Tensor Decomposition based Bearing-Only Target Tracking - an Analysis based on Real Data

Joshua Gehlen, Martin Ulmke, Jannik Springer, Felix Govaers, Wolfgang Koch
Sensor Data and Information Fusion (SDF)
Fraunhofer FKIE
53343 Wachtberg, Germany
{joshua.gehlen, martin.ulmke, jannik.springer, felix.govaers, wolfgang.koch}@fkie.fraunhofer.de

***Abstract*—This paper presents the application of a novel target tracking technique employing tensor decompositions for discretizing the target state space. The time evolution of the conditional probability density is realized by a Fokker-Planck equation solver and the measurement update, as usual, by applying Bayes' rule. The method is applicable to non-Gaussian and non-linear system equations and enables the treatment of complex non-Gaussian target state densities. In addition, the efficient tensor decomposition scheme, in principle, allows for high-dimensional target states. The new tracking filter is applied to the problem of tracking an agile air target using bearing measurements from distributed acoustic and electromagnetic array sensors based on real data. It is shown that the new filter is able to initiate and maintain the target track with localization errors comparable to those of a standard particle filter.**

***Index Terms*—target tracking, non-linear filtering, tensor decomposition, particle filter, real data**

I. Introduction

Target tracking is a fundamental problem in many applications such as surveillance, robotics, and autonomous vehicles. Sensors are used to detect and estimate the states of various targets by processing their measurements. While, for simplicity, the sensor model is often assumed to be linear with Gaussian noise, in many real-world scenarios, the sensor model is non-Gaussian and highly non-linear. Even if the sensor model is assumed linear and Gaussian in measurement coordinates, the corresponding likelihood function, in general, is non-Gaussian in target state coordinates due to non-linear measurement functions. Typical examples are sensors that provide bearing measurements such as cameras, radio frequency (RF) receivers, or microphone arrays. Common approaches to deal with such non-linearities employ approximations such as linearization or unscented transform leading to generalizations of the Kalman filter, e.g., the Extended and the Unscented Kalman filter. A different class of solutions does not approximate the physical models or the state densities but the numerical solution to them by discretizing either the density distributions or the target state space. They have the advantages not to enforce Gaussian prior or posterior densities and to systematically improve their accuracy by enhancing the number of samples or discretization cells. The discretization of the density distribution by samples of point masses leads to sequential Monte Carlo, or particle filter methods. A new approach that employs the state space discretization based on tensor decompositions is the topic of the present work. It is based on the references [1]–[3]. The methodology is explained in Sec. II.

The new tracking filter was already applied on simulated data for algorithm development for example in [4], [5]. In this work it is applied to real data from a measurement campaign at FKIE in which a multicopter drone is detected by an RF direction finding (DF) sensor and two microphone arrays, as detailed in Sec. III. The results of the new tracking filter and the comparison to a bootstrap particle filter are presented and discussed in Sec. IV. Sec. V summarizes the work and identifies future research questions.

II. Methodology

In the following section we will present the fundamentals of the tensor based target tracking approach which is based on the Bayesian tracking cycle (see, e.g., [6], Chapter 3).

A. Bayesian Tracking

Bayesian tracking follows the **Iterative Bayesian tracking formalism** consisting of a target state prediction and a filtering step. The last posterior at time step $k-1$, $p(\mathbf{x}_{k-1}|Z^{k-1})$, for a target with state vector $\mathbf{x}$ is predicted with the help of a physics based evolution model to $p(\mathbf{x}_k|Z^{k-1})$ using the **Chapman-Kolmogorov equation**:

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

We can also express the continuous time prediction in the form of the **Fokker-Planck equation**

$$\frac{\partial p(\mathbf{x},t)}{\partial t} = -\sum_{i=1}^{N} \frac{\partial}{\partial \mathbf{x}_i} f(\mathbf{x},t) p(\mathbf{x},t) + \sum_{i,j=1}^{N} \frac{\partial^2}{\partial \mathbf{x}_i \partial \mathbf{x}_j} [\mathbf{D}_{ij}(\mathbf{x},t) p(\mathbf{x},t)] \quad (2)$$

with f as a drift function and $\mathbf{D}$ the diffusion matrix. These can be derived from the stochastic evolution process of the target

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x})\mathrm{d}t + d\mathbf{w} \quad (3)$$

with process noise vector $\mathbf{w}$. Measurements Z_k are taken at discrete time steps t_k $(k = 1, 2, \dots)$ and consist of a set of

$m_k \geq 0$ detections, i.e. $Z_k = \{\mathbf{z}_k^{(1)}, \ldots, \mathbf{z}_k^{(m_k)}\}$, which are either false alarms or a true target detection. The sequence of measurements for all time steps up to k is denoted as $Z^k = (Z_1, \ldots, Z_k)$. The measurement Z_k is processed for $\mathbf{x}_k \equiv \mathbf{x}(t_k)$ in the filtering step using the **Bayes theorem**

$$p(\mathbf{x}_k|Z^k) = \frac{p(Z_k|\mathbf{x}_k)p(\mathbf{x}_k|Z^{k-1})}{\int d\mathbf{x}_k p(Z_k|\mathbf{x}_k)p(\mathbf{x}_k|Z^{k-1})} \tag{4}$$

where $p(Z_k|\mathbf{x}_k)$ is the measurement likelihood function. Assuming a homogeneous false alarm distribution and at most one true target detection (cmp. chapter 2.3 in [6]) the likelihood function can be written as a sum over mutual exclusive detection and non-detection events:

$$p(Z_k|\mathbf{x}_k) = P_D \sum_{j=1}^{m_k} p(\mathbf{z}_k^{(j)}|\mathbf{x}_k)P_f(m_k - 1) + (1 - P_D)P_f(m_k) \tag{5}$$

where $P_f(n)$ is the probability for having n false alarms, P_D the detection probability, and $p(\mathbf{z}_k^j|\mathbf{x}_k)$ the single-detection likelihood function derived from the sensor specific statistical measurement equation:

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \tag{6}$$

with measurement noise $\mathbf{v}_k$. Both, process and measurement noise vectors, can be non-Gaussian.

B. Tensor based Target Tracking

The Tensor based Target Tracking (TbTT) discretizes the state space. Subsequently, it is possible to represent linear as well as non-linear densities. For discretization, the FoV is divided into equal sized cells for which the probability density function is evaluated. If the dimension $i \in \{1, \ldots d\}$ is devided into n_i cells, we can index

$$[p(\mathbf{x})]_{i_1,\ldots,i_d} = p(x_{i_1,\ldots,i_n}) \tag{7}$$

with i_j as the j-th index of the dimension i. The number of cells, therefore, is $n_1 * \cdots * n_d$. For high dimensional densities this discretization is infeasible with regard to the memory size. To overcome this challenge, we use the CANDECOMP/PARAFAC (CP) decomposition. A d-dimensional tensor $\mathcal{T} \in \mathbb{R}^{(n_1,\ldots,n_d)}$ can be decomposed into a sum of rank-1 tensors.

$$\mathcal{T} = \sum_{r=1}^{R} \mathbf{a}_{n_1,r} \circ \mathbf{a}_{n_2,r} \circ \ldots \circ \mathbf{a}_{n_d,r} \tag{8}$$

The rank-1 tensors are computed by the outer product of the *loading vectors* $\mathbf{a}_{n_i,r} \in \mathbb{R}^{n_i}$. If it consists of R components, it has rank R [7]. When stacking together the loading vectors of each dimension to a matrix (also called factor matrix) of $\mathbf{A}_i \in \mathbb{R}^{n_i \times R}$, we write $\mathcal{T} = [\![\mathbf{A}_1, \ldots, \mathbf{A}_n]\!]$. The latter can be calculated by the Alternating Least Squares (ALS) algorithm [8].

1) ALS: The ALS is an algorithm for matrix factorization similar to a Singular Value Decomposition which can be generalized for tensors. In an iterative manner the ALS minimizes the squared Frobenius Norm of the error. First, we consider a matrix $\mathbf{X}$ which is decomposed into component matrices $\mathbf{Y}, \mathbf{Z}$ by minimizing the error function

$$\begin{aligned} D_F(\mathbf{X}||\mathbf{Y}\,\mathbf{Z}) &= \frac{1}{2}\left\|\mathbf{X} - \mathbf{Y}\mathbf{Z}\right\|_F^2 \\ &= \frac{1}{2}\mathrm{tr}\left((\mathbf{X} - \mathbf{Y}\mathbf{Z})^T(\mathbf{X} - \mathbf{Y}\mathbf{Z})\right) \end{aligned} \tag{9}$$

subject to $\mathbf{Y}, \mathbf{Z} \geq 0$.

Subsequently, the error function can be minimized by alternatingly updating the matrices $\mathbf{Y}$ and $\mathbf{Z}$ while keeping the other matrix fixed. Mathematically, this can be achieved by calculating the least squares minimization. This yields to Algorithm 1.

Algorithm 1 ALS for matrix factorization

1: Initialize $\mathbf{Y}^{(0)}$ and $\mathbf{Z}^{(0)}$
2: **for** $k = 1, 2, \ldots$ **do**
3: $\quad \mathbf{Y}^{(k)} \leftarrow \mathbf{X}\mathbf{Z}^{(k-1)}(\mathbf{Z}^{(k-1)^T}\mathbf{Z}^{(k-1)})^{-1}$
4: $\quad \mathbf{Z}^{(k)} \leftarrow (\mathbf{Y}^{(k)^T}\mathbf{Y}^{(k)})^{-1}\mathbf{Y}^{(k)^T}\mathbf{X}$
5: **end for**

The latter can be generalized for the rank R decomposition of a tensor $\mathcal{T} \approx [\![\mathbf{U}_1, \ldots, \mathbf{U}_n]\!]$ by fixing all but one dimension. The non-fixed matrix (also called factor matrix) is then updated by the equation

$$\mathbf{U}_m = \mathcal{A}_{(m)} \left(\bigodot_{l \neq m}^{D} \mathbf{U}_l \right) \left(\circledast_{l \neq m}^{D} \mathbf{U}_l^T \mathbf{U}_l \right)^{\dagger} \tag{10}$$

where $\mathcal{A}_{(m)}$ describes the mode-m matricization [1]. $\circledast$ is the pointwise Hadamard product and $\odot$ the Khatri-Rao product. For given matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times m}$ it is defined as the column wise Kronecker product by

$$\mathbf{A} \odot \mathbf{B} = \begin{bmatrix} \mathbf{A}_1 \otimes \mathbf{B}_1 & \ldots & \mathbf{A}_m \otimes \mathbf{B}_m \end{bmatrix}. \tag{11}$$

For escaping the algorithmic loop one has to calculate the error of the tensor and its decomposition by means of the 2-norm. The ALS can also be applied in a variant form for the task of tensor deflation, in which the rank of an already decomposed tensor in CP format is reduced.

2) Prediction: For the prediction, we solve the Fokker-Planck equation (2) by computing the Focker-Planck operator and solving the equation by using the exponential function. To obtain a simple form, we assume that all densitites are given in a vectorized form. Apart from that, the drift and diffusion part of the differential equation is assumed to be independent in dimensions and can thus be represented as the sum of products of one-dimensional functions. From these functions the Fokker-Planck operator $\mathcal{L}$ can be derived (see [3]). As a result, the Fokker-Planck operator can be solved by calculating

$$p(\mathbf{x}_k|Z^{k-1}) = \exp(\mathcal{L}\Delta t)p(\mathbf{x}_{k-1}|Z^{k-1}) \tag{12}$$

Because of the high dimensionality, it is not possible to compute the exponential function. Thus, it is approximated by its Taylor series. Calculating its CPD decomposition leads to a strong increase of the tensor rank so that before continuing to the next step it has to be decreased by applying a deflation method.

3) Filtering: For the filtering the Bayes Theorem is used (see (4), (5)). The likelihood function has to be discretized in the CPD form $[\![\mathbf{L}_1,\dots,\mathbf{L}_D]\!]$ in the state space. Because of considering false measurements and detection probability $P_D < 1$, the likelihood consists of multiple parts. Each one's discretization is computed and then the CPD-forms are summed up. The latter is multiplied with the prior by the Hadamard product. In the CP-decomposition this Hadamard product can be easily calculated using the transposed Khatri-Rao product

$$[\![\mathbf{U}_1,\dots,\mathbf{U}_D]\!] * [\![\mathbf{V}_1,\dots,\mathbf{V}_D]\!] = [\![\mathbf{U}_1 \odot^T \mathbf{V}_1,\dots,\mathbf{U}_D \odot^T \mathbf{V}_D]\!] \quad (13)$$

where $\mathbf{A} \odot^T \mathbf{B} := (\mathbf{A}^T \odot \mathbf{B}^T)^T$. As the second step, the product is normalized.

$$[\![\mathbf{U}_1,\dots,\mathbf{U}_n]\!]_{\mathbf{x}_k|\mathbf{x}_k} = \frac{[\![\mathbf{L}_1,\dots,\mathbf{L}_n]\!] * [\![\mathbf{U}_1,\dots,\mathbf{U}_n]\!]_{\mathbf{x}_k|\mathbf{x}_{k-1}}}{\int [\![\mathbf{L}_1,\dots,\mathbf{L}_n]\!] * [\![\mathbf{U}_1,\dots,\mathbf{U}_n]\!]_{\mathbf{x}_k|\mathbf{x}_{k-1}} d\mathbf{x}_{k|k-1}}. \quad (14)$$

For getting the mean and covariance as well as for the normalization of a density in the CP decomposition we can use the fact that the integral can be computed by

$$\int d\mathbf{x}\, [\![\mathbf{U}_1,\dots,\mathbf{U}_D]\!] = \int d\mathbf{x} \sum_{r=1}^{R} \bigotimes_{d=1}^{D} \mathbf{u}_d^r \quad (15)$$

$$= \sum_{r=1}^{R} \int d\mathbf{x} \bigotimes_{d=1}^{D} \mathbf{u}_d^r \quad (16)$$

$$= \sum_{r=1}^{R} \sum_{i_1,\dots,i_D} \prod_{d=1}^{D} [\mathbf{u}_d^r]_{i_d} \Delta_{x_d} \quad (17)$$

$$= \sum_{r=1}^{R} \prod_{d=1}^{D} \Delta_{x_d} \sum_{i_d=1}^{N_d} [\mathbf{u}_d^r]_{i_d} \quad (18)$$

with N_d as the discretization range of dimension d and $\Delta_{x_d} = [x_d]_1 - [x_d]_0$. The mean follows directly by

$$\hat{x}_d = \int x_d [\![\mathbf{U}_1,\dots,\mathbf{U}_D]\!] dx_d \quad (19)$$

$$= \sum_{r=1}^{R} \prod_{j\neq d} \left(\Delta_{x_j} \sum_{i_j=1}^{N_j} [\mathbf{u}_j^r]_{i_j} \right) \Delta_{x_d} \sum_{i_d=1}^{N_d} [\mathrm{diag}\left([[\mathbf{x}_d]]\right) \mathbf{u}_d^r]_{i_d} \quad (20)$$

where $[[\mathbf{x}_d]]$ is the discretization for the d-th dimension and respectively the covariance by

$$\mathbf{P}_{i,j} = \int \mathrm{d}\mathbf{x}\, (\mathbf{x}_i - \hat{\mathbf{x}}_i)\, (\mathbf{x}_j - \hat{\mathbf{x}}_j) \sum_{r=1}^{R} \bigotimes_{d=1}^{D} \mathbf{u}_d^r \quad (21)$$

After definition of $\mathbf{T}_{\hat{\mathbf{x}}_j} \mathbf{x}_j = [(\mathbf{x}_j - \hat{\mathbf{x}}_j)]_j$ we can define the covariance diagonal entries by

$$\mathbf{P}_{ii} = \sum_{r=1}^{R} \prod_{k\neq i} \{\Delta_{x_k} \sum_{i_k=1}^{N_k} [\mathbf{u}_k^r]_{i_k}\} \cdot \Delta_{x_i} \sum_{i_i=1}^{N_i} [\mathrm{diag}([([\mathbf{x}_i]_r - \hat{\mathbf{x}}_i)^2]_r) \mathbf{u}_i^r]_{i_i} \quad (22)$$

with cross covariance entries

$$\mathbf{P}_{ij} = \sum_{r=1}^{R} \prod_{k\neq i \wedge k\neq j} \{\Delta_{x_k} \sum_{i_k=1}^{N_k} [\mathbf{u}_k^r]_{i_k}\} \cdot \Delta_{x_i} \sum_{i_i=1}^{N_i} [\mathrm{diag}([\mathbf{T}_{\hat{\mathbf{x}}_i} \mathbf{x}_i]) \mathbf{u}_i^r]_{i_i} \cdot \Delta_{x_j} \sum_{i_j=1}^{N_j} [\mathrm{diag}([\mathbf{T}_{\hat{\mathbf{x}}_j} \mathbf{x}_j]) \mathbf{u}_j^r]_{i_j} \quad (23)$$

III. The Scenario

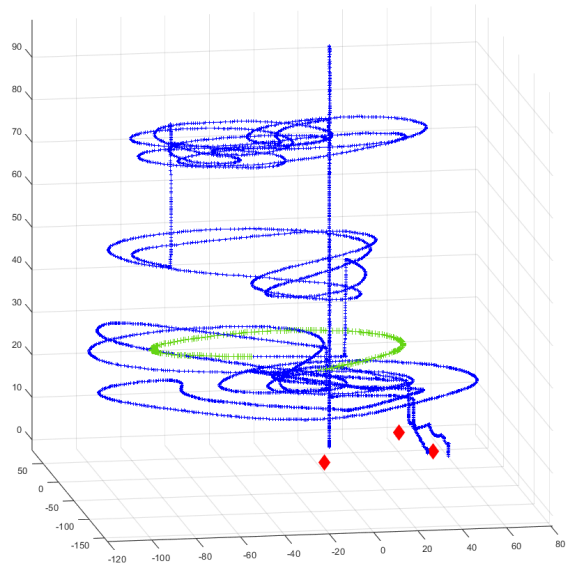

Fig. 1: The complete ground truth of the drone flight (blue), with the considered excerpt (green) and the sensor positions (red diamonds).

We apply the TbTT filter on data recorded during a measurement campaign at FKIE with three DF sensors for the detection and localization of small drones. A single drone was used as target and the full flight path is computed from its GPS positions and is shown in Fig. 1. Moreover, the positions of the DF sensors are marked with red diamonds and the excerpt of the flight considered in this work is marked in green. The sensors were equipped with an inertial navigation system (INS) that measures the position and orientation of the sensors

with respect to a world coordinate system. It is worth noting, that orientation errors from the INS can cause systematic DF errors. The DF sensors deployed for the measurements are array based and therefore consist of multiple spatially distributed sensor elements that are coherently processed. The acoustic array sensor can detect the rotor noises of the drone and estimate the direction-of-arrival (DOA) of the emitted sound waves. It employs a volumetric microphone array with 16 elements that are randomly distributed in a sphere (Fig. 2a), that allows an unambiguous estimation of all possible target DOAs. The RF array sensor is equipped with two dual-circularly polarized uniform circular antenna arrays (Fig. 2b) that are optimized for the 2.4 GHz and 5.8 GHz ISM-band, respectively [9]. During the experiment a drone was used that transmits a video downlink in the 2.4 GHz ISM-band to the remote control of the pilot. It is assumed that the drone is in the far-field of the array sensor such that the DOA of the emitted electromagnetic wave can be estimated by the array sensor. To reduce systematic DF errors of the RF sensor, a calibration of the antenna array was performed prior to the measurements [10].

It is assumed that the DF sensors independently measure the target DOA with azimuth angle α and elevation angle θ. Given the three-dimensional target position $\mathbf{p} = (x, y, z)^T$, the measurement equation can be formulated as follows

$$\begin{bmatrix} \alpha \\ \theta \end{bmatrix} = \begin{bmatrix} \text{atan2}\,(x, y) \\ \arcsin\left(z/\sqrt{x^2+y^2}\right) \end{bmatrix}. \tag{24}$$

IV. The Evaluation

As a benchmark, we compare the tracking results with those of a standard sequential importance resampling (SIR) particle filter, also known as *bootstrap* particle filter [11]. For initialization, particles are homogeneously distributed in a 3-D box including the field of view. Velocities are initiated by a Gaussian distribution with variance 1 m/s in each dimension. We observe that a number of 10,000 particles is sufficient both for track extraction and stable track update.

(a) Acoustic sensor (b) RF sensor

Fig. 2: The direction finding sensors deployed for the measurement campaign at FKIE.

For both filters, the nearly constant velocity model with a process noise strength of $q = 10$ is employed. An error of 3° is assumed in the bearing measurements. Besides, a detection probability of 0.95 and a false alarm density of $0.001 * 1/rad^2$ is assumend which corresponds to 2% of false alarms on average. For track deletion, a sequential likelihood ratio test is employed [12].

For an azimuth and elevation angle measurement, the likelihood can be seen in Fig 3. It consists of a cone looking in the direction where the target is supposed to be. Because of the transformation into the Cartesian space, the two dimensional measurement is transformed into a three dimensional positional target space. The target state space consists of the three dimensional positional space and the three dimensional velocity components. The TbTT filter handles false measurements with gating by excluding all measurements which density Mahalanobis distance is smaller than 90 percent. For this we approximate the well known gating using the chi-square distribution [6]

$$\mathbf{z}_k \in \mathcal{G}(\mathbf{x}_k) \Leftrightarrow (\mathbf{z}_k - \mathbf{h}(\mathbf{x}_k))^T \mathbf{S}^{-1} (\mathbf{z}_k - \mathbf{h}(\mathbf{x}_k)) \leq \chi^2_{\alpha,n} \tag{25}$$

by

$$\left(\log\left(\int [\![\mathbf{U}_1, \ldots, \mathbf{U}_n]\!] \circledast [\![\mathbf{L}_1, \ldots, \mathbf{L}_n]\!] d\mathbf{x}\right) + \log\left(|2 * \pi * 2 * \mathbf{R}|\right)/2\right) * (-2) \leq \chi^2_{\alpha,n} \tag{26}$$

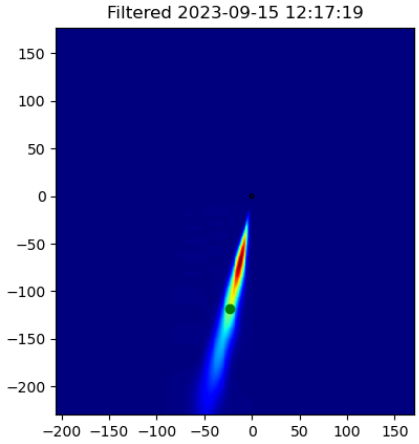

Fig. 4: The projection in x-y plane of a posterior density. The point depicts the GPS position of the target.

We consider a tensor rank of 30 and a discretization size of 1 meter for the positional axes and a size of $0.2\frac{m}{s}$ for the velocity axes.

For the evaluation we consider 424 measurements. In this time interval the drone flies nearly a circle. The resulting trajectories are shown in Fig. 6 and respectively the positional error in Fig. 7. The TbTT filter is able to initiate and maintain the track with localization errors comparable to those of the

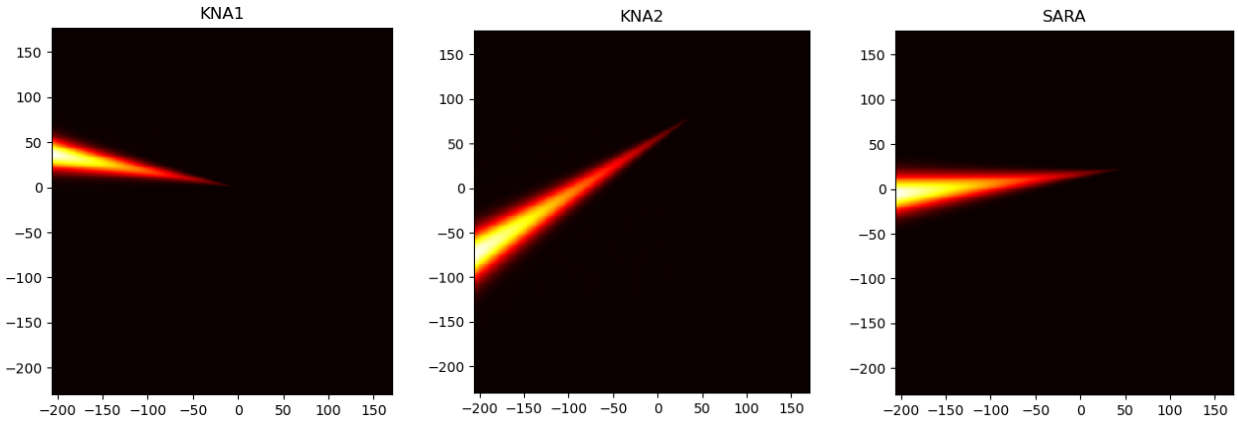

Fig. 3: The likelihood of the bearing measurements (x-y projection) for the two acoustic (KNA1, KNA2), and the RF (SARA) sensor arrays.

particle filter. It can be observed that the part of the trajectory in which the sensors have colinearities both filters have larger errors. Besides, it can be seen that at one point the TbTT looses the track for a few steps and has to be reinitialized. When considering the root mean squared error for the particle filter it is $8.40\,\mathrm{m}$ and for the TbTT filter $8.82\,\mathrm{m}$. A typical track density can be seen in Fig. 4.

Besides, it can be observed that for the initialization the TbTT initializes faster than the particle filter. In Fig. 5 the velocity in x, y and z dimension is considered. It can be seen that the velocity components of the TbTT is more stable than the particle filter.

V. Conclusion

In this paper, for a proof of concept, we apply the novel TbTT filter to real bearing measurements from a drone flight experiment. The results are compared with those of a bootstrap particle filter. Both filters can track the target with similar accuracy, even though the likelihoods are highly non-linear. As an outlook, we would like to investigate the TbTT on more complex scenarios, in particular with multiple targets, while speeding up its computation time. In addition, it would be interesting to consider hybrid filters which uses the TbTT tracking for the initialization. After the convergence towards a Gaussian density, a particle filter, EKF or UKF could then overtake the tracking.

References

[1] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[2] Y. Sun and M. Kumar, "Nonlinear bayesian filtering based on fokker-planck equation and tensor decomposition," in *2015 18th International Conference on Information Fusion (Fusion)*, 2015, pp. 1483–1488.

[3] B. Demissie, M. A. Khan, and F. Govaers, "Nonlinear filter design using Fokker-Planck propagator in Kronecker tensor format," *FUSION 2016 - 19th International Conference on Information Fusion, Proceedings*, vol. 1, pp. 1–8, 2016.

[4] J. Gehlen, F. Govaers, and W. Koch, "On Tracking Closely-Spaced Targets in a PARAFAC-Representation of the Fermionic Wave Function Formulation," in *2021 IEEE 24th Intern. Conf. on Information Fusion (FUSION)*, 2021, pp. 1–7.

[5] F. Govaers, B. Demissie, A. Khan, M. Ulmke, and W. Koch, "Tensor decomposition-based multitarget tracking in cluttered environments," *Journal of Advances in Information Fusion*, vol. 14, no. 1, pp. 86–97, 2019.

[6] W. Koch, *Tracking and Sensor Data Fusion*. Springer Berlin, 2014. [Online]. Available: http://link.springer.com/content/pdf/10.1007/978-3-642-39271-9.pdf

[7] F. L. Hitchcock, "The Expression of a Tensor or a Polyadic as a Sum of Products," *Journal of Mathematics and Physics*, vol. 6, pp. 164–189, 1927.

[8] A. Cichocki, R. Zdunek, A. H. Phan, and S.-i. Amari, *Nonnegative Matrix and Tensor Factorizations*, 2009, vol. 369, no. 1.

[9] W. Alshrafi, U. Engel, B. Cihan, J. Springer, G. Francois, and D. Heberling, "A dual-band dual-circular polarised antenna array for uav detection and tracking," in *2024 15th German Microwave Conference (GeMiC)*, 2024, pp. 69–72.

[10] J. Springer, M. Oispuu, W. Koch, and P. Knott, "Array calibration using neural networks," in *2023 IEEE 9th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2023, pp. 51–55.

[11] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004.

[12] F. Govaers, "Object Detection in Tensor Decomposition Based Multi Target Tracking," in *FUSION 2019 - 22nd International Conference on Information Fusion*. Institute of Electrical and Electronics Engineers Inc., 2019.

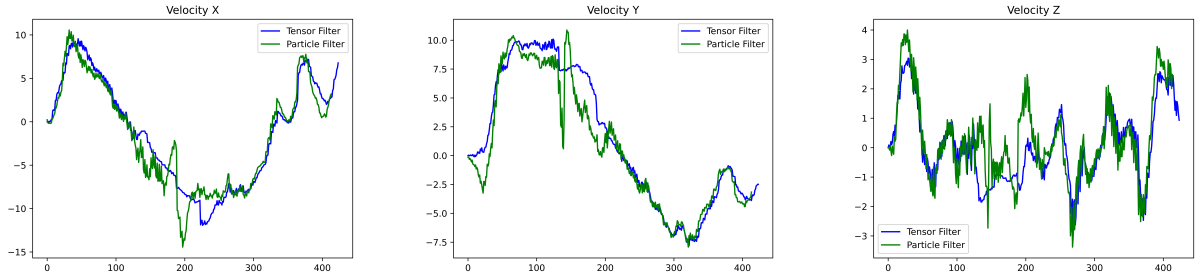

Fig. 5: The velocity components of the TbTT filter and the particle filter.

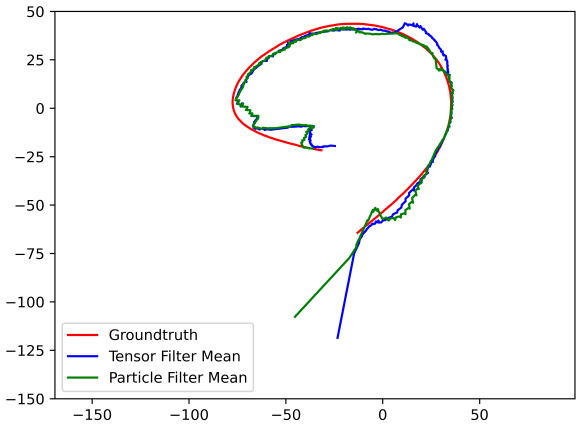

Fig. 6: The results of the TbTT filter and the particle filter in the x-y plane.

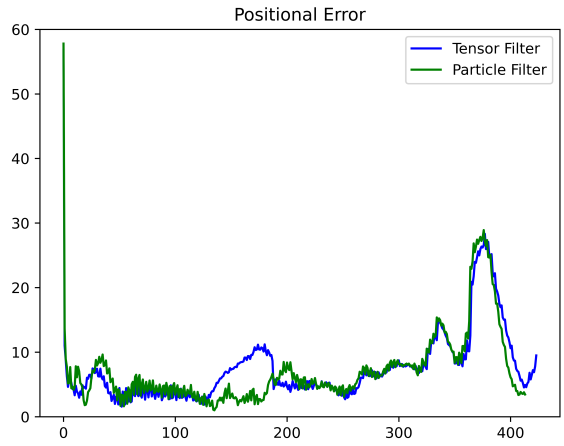

Fig. 7: The positional error of the TbTT filter and the particle filter.