} & \mathbf{Q}_w^{\ell\ell} \end{bmatrix} \quad ; \quad \mathbf{F}_k^n = \mathbf{I}_2 \quad ; \quad \mathbf{F}_k^{n\ell} = T\mathbf{I}_2 \quad (52b)$$

$$\mathbf{F}_k^{\ell n} = \mathbf{0}_2 \quad ; \quad \mathbf{F}_k^{\ell\ell} = \mathbf{I}_2$$

where  $(x_k, y_k)$  and  $(\dot{x}_k, \dot{y}_k)$  are, respectively, the target Cartesian position and velocities.  $\mathbf{Q}_w^n = \frac{\sigma_w^2 T^4}{4} \mathbf{I}_2$ ,  $\mathbf{Q}_w^\ell = \sigma_w^2 T^2 \mathbf{I}_2$  and  $\mathbf{Q}_w^{n\ell} = \mathbf{Q}_w^{\ell n} = \frac{\sigma_w^2 T^3}{2} \mathbf{I}_2$ .  $T$  is the sampling interval, and  $\sigma_w^2$  is computed based on the amount of expected velocity error [26].

##### B. CTRV model

This model is defined as [6], [31]:

$$\begin{bmatrix} x \\ y \\ v \\ \theta \\ \omega \end{bmatrix}_{k+1} = \begin{bmatrix} f_x \\ f_y \\ v \\ \theta + \omega T \\ \omega \end{bmatrix}_k + \mathbf{w}_k, \quad (53a)$$

$$\text{with: } f_x = x + \frac{2v}{\omega} \sin\left(\frac{\omega T}{2}\right) \cos\left(\theta + \frac{\omega T}{2}\right) \quad (53b)$$

$$f_y = y + \frac{2v}{\omega} \sin\left(\frac{\omega T}{2}\right) \sin\left(\theta + \frac{\omega T}{2}\right) \quad (53c)$$

where  $v$  is the magnitude of velocity and  $(\theta, \omega)$  are respectively the yaw angle and rate. Such a model is first linearized and then rearranged into the form (1) as:

$$\mathbf{x}_k^n = [x_k, y_k]^T \quad ; \quad \mathbf{x}_k^\ell = [v_k, \theta_k, \omega_k]^T \quad (54)$$

$$\mathbf{F}_k^n = \mathbf{I}_2 \quad ; \quad \mathbf{F}_k^{\ell n} = \mathbf{0}_{3 \times 2} \quad (55)$$

$$\mathbf{F}_k^{n\ell} = \begin{bmatrix} \frac{\partial f_x}{\partial v} & \frac{\partial f_x}{\partial \theta} & \frac{\partial f_x}{\partial \omega} \\ \frac{\partial f_y}{\partial v} & \frac{\partial f_y}{\partial \theta} & \frac{\partial f_y}{\partial \omega} \end{bmatrix} \quad ; \quad \mathbf{F}_k^\ell = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \quad (56)$$

In addition, the process covariance matrix is given by:  $\mathbf{Q}_w^n = \mathbf{0}_2$ ,  $\mathbf{Q}_w^{n\ell} = \mathbf{0}_{2 \times 3}$ ,  $\mathbf{Q}_w^{\ell n} = (\mathbf{Q}_w^{\ell\ell})^T$ , and:

$$\mathbf{Q}_w^{\ell\ell} = \text{diag}\{T^2 \sigma_v^2, T^2 \sigma_\omega^2 \left[ \begin{array}{cc} T/3 & 1/2 \\ 1/2 & 1 \end{array} \right]\}, \quad (57)$$

where  $\sigma_v^2$  and  $\sigma_\omega^2$  are maximum expected changes in  $v$  and  $\omega$ , respectively [31].

### APPENDIX B

#### COMPLEXITY OF ALGORITHMS

The number of Flops due to multiplications per step is given in Table II. Similarly, the number of Flops due to additions is presented in Table III. We derived such expressions following the procedure proposed in [30]. Therefore, we kindly refer the reader to the original paper.

TABLE II  
MULTIPLICATION-RELATED COMPLEXITY PER STEP

Step	Multiplication
Grid Init	$(2n^2 + 1)(n + 1)N_s$
Correction	$(k_m + h_m + 2)N_s$
Moment Comp.	$2n(3n + 1)N_s$
Prediction	$(2n^2)N_s^2 + (9n^3 + 3n^2 + n + 2)N_s + 5n^3$
GM reduction	$(2n^2 + n + 1)N_s^2$
LKF-Pred.	$4n^2(4n + 1)$

TABLE III  
ADDITION-RELATED COMPLEXITY PER STEP

Step	Additions
Grid Init.	$(2n - 1)(n^2 + n + 1)N_s$
Correction	$(1 + k_a + h_a)N_s - 1$
Moment Comp.	$(3n^2 + 4n)N_s - (3n^2 + 2n)$
Prediction	$2n^2 N_s^2 + (9n^3 - 3n^2 - 2n)N_s + 5n^3 - 3n^2 - 1$
GM reduction	$2(n^2 + n)N_s^2 - n(n + 1) * N_s$
LKF-Pred.	$2n(8n^2 - 1)$

## ACKNOWLEDGMENT

The authors acknowledge the financial support of the Walloon Region through the WIN2Wal/2018/1/DI/34 LUMINET project.

## REFERENCES

- [1] J. A. del Peral-Rosado, R. Raulafs, J. A. López-Salcedo, and G. Seco-Granados, "Survey of cellular mobile radio localization methods: From 1g to 5g," *IEEE Communications Surveys&Tutorials*, vol. 20, no. 2, pp. 1124–1148, 2018.
- [2] S. Sivasakthiselvan and V. Nagarajan, "Localization techniques of wireless sensor networks: A review," in *2020 International Conference on Communication and Signal Processing (ICCSPP)*, 2020, pp. 1643–1648.
- [3] "Evolved universal terrestrial radio access (e-utra); physical channels and modulation (release 13)," in *Technical Specification 36.211 V13.13.0*, January 2020.
- [4] A. Hakkarainen, J. Werner, M. Costa, K. Leppanen, and M. Valkama, "High-efficiency device localization in 5g ultra-dense networks: Prospects and enabling technologies," in *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*, 2015, pp. 1–5.
- [5] S. Kuutti, S. Fallah, K. Katsaros, M. Dianati, F. McCullough, and A. Mouzakitis, "A survey of the state-of-the-art localization techniques and their potentials for autonomous vehicle applications," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 829–846, 2018.
- [6] R. Schubert, E. Richter, and G. Wanielik, "Comparison and evaluation of advanced motion models for vehicle tracking," *2008 11th International Conference on Information Fusion*, pp. 1–6, 2008.
- [7] A. Amar and A. J. Weiss, "New asymptotic results on two fundamental approaches to mobile terminal location," in *2008 3rd International Symposium on Communications, Control and Signal Processing*, 2008, pp. 1320–1323.
- [8] L. Storrer, H. C. Yildirim, M. Crauwels, E. I. P. Copa, S. Pollin, J. Louveaux, P. De Doncker, and F. Horlin, "Indoor tracking of multiple individuals with an 802.11ax wi-fi-based multi-antenna passive radar," *IEEE Sensors Journal*, vol. 21, no. 18, pp. 20 462–20 474, 2021.
- [9] M. Hehn, E. Sippel, and M. Vossiek, "An iterative extended kalman filter for coherent measurements of incoherent network nodes in positioning systems," *IEEE Access*, vol. 8, pp. 36 714–36 727, 2020.
- [10] E. I. Pocoma Copa, F. Rottenberg, F. Quitin, L. Vandendorpe, P. De Doncker, and F. Horlin, "Iterative toa-based localization of wireless transmitters using dirichlet-kernel-based range representation," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, 2021, pp. 1–5.
- [11] A. Y. Sidi and A. J. Weiss, "Delay and doppler induced direct tracking by particle filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 1, pp. 559–572, 2014.
- [12] F. Horlin, M. Van Eeckhaute, T. Van der Vorst, A. Bourdoux, F. Quitin, and P. De Doncker, "Iterative toa-based terminal positioning in emerging cellular systems," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–5.
- [13] M. Van Eeckhaute, T. Van der Vorst, A. Bourdoux, F. Quitin, P. De Doncker, and F. Horlin, "Low complexity iterative localization of time-misaligned terminals in cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 10 730–10 739, 2018.
- [14] P. Vu, A. M. Haimovich, and B. Himed, "Direct tracking of multiple targets in mimo radar," in *2016 50th Asilomar Conference on Signals, Systems and Computers*, 2016, pp. 1139–1143.
- [15] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerospace and Electronic Systems Magazine*, vol. 25, no. 7, pp. 53–82, 2010.
- [16] C.-K. Sung and S. J. Lee, "Reliable time propagation algorithms for pmf and rbpmf," *Sensors*, vol. 21, no. 1, 2021. [Online]. Available: <https://www.mdpi.com/1424-8220/21/1/261>
- [17] J. Elfring, E. Torta, and R. van de Molengraft, "Particle filters: A hands-on tutorial," *Sensors*, vol. 21, no. 2, 2021. [Online]. Available: <https://www.mdpi.com/1424-8220/21/2/438>
- [18] N. Bergman and L. Ljung, "Point-mass filter and cramer-rao bound for terrain-aided navigation," in *Proceedings of the 36th IEEE Conference on Decision and Control*, vol. 1, 1997, pp. 565–570 vol.1.
- [19] T. Schon, F. Gustafsson, and P.-J. Nordlund, "Marginalized particle filters for mixed linear/nonlinear state-space models," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2279–2289, 2005.
- [20] J. Duník, M. Soták, M. Veselý, O. Straka, and W. Hawkinson, "Design of rao-blackwellized point-mass filter with application in terrain aided navigation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 55, no. 1, pp. 251–272, 2019.
- [21] G. Hendeby, R. Karlsson, and F. Gustafsson, "A new formulation of the rao-blackwellized particle filter," in *2007 IEEE/SP 14th Workshop on Statistical Signal Processing*, 2007, pp. 84–88.
- [22] E. I. Pocoma Copa, F. Quitin, L. Vandendorpe, P. D. Doncker, and F. Horlin, "Self-synchronization based distributed localization of wireless transmitters," in *2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2021, pp. 1222–1227.
- [23] T. Schon, F. Gustafsson, and P.-J. Nordlund, "Marginalized particle filters for mixed linear/nonlinear state-space models," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2279–2289, 2005.
- [24] E. I. Pocoma Copa, F. Quitin, L. Vandendorpe, D. D. Philippe, and F. Horlin, "Self-synchronization based localization of a time-misaligned transmitter in cellular networks," *IEEE Transactions on Vehicular Technology*, pp. 1–1, 2022.
- [25] M. Kolat, O. Törő, and T. Bécsi, "Performance evaluation of a maneuver classification algorithm using different motion models in a multi-model framework," *Sensors*, vol. 22, no. 1, 2022. [Online]. Available: <https://www.mdpi.com/1424-8220/22/1/347>
- [26] Y. J. Liu Chongyi, Lu Cheng, *Estimation for Kinematic Models*. John Wiley & Sons, Ltd, 2017, ch. 6, pp. 267–299. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/0471221279.ch6>
- [27] H. Fang, N. Tian, Y. Wang, M. Zhou, and M. A. Haile, "Nonlinear bayesian estimation: from kalman filtering to a broader horizon," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 2, pp. 401–417, 2018.
- [28] L. Hartjen, R. Philipp, F. Schuldt, F. Howar, and B. Friedrich, "Classification of driving maneuvers in urban traffic for parametrization of test scenarios," in *9th Tagung Automatisiertes Fahren*, 2019, pp. 1–7. [Online]. Available: <https://shorturl.at/xORU4>
- [29] R. Kaune, J. Hörst, and W. Koch, "Accuracy analysis for toa localization in sensor networks," in *14th International Conference on Information Fusion*, 2011, pp. 1–8.
- [30] R. Karlsson, T. Schon, and F. Gustafsson, "Complexity analysis of the marginalized particle filter," *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4408–4411, 2005.
- [31] X. Rong Li and V. Jilkov, "Survey of maneuvering target tracking. part i. dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.



**Evert I. Pocoma Copa** was born in 1990 in El Alto, Bolivia. He received a scholarship from the Foundation Simón I. Patiño that helped him to receive the M.Sc. degree in electrical engineering from the Université Libre de Bruxelles (ULB), Brussels, Belgium, in 2019. Since September 2019, he is working toward the Ph.D degree with OPERA WCG, ULB. He is currently working on distributed localization and tracking techniques in wireless sensor networks.



**Laurent Storrer** was born in Brussels in 1996, Laurent Storrer is currently a Ph.D. researcher jointly at Université Libre de Bruxelles in the OPERA - Wireless Communications Group and at Katholieke Universiteit Leuven in the ESAT-TELEMIC department, with a scholarship from F.N.R.S.-E.O.S. He received his MSc in electrical engineering from ULB & VUB in 2019. He is working on multi-antenna Wi-Fi-based passive radars, target tracking and classification for crowd monitoring applications.



**François Quitin** received the Ph.D. degree in electrical engineering from the Université Libre de Bruxelles (ULB), Brussels, Belgium, and the Université Catholique de Louvain, Louvain-La-Neuve, Belgium, in 2011. He is an Assistant Professor with ULB. From 2011 to 2013, he was a Postdoctoral Researcher with the University of California, Santa Barbara, and from 2013 to 2015, he was a Postdoctoral Research Fellow with Nanyang Technological University, Singapore. His research interests include

experimental and prototyping aspects in wireless communications, taking advanced theoretical ideas all the way to practice. He was the recipient of the Alcatel-Lucent Bell Scientific Award in 2012.



**François Horlin** received the Ph.D. degree from the Université Catholique de Louvain, Louvain-La-Neuve, Belgium, in 2002. He specialized in the field of signal processing for digital communications. He joined the Interuniversity Micro-Electronics Center in 2002. He led the project aiming at developing a 4G cellular communication system in collaboration with Samsung Korea. In 2007, he became a Professor with the Université Libre de Bruxelles (ULB), Brussels, Belgium. He is currently supervising

a research team working on next-generation communication systems. Localization based on 5G signals, filterbank-based modulations, massive multiple input and multiple output, and dynamic spectrum access are examples of currently investigated research topics. He has been an Academic Representative to the Executive Board of ULB from 2010 to 2015. Since 2017, he has been the Vice Dean for research with the Ecole Polytechnique de Bruxelles.



**Luc Vandendorpe** (Fellow, IEEE) was born in Mouscron, Belgium, in 1962. He received the degree (summa cum laude) in electrical engineering and the Ph.D. degree in applied science from the Catholic University of Louvain (UCLouvain), Louvain La Neuve, Belgium, in 1985 and 1991, respectively. Since 1985, he has been with the Communications and Remote Sensing Laboratory, UCL, where he first worked in the field of bit rate reduction techniques for video coding. In 1992, he was a Visiting Scientist and a Research Fellow

with the Telecommunications and Traffic Control Systems Group, Delft Technical University, The Netherlands, where he worked on spread spectrum techniques for personal communications systems. From October 1992 to August 1997, he was a Senior Research Associate with the Belgian NSF, UCL, and an invited Assistant Professor. He is currently a Full Professor with the Institute for Information and Communication Technologies, Electronics, and Applied Mathematics, UCLouvain. His research interests include digital communication systems and more precisely resource allocation for OFDM(A)-based multicell systems, MIMO and distributed MIMO, sensor networks, UWB-based positioning, and wireless power transfer. He was the Chair of the IEEE Benelux Joint Chapter on Communications and Vehicular Technology from 1999 to 2003. He was a Co-Technical Chair for IEEE ICASSP 2006. He served as an Editor for Synchronization and Equalization of IEEE Transactions on Communications from 2000 to 2002, and as an Associate Editor for IEEE Transactions on Wireless Communications from 2003 to 2005, and IEEE Transactions on Signal Processing from 2004 to 2006. He is or has been a TPC Member for numerous IEEE conferences (VTC, GLOBECOM, SPAWC, ICC, PIMRC, and WCNC). He was an Elected Member of the Signal Processing for Communications Committee from 2000 to 2005, and the Sensor Array and Multichannel Signal Processing Committee of the Signal Processing Society from 2006 to 2008 and from 2009 to 2011.



**Philippe De Doncker** received the M.Sc. degree in physics engineering and the Ph.D. degree in science engineering from the Université Libre de Bruxelles (ULB), Brussels, Belgium, in 1996 and 2001, respectively. He is currently a Professor with the ULB, where he leads the research activities on wireless channel modeling and (bio) electromagnetics.