The Kernel-SME Filter for Multiple Target Tracking

Marcus Baum and **Uwe D. Hanebeck**
Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
marcus.baum@kit.edu, uwe.hanebeck@ieee.org

***Abstract*—We present a novel method for tracking multiple targets, called Kernel-SME filter, that does not require an enumeration of measurement-to-target associations. This method is a further development of the symmetric measurement equation (SME) filter that removes the data association uncertainty of the original measurement equation with the help of a symmetric transformation. The key idea of the Kernel-SME filter is to define a symmetric transformation that maps the measurements to a Gaussian mixture function. This transformation is scalable to a large number of targets and allows for deriving a Gaussian state estimator that only has a cubic runtime complexity in the number of targets.**

***Index Terms*—Multiple Target Tracking, Data Association, Symmetric Measurement Equation.**

I. Introduction

A main challenge in multiple target tracking [1], [2] is that the association of measurements to targets is unknown. In this context, a variety of different multiple target tracking methods has been developed. For example, the *Joint Probabilistic Data Association Filter (JPDAF)* [3] enumerates all feasible association hypotheses in order to compute a Gaussian approximation of the posterior density of the target states. Unfortunately, the number of possible association hypotheses grows exponentially with the number of targets so that the tracking of a large number of closely-spaced targets becomes a serious challenge (see [4]–[7] for approaches for dealing with the complexity). The *Probability Hypothesis Density (PHD) filter* [8], [9] maintains the first moment of the multi-target posterior random set called PHD. By this means, association hypotheses are not explicitly enumerated, i.e., data association is performed implicitly. The PHD, however, contains less information than the full posterior random set [8], e.g., it does not maintain correlations between targets.

This article is about an implicit data association approach named *Symmetric Measurement Equation (SME)* filter [10], [11]. The SME filter removes the data association uncertainty from the original measurement equation using a symmetric transformation. This allows to bypass the combinatorial complexity of the data association problem. Unfortunately, existing SMEs suffer from strong nonlinearities and lack an intuitive semantic so that existing SME filters are not competitive to established approaches such as JPDAF and PHD filters.

In this article, we introduce the so-called Kernel-SME filter that can be seen as an extension of the SME approach. The basic idea is to define a symmetric transformation that maps the set of measurements to a function, i.e., a Gaussian mixture, and deterministic sampling of this function gives the symmetric transformation. In this manner, a data-dependent, i.e., non-parametric, symmetric transformation is obtained. The Kernel-SME has an intuitive semantic and it is suitable for a large number of closely-spaced targets due to a cubic time complexity. In this vein, the advantages of an implicit data association method are exploited while having the full joint density of the multi-target state available. In addition, there is an intriguing connection to the PHD filter that renders the Kernel-SME filter to an in-between of the PHD filter and the JPDAF.

The remainder of this paper is structured as follows: In the next section, we give a detailed description of the considered multi-target tracking problem. The subsequent Section III, introduces and discusses the original SME approach. The novel Kernel-SME filter is introduced in Section IV. In Section V, the Kernel-SME filter is compared by means of simulations with the Gaussian mixture implementation of the PHD filter [9] in various scenarios with a known number of targets. This paper is concluded in Section VI.

Remark 1. A preprint of this article is available at [12].

II. Problem Formulation

We consider the tracking of multiple targets based on noisy measurements, where the target-to-measurement association is unknown. Specifically, we make the following assumptions:

- A1 The number of targets is known.
- A2 Each target gives rise to exactly one single measurement per time instant, i.e., no missed detection.
- A3 There are no false measurements, i.e., each measurement originates from a target.

The n-dimensional single target state vectors are denoted with $\underline{\boldsymbol{x}}_k^1, \ldots, \underline{\boldsymbol{x}}_k^N$, where k denotes the discrete time and N is the number of targets[1]. The single target states are stacked in a joint target state vector $\underline{\boldsymbol{x}}_k = \left[(\underline{\boldsymbol{x}}_k^1)^T, \ldots, (\underline{\boldsymbol{x}}_k^N)^T\right]^T \in \mathbb{R}^{n \cdot N}$.

A. Measurement Model

At each time step k, N measurements $\underline{\boldsymbol{y}}_k^1, \ldots, \underline{\boldsymbol{y}}_k^N$ are available. Each measurement is related to a single target through the linear measurement model

$$\underline{\boldsymbol{y}}_k^{\pi_k(l)} = \mathbf{H}_k^l \underline{\boldsymbol{x}}_k^l + \underline{\boldsymbol{v}}_k^l \ , \qquad (1)$$

1 Note that vectors are underlined, e.g., $\underline{x}$, and random variables are printed in bold, e.g., $\boldsymbol{x}$ and $\underline{\boldsymbol{x}}$.

where $\pi_k \in \Pi_N$ is a permutation in the symmetric group Π_N that specifies the *unknown* target-to-measurement assignment and $\underline{\boldsymbol{v}}_k^l$ is additive zero-mean white noise with covariance matrix $\Sigma_{k,l}^v$. The single target measurement equations (1) can be composed to an overall measurement equation

$$\underbrace{\begin{bmatrix} \underline{\boldsymbol{y}}_k^{\pi_k(1)} \\ \vdots \\ \underline{\boldsymbol{y}}_k^{\pi_k(N)} \end{bmatrix}}_{=P_{\pi_k}(\underline{\boldsymbol{y}}_k)} = \underbrace{\begin{bmatrix} \mathbf{H}_k^1 & & \\ & \ddots & \\ & & \mathbf{H}_k^N \end{bmatrix}}_{=\mathbf{H}_k} \cdot \underbrace{\begin{bmatrix} \underline{\boldsymbol{x}}_k^1 \\ \vdots \\ \underline{\boldsymbol{x}}_k^N \end{bmatrix}}_{=\underline{\boldsymbol{x}}_k} + \underbrace{\begin{bmatrix} \underline{\boldsymbol{v}}_k^1 \\ \vdots \\ \underline{\boldsymbol{v}}_k^N \end{bmatrix}}_{=\underline{\boldsymbol{v}}_k} , \qquad (2)$$

where $\underline{\boldsymbol{y}}_k := \left[(\underline{\boldsymbol{y}}_k^1)^T, \ldots, (\underline{\boldsymbol{y}}_k^N)^T \right]^T$ and $P_{\pi_k}(\underline{\boldsymbol{y}}_k)$ permutes the single measurements in $\underline{\boldsymbol{y}}_k$ according to π_k.

B. System Model

The temporal evolution of a single target is specified by a linear motion model

$$\underline{\boldsymbol{x}}_{k+1}^l = \mathbf{A}_k^l \underline{\boldsymbol{x}}_k^l + \underline{\boldsymbol{w}}_k^l \ , \qquad (3)$$

where $\mathbf{A}_k^l$ is the system matrix and $\underline{\boldsymbol{w}}_k^l$ is additive white noise with covariance matrix $\Sigma_k^{w_l}$. The single target motion models (3) can be composed as

$$\underbrace{\begin{bmatrix} \underline{\boldsymbol{x}}_{k+1}^1 \\ \vdots \\ \underline{\boldsymbol{x}}_{k+1}^N \end{bmatrix}}_{=\underline{\boldsymbol{x}}_{k+1}} = \underbrace{\begin{bmatrix} \mathbf{A}_k^1 & & \\ & \ddots & \\ & & \mathbf{A}_k^N \end{bmatrix}}_{:=\mathbf{A}_k} \cdot \underbrace{\begin{bmatrix} \underline{\boldsymbol{x}}_k^1 \\ \vdots \\ \underline{\boldsymbol{x}}_k^N \end{bmatrix}}_{=\underline{\boldsymbol{x}}_k} + \underbrace{\begin{bmatrix} \underline{\boldsymbol{w}}_k^1 \\ \vdots \\ \underline{\boldsymbol{w}}_k^N \end{bmatrix}}_{=\underline{\boldsymbol{w}}_k} . \qquad (4)$$

III. SME-FILTER

This section is about the *Symmetric Measurement Equation (SME)* filter as introduced by Kamen [10], [11]. The basic idea of the SME filter is to remove the association uncertainty π_k from the measurement equation (2) by applying a symmetric transformation to the measurement vector.

Definition 1. A transformation $S(\underline{y})$ of a measurement vector $\underline{y}$ with $S \colon \mathbb{R}^{N \cdot n} \to \mathbb{R}^{N_a}$ is called symmetric if

$$S(\underline{y}) = S(P_\pi(\underline{y})) \qquad (5)$$

for all $\pi \in \Pi_N$.

Remark 2. Of course, the symmetric transformation should not remove information, i.e., it should be injective up to permutation.

Example 1. The *Sum-Of-Powers* [10], [11], [13], [14] transformation for two targets and one-dimensional measurements y^1 and y^2 is given by

$$S\left(\left[y^1, y^2 \right]^T \right) = \begin{bmatrix} y^1 + y^2 \\ \left(y^1\right)^2 + \left(y^2\right)^2 \end{bmatrix} .$$

The application of a symmetric function S to (2) yields

$$\underline{\boldsymbol{s}}_k := \underbrace{S(P_{\pi_k}(\underline{\boldsymbol{y}}_k))}_{=S(\underline{\boldsymbol{y}}_k)} = S(\mathbf{H}_k \cdot \underline{\boldsymbol{x}}_k + \underline{\boldsymbol{v}}_k) \ , \qquad (6)$$

where $\underline{\boldsymbol{s}}_k$ is a pseudo-measurement constructed from the original measurement vector $\underline{\boldsymbol{y}}_k$. The pseudo-measurement $\underline{\boldsymbol{s}}_k$ can be determined without knowing π_k due to the symmetry property of S. Hence, the data association uncertainty has been removed, however, instead a nonlinear measurement equation is obtained. Based on the nonlinear measurement equation (6), nonlinear Bayesian state estimators such as the Extended Kalman Filter (EKF) or Unscented Kalman Filter (UKF) [13], [14] can be used for performing inference.

Although the SME approach is a very neat way for dealing with data association uncertainties, it comes with some serious challenges:

1.) The generalization of existing symmetric transformations, i.e., the *Sum-Of-Powers* and [10], [11], [13]–[16], to states with dimension larger than 1 is nontrivial due to the so-called ghost target problem [13], [14] resulting from non-injective transformations. As a consequence, tedious and highly nonlinear symmetric functions that have no intuitive, physical meaning are obtained. Additionally, these symmetric transformations are unsuitable for larger target numbers as the order of the involved polynomial increases with the number of targets, i.e., for 10 targets polynomials up to order 10 are required.

2.) Due to 1.), the resulting nonlinear estimation problem is very difficult. As there is non-additive Gaussian noise in (6), the EKF cannot be applied directly and an approximate measurement equation with additive noise has to be derived first. The derivation of the additive noise term is usually complicated and time-consuming. Besides, Linear Regression Kalman Filters (LRKFs) such as the UKF [13], [14] and analytic moment calculation [17] do not give satisfying results due to the strong nonlinearities and numerical instabilities.

IV. KERNEL-SME FILTER

In the following, the Kernel-SME filter is introduced in two steps. First, the basic idea for constructing a symmetric measurement equation is discussed (see Section IV-A). Second, a particular Gaussian estimator is developed based on this symmetric measurement equation (see Section IV-B).

A. Kernel-SME

The basic idea of the Kernel-SME filter is to interpret the measurements as the parameters of a function, where the function is a sum of kernel functions that are placed at the measurement locations. We focus on Gaussian kernels, nevertheless other types of kernels may also be reasonable.

Definition 2 (Kernel Transformation)**.** Let $\mathcal{H}_n^N$ denote the space of all n-dimensional Gaussian mixtures with N components. The kernel transformation $S^K : \mathbb{R}^{N \cdot n} \to \mathcal{H}_n^N$, which maps $\underline{y} = \left[\underline{y}^1, \ldots, \underline{y}^N \right]^T \in \mathbb{R}^{N \cdot n}$ to a function $F_{\underline{y}} \in \mathcal{H}_n^N$, is defined as

$$S^K(\underline{y}) = F_{\underline{y}} \ \text{ with} \qquad (7)$$

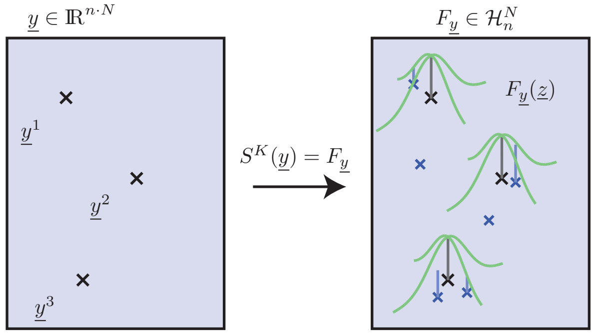


Fig. 1: Illustration of the Kernel-SME: The measurements are mapped to a Gaussian mixture function. The Gaussian mixture function may be evaluated at specific test points (small blue markers).

$$F_{\underline{y}}(\underline{z}) = \sum_{l=1}^{N} \mathcal{N}\left(\underline{z}; \underline{y}^l, \Gamma\right) \quad , \tag{8}$$

where $\mathcal{N}\left(\underline{z}; \underline{y}^l, \Gamma\right)$ is a Gaussian kernel located at $\underline{y}^l$ with kernel width Γ.

Remark 3. The transformation (7) is symmetric due to $F_{\underline{y}}(\underline{z}) = F_{P_\pi(\underline{y})}(\underline{z})$ for all $\underline{z} \in \mathbb{R}^n$. Furthermore, (7) is injective up to permutation due to the identifiability of the parameters of a multivariate Gaussian mixture density [18]. Hence, there is no ghost target problem (see for example [13]).

Remark 4. Note that the authors used kernel distances in [19] to extract optimal point estimates from multi-target probability densities. However, the data association problem has not been treated in [19].

The transformation (7) has an intuitive semantic: The set of measurements is interpreted as a continuous image, i.e., high values of $F_{\underline{y}}(\underline{z})$ indicate a high measurement concentration. Hence, the Kernel-SME reformulates the multi-target model for thresholded data, i.e., point measurements, as a model for unthresholded data (in a continuous domain). In this context, see track-before-detect algorithms [20]–[24] that directly work with unthresholded sensor data on a grid, i.e., a discrete domain.

As the kernel transformation (7) maps the measurement vector to a function, it can be seen as a generalization of the original SME approach [10], [11] that maps the measurement vector to vector again. Furthermore, this transformation is suitable for a large target number as with an increasing target number, only the number of summands (7) in increases.

Of course, the choice of a suitable kernel width Γ in (7) is essential. It should be chosen similar to the measurement noise covariance in order to ensure that the the kernels cover the predicted measurements.

The measurements contained in the stacked measurement vector $\underline{\boldsymbol{y}}_k$ can be used to form a random set $\{\underline{\boldsymbol{y}}_k^1, \ldots, \underline{\boldsymbol{y}}_k^N\}$ for which the first moment, i.e., the PHD, is [8]

$$D_{\underline{\boldsymbol{y}}_k}(\underline{z}) \quad := \quad \sum_i p_{\underline{y}_k^i}(\underline{z}) \quad , \tag{9}$$

where $p_{\underline{\boldsymbol{y}}_k^i}(\underline{z})$ is the probability density of $\underline{\boldsymbol{y}}_k^i$. The following theorem describes an insightful, inherent relationship between the transformed measurements $S^K(\underline{\boldsymbol{y}}_k)$ and the PHD of the stacked measurements $\underline{\boldsymbol{y}}_k$.

Theorem 1. *The expected kernel transformation of the stacked measurements $\underline{\boldsymbol{y}}_k$ coincides with the convolution of the PHD with the kernel, i.e.,* $\mathrm{E}\left\{F_{\underline{\boldsymbol{y}}_k}(\underline{z})\right\} = \int D_{\underline{\boldsymbol{y}}_k}(\underline{t}) \cdot \mathcal{N}(\underline{t}; \underline{z}, \Gamma) \, \mathrm{d}\underline{s}$.

PROOF. According to [19], the following holds

$$\begin{aligned}
\mathrm{E}\left\{F_{\underline{\boldsymbol{y}}_k}(\underline{z})\right\} &= \int F_{\underline{y}_k}(\underline{z}) \cdot p(\underline{y}_k) \, \mathrm{d}\underline{y}_k \\
&= \int \int \sum_i \delta(\underline{t} - \underline{y}_k^i) \cdot \mathcal{N}(\underline{t}; \underline{z}, \Gamma) \, \mathrm{d}\underline{t} \, p(\underline{y}_k) \, \mathrm{d}\underline{y}_k \\
&= \int \int \sum_i \delta(\underline{t} - \underline{y}_k^i) \cdot p(\underline{y}_k) \, \mathrm{d}\underline{y}_k \cdot \mathcal{N}(\underline{t}; \underline{z}, \Gamma) \, \mathrm{d}\underline{t} \\
&= \int D_{\underline{\boldsymbol{y}}_k}(\underline{t}) \cdot \mathcal{N}(\underline{t}; \underline{z}, \Gamma) \, \mathrm{d}\underline{t} \ .
\end{aligned}$$

□

As probabilistic inference in function spaces such as $\mathcal{H}_n^N$ may be tedious, we propose to evaluate the function $F_{\underline{\boldsymbol{y}}_k}(\underline{z})$ at specific test vectors $\underline{a}_k^1, \ldots, \underline{a}_k^{N_a}$. By this means, we obtain again a usual measurement equation that relates the state vector to a measurement vector. More precise, we define a "sampled" version of (7) as follows

$$S^K_{\underline{a}_k^1, \ldots, \underline{a}_k^{N_a}}(\underline{y}) = \begin{bmatrix} F_{\underline{y}}(\underline{a}_k^1) \\ \vdots \\ F_{\underline{y}}(\underline{a}_k^{N_a}) \end{bmatrix} \tag{10}$$

How to choose the number and locations of the test vectors is discussed in Section 5.

The application of (10) to (2) gives the following symmetric measurement equation

$$\underline{\boldsymbol{s}}_k = S^K_{\underline{a}_k^1, \ldots, \underline{a}_k^{N_a}}(\underline{\boldsymbol{y}}_k) = S^K_{\underline{a}_k^1, \ldots, \underline{a}_k^{N_a}}(\mathbf{H}_k \cdot \underline{\boldsymbol{x}}_k + \underline{\boldsymbol{v}}_k) \quad , \tag{11}$$

where $\underline{\boldsymbol{s}}_k$ is the pseudo-measurement.

The symmetric measurement equation (11) is data-dependent, i.e., it is constructed based on the measurements. In contrast, the traditional SME approach [10], [11] is data independent (see also Example 1).

Remark 5. Reasonably, the locations of the test vectors should be chosen such that the transformed measurement vector, i.e., the Gaussian mixture function $F_{\underline{y}}(\underline{z})$ in (8), is approximated well for all potential measurement vectors $\underline{y}$. Hence, we choose the test points depending on $\underline{\boldsymbol{y}}_k$ as the potential values of $\underline{y}$ are expected to lie around $\underline{\boldsymbol{y}}_k$. As $F_{\underline{\boldsymbol{y}}_k}(\underline{z})$ is a Gaussian mixture, the test vectors can be seen as deterministic samples of the Gaussian mixture and there is a strong relationship to deterministic sampling problems that occur for example in the UKF [25]. Due to this analogy, we propose to add $2 \cdot n$ test vectors for each Gaussian component $\mathcal{N}\left(\underline{z}; \underline{\boldsymbol{y}}_k^l, \Gamma\right)$ in (8) according to the deterministic sampling rule of the UKF

[25], i.e., the total number of test vectors is $N_a = 2 \cdot n \cdot N$. It is important to note that same test points are used for the left hand side and right hand side of the measurement equation (11).

Example 2. For two targets and one-dimensional measurements y^1 and y^2, (10) becomes

$$S^K\left(\left[y^1, y^2\right]^T\right) = \begin{bmatrix} \mathcal{N}\left(a_k^1; y^1, \Gamma\right) + \mathcal{N}\left(a_k^1; y^2, \Gamma\right) \\ \mathcal{N}\left(a_k^2; y^1, \Gamma\right) + \mathcal{N}\left(a_k^2; y^2, \Gamma\right) \end{bmatrix}$$

when using two test vectors a_k^1 and a_k^2. Note the test vectors depend on the time index as they are chosen depending on the particular measurement $\underline{\boldsymbol{y}}_k^l$.

B. Gaussian Estimator

Based on the (association-free) measurement equation (11), we develop a Gaussian state estimator for the targets, i.e., a Gaussian approximation of the posterior probability density function for $\underline{\boldsymbol{x}}_k$ given the pseudo-measurements $\mathbf{S}_k := \{\underline{\boldsymbol{s}}_1, \ldots, \underline{\boldsymbol{s}}_k\}$

$$p(\underline{x}_k \,|\, \mathbf{S}_k) = \mathcal{N}\left(\underline{x}_k; \underline{\mu}_k^x, \Sigma_k^x\right) \tag{12}$$

is to be computed, where $\underline{\mu}_k^x$ is the mean and Σ_k^x the covariance matrix of the Gaussian.

1) Time Update: The time update step determines $p(\underline{x}_k \,|\, \mathbf{S}_{k-1}) = \mathcal{N}\left(\underline{x}_k; \underline{\mu}_{k|k-1}^x, \Sigma_{k|k-1}^x\right)$ based on the previous density $p(\underline{x}_{k-1} \,|\, \mathbf{S}_{k-1})$. Due to the linear system model, the prediction can be performed according the Kalman filter formulas

$$\underline{\mu}_{k|k-1}^x = \mathbf{A}_k \cdot \underline{\mu}_{k-1}^x \text{ , and} \tag{13}$$

$$\Sigma_{k|k-1}^x = \mathbf{A}_k \Sigma_{k-1}^x (\mathbf{A}_k)^T + \Sigma_k^w \text{ ,} \tag{14}$$

where Σ_k^w denotes the covariance matrix of the stacked system noise $\underline{\boldsymbol{w}}_k$. In the measurement update step, the prediction $\mathcal{N}\left(\underline{x}_k; \underline{\mu}_{k|k-1}^x, \Sigma_{k|k-1}^x\right)$ is updated with the stacked measurement vector $\underline{\boldsymbol{y}}_k$.

2) Measurement Update: In order to perform the measurement update, we propose to derive a *Linear Minimum Mean Squared Error (LMMSE)* estimator [26] based on (11). For a given prediction of the state $\underline{\mu}_{k|k-1}^x$ with covariance matrix $\Sigma_{k|k-1}^x$, the updated estimate $\underline{\mu}_k^x$ and Σ_k^x according to (11) is given by the Kalman filter formulas

$$\underline{\mu}_k^x = \underline{\mu}_{k|k-1}^x + \Sigma_k^{xs}(\Sigma_k^{ss})^{-1}\left(\underline{\boldsymbol{s}}_k - \underline{\mu}_k^s\right) \text{ , and} \tag{15}$$

$$\Sigma_k^x = \Sigma_{k|k-1}^x - \Sigma_k^{xs}(\Sigma_k^{ss})^{-1}\Sigma_k^{sx}, \tag{16}$$

where

- $\underline{\boldsymbol{s}}_k$ is the pseudo-measurement,
- $\underline{\mu}_k^s$ is the predicted pseudo-measurement,
- Σ_k^{xs} is the covariance between the state vector $\underline{\boldsymbol{x}}_k$ and the pseudo-measurement $\underline{s}_k$, and
- Σ_k^{ss} is the variance of the pseudo-measurement $\underline{s}_k$.

Closed-form expressions for the above moments are derived in Appendix A.

Remark 6. According to Theorem 1 the predicted mean $\underline{\mu}_k^s$ can be interpreted as the PHD of the predicted measurement smoothed with a kernel (evaluated at the test points). The pseudo-measurement $\underline{\boldsymbol{s}}_k$ can be interpreted as the smoothed measurements (evaluated at the test points). Hence, the above linear filter minimizes the kernel distance [19], [27] between the measurements and the PHD of the predicted measurements.

Fig. 2 depicts pseudo-code of the overall algorithm for the measurement update.

- Step 1 computes the test vectors by calculating deterministic samples of the Gaussians according to the sampling rule of the UKF (linear time complexity in the number of measurements).
- Step 2 calculates the pseudo-measurement according to (10) (quadratic time complexity as each component of the pseudo-measurement can be determined in linear time).
- Step 3 calculates the moments required for performing the LMMSE update in (15) and (16), see also Appendix A. The mean in Step 3b is determined in quadratic time. The covariance matrix in Step 3c is calculated with a cubic time complexity as each entry can be determined in linear time. Note that the double sum in Step 3c can be determined in linear time as the inner sum does not have to be computed from scratch for each summand. Step 3d determines the cross-covariance matrix in cubic time.
- Step 4 performs the LMMSE update that has a cubic time complexity.

All in all, the measurement update for the Kernel-SME filter as presented here has a cubic time complexity.

V. Evaluation

The performance of the Kernel-SME filter is demonstrated with respect to the Gaussian mixture implementation of the PHD filter (GM-PHD) [9]. Note that the PHD filter is also capable to deal with false detections, missed detections, and an unknown number of targets. Here, however, we assume the number of targets to be given and neither false nor missed detections (see Section II) may occur. Under these assumptions, the total mass of the PHD always coincides with the exact number of targets.

We consider three different scenarios, where all scenarios consider targets that evolve according to a random walk model, i.e., the dimension of the state vector is $n = 2$ and the system matrix is the identity matrix, i.e., $\mathbf{H}_k^i = \mathbf{A}_k^i = \mathbf{I}_2$ for $i = 1 \ldots N$, where $\mathbf{I}_2$ is the identity matrix of dimension 2. Furthermore, we assume that two-dimensional position measurements of the targets are available, i.e., the measurement matrix is $\mathbf{H}_k^i = \mathbf{I}_2$ for $i = 1 \ldots N$.

In all scenarios, the Kernel-SME filter employs a kernel with covariance matrix $\Sigma = \mathbf{I}_2$. The GM-PHD filter maintains a Gaussian mixture with 50 components in order to represent the PHD. The parameters for the Gaussian mixture reduction have been optimized for the best results and the mixture components with the largest weights serve as point estimates for the single targets.

Input:

- Mean $\underline{\mu}_{k|k-1}^{x} = \left[(\underline{\mu}_{k|k-1}^{x_1})^T, \ldots, (\underline{\mu}_{k|k-1}^{x_N})^T\right]^T$ and covariance matrix $\Sigma_k^x = \left(\Sigma_{k|k-1}^{x_i x_j}\right)_{i,j=1\ldots N}$
- Measurements $\underline{\boldsymbol{y}}_k^1, \ldots, \underline{\boldsymbol{y}}_k^N$ (order of measurements is irrelevant)

Output:

- Updated mean $\underline{\mu}_k^x = \left[(\underline{\mu}_k^{x_1})^T, \ldots, (\underline{\mu}_k^{x_N})^T\right]^T$ and covariance matrix $\Sigma_k^x = \left(\Sigma_k^{x_i x_j}\right)_{i,j=1\ldots N}$

Algorithm:

1) Determine test vectors $\underline{a}_k^1, \ldots, \underline{a}_k^{N_a}$ with $N_a = 2 \cdot n \cdot N$ according to

$$\underline{a}_k^{l+i-1} := \underline{\boldsymbol{y}}_k^l + \left(\sqrt{n\Gamma}\right)_i \quad \text{and} \quad \underline{a}_k^{l+2(i-1)} := \underline{\boldsymbol{y}}_k^l - \left(\sqrt{n\Gamma}\right)_i$$

for $i = 1, \ldots, N$ and $l = 1, \ldots, n$, where $\left(\sqrt{n\Gamma}\right)_i$ denotes the i-th column of $\sqrt{n\Gamma}$.

2) Compute pseudo-measurement $\underline{\boldsymbol{s}}_k = \left[\underline{\boldsymbol{s}}_k^1, \ldots, \underline{\boldsymbol{s}}_k^{N_a}\right]^T$ with

$$\underline{\boldsymbol{s}}_k^i = \sum_{l=1}^{N} \mathcal{N}\left(\underline{a}_k^i; \underline{\boldsymbol{y}}_k^l, \Gamma\right)$$

.

3) Determine moments for LMMSE update:

   a) Define function $P_l^\Gamma(\underline{z}) := \mathcal{N}\left(\underline{z}; \mathbf{H}_k^l \underline{\mu}_l^x, \mathbf{H}_k^l \Sigma_{k|k-1}^{x_l} (\mathbf{H}_k^l)^T + \Sigma_k^v + \Gamma\right)$

   b) Mean $\underline{\mu}_k^s = \left[\underline{\mu}_k^{s_1}, \ldots, \underline{\mu}_k^{s_{N_a}}\right]^T$ of predicted pseudo-measurement:

$$\underline{\mu}_{k,i}^s = \sum_{l=1\ldots N} P_l^\Gamma(\underline{a}_k^i)$$

   c) Covariance $\Sigma_k^{ss} = (\Sigma_k^{s_i s_j})_{i,j=1,\ldots,N_a}$ of predicted pseudo-measurement:

$$\Sigma_k^{s_i s_j} = \left(\sum_{l=1}^{N} P_l^\Gamma(\underline{a}_k^i) \sum_{m=1, m\neq l}^{N} P_l^\Gamma(\underline{a}_k^j)\right) + \mathcal{N}\left(\underline{a}_k^i; \underline{a}_k^j, 2\Gamma\right) \cdot \sum_{l=1\ldots N} P_l^{0.5\Gamma}\left(\tfrac{1}{2}(\underline{a}_k^i + \underline{a}_k^j)\right) - \underline{\mu}_{k,i}^s \cdot \underline{\mu}_{k,j}^s$$

   d) Cross-covariance $\Sigma_k^{xs} = \left[\Sigma_k^{xs_1}, \ldots, \Sigma_k^{xs_{N_a}}\right]$ between predicted pseudo-measurement:

$$\begin{aligned}
\Sigma_k^{xs_i} &= -\underline{\mu}_k^x \cdot \underline{\mu}_k^{s_i} + \sum_{l=1\ldots N} P_l^\Gamma(\underline{a}_k^i) \cdot (\underline{\mu}_{k|k-1}^x + \mathbf{K}_k^l(\underline{a}_k^i - \mathbf{H}_k^l \underline{\mu}_k^{x_l})) \text{ , where} \\
\mathbf{K}_k^l &= \left[\Sigma_{k|k-1}^{x_1 x_l}, \ldots, \Sigma_{k|k-1}^{x_N x_l}\right]^T \mathbf{H}_k^l \cdot \left(\mathbf{H}_k^l \Sigma_{k|k-1}^{x_l} (\mathbf{H}_k^l)^T + \Gamma + \Sigma_k^v\right)^{-1}
\end{aligned}$$

4) Perform LMMSE update:

$$\begin{aligned}
\underline{\mu}_k^x &= \underline{\mu}_{k|k-1}^x + \Sigma_k^{xs} (\Sigma_k^{ss})^{-1} \left(\underline{\boldsymbol{s}}_k - \underline{\mu}_k^s\right) \text{ , and} \\
\Sigma_k^x &= \Sigma_{k|k-1}^x - \Sigma_k^{xs} (\Sigma_k^{ss})^{-1} \Sigma_k^{sx} \text{ .}
\end{aligned}$$

Fig. 2: Pseudo-code for a measurement update according to the Kernel-SME filter (see [28] for an example implementation).

The first estimate for the Kernel-SME filter is initialized with the covariance matrix $\Sigma_0^x = 0.5\,\mathbf{I}_{2N}$ and the mean $\underline{\mu}_0^x$ is sampled randomly from $\mathcal{N}\left(\tilde{\underline{x}}_0; \underline{0}, \Sigma_0^x\right)$, where $\tilde{\underline{x}}_0$ denotes the true target position at time instant 0. The GM-PHD filter is initialized with the corresponding PHD.

Scenario 1: Eight Targets, Large Noise

In the first scenario, eight targets are initialized in a grid as indicated in Fig. 3a. The system noise is $\Sigma_k^w = 0.05\,\mathbf{I}_2$ (Fig. 3a shows an example run). The measurement noise is given by $\Sigma_k^v = 0.7\,\mathbf{I}_2$, which is high compared to the inter-target distance (see Fig. 4b). As a consequence, the tracking of the targets is a serious challenge for both filters. As the PHD filter itself does not maintain target labels, the performance of both filters is assessed with the *Optimal Sub-Pattern Assignment (OSPA)* metric [29] that ignores target labels. The averaged OSPA distance over 30 Monte Carlo runs is depicted Fig. 3c. The Kernel-SME filter outperforms the GM-PHD in this scenario. The reason is that the GM-PHD filter tends to merge closely-spaced targets in a bulk so that they cannot be differed anymore.

Scenario 2: Eight Targets, Medium Noise

The second scenario coincides exactly with the first scenario, however, the measurement noise is smaller, i.e., $\Sigma_k^v = 0.3\,\mathbf{I}_2$. Both filters give better estimates than for the first scenario. Nevertheless, the Kernel-SME filter still performs better than the PHD filter according to Fig. 3c.

Scenario 3: Two Targets, Correlated System Noise

The purpose of the second scenario is to convey the effect of neglecting the correlation between target states. For this purpose, two targets whose temporal evolutions are correlated is investigated. Specifically, the overall system noise (3) is set to

$$\Sigma_k^w = 1.5 \cdot \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \quad ,$$

which essentially means that the x- and y-coordinates of both targets are fully correlated. From a practical point of view, such correlated system noise may be caused by disturbances affecting both targets, e.g., wind may cause a similar effect in air-surveillance as it is common to all targets. The measurement noise is set to $\Sigma^v = 0.1\,\mathbf{I}_2$ in this scenario, which is not that high (in comparison with the distance of the targets). As the PHD is not capable of representing correlations between the targets, the Kernel-SME filter performs significantly better than the GM-PHD filter in this scenario.

VI. Conclusions and Future Work

This article presented a novel type of SME filter that is based on a mapping from the measurements to a Gaussian mixture. Intuitively, the filter recursively minimizes the kernel distance between the measurements and the PHD of the predicted measurements. By this means, shortcomings of existing SME approaches are remedied so that the Kernel-SME filter is a competitive alternative to traditional tracking algorithms such as JPDAF and PHD filters. The Kernel-SME filter is particular advantageous for a large number of closely-spaced targets.

Future investigations will concentrate on the effect of the kernel width and the selected test points (number and locations) in the Kernel-SME filter update. In this context, it may be interesting to employ filtering techniques for infinite-dimensional measurement spaces in order to bypass the need for selecting specific test vectors. Finally, the Kernel-SME filter will be extended to missed detections and clutter measurements in order render it more appealing for real-world scenarios (see A2 and A3 in Section II). Of course, a more exhaustive evaluation and comparison with other multi-target tracking algorithms will highlight further strengths and weaknesses of the Kernel-SME filter.

References

[1] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House Publishers, 2004.

[2] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, 2011.

[3] Y. Bar-Shalom, F. Daum, and J. Huang, “The Probabilistic Data Association Filter,” *IEEE Control Systems Magazine*, vol. 29, no. 6, pp. 82–100, December 2009.

[4] P. Horridge and S. Maskell, “Real-Time Tracking Of Hundreds Of Targets With Efficient Exact JPDAF Implementation,” in *9th International Conference on Information Fusion (Fusion 2006)*, July 2006, pp. 1 –8.

[5] S. Oh, S. Russell, and S. Sastry, “Markov Chain Monte Carlo Data Association for Multi-Target Tracking,” *IEEE Transactions on Automatic Control*, vol. 54, no. 3, pp. 481 –497, march 2009.

[6] J. Williams and R. Lau, “Data Association by Loopy Belief Propagation,” in *2010 13th Conference on Information Fusion (Fusion 2010)*, July 2010, pp. 1 –8.

[7] D. F. Crouse, Y. Bar-Shalom, P. Willett, and L. Svensson, “The JPDAF in Practical Systems: Computation and Snake Oil,” in *Proceedings of Signal Processing of Small Targets, SPIE, 7698*, 2010, pp. 769 813–769 813–15.

[8] R. P. S. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Boston, Mass.: Artech House, 2007.

[9] B.-N. Vo and W.-K. Ma, “The Gaussian Mixture Probability Hypothesis Density Filter,” *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091 –4104, Nov. 2006.

[10] E. Kamen, “Multiple Target Tracking based on Symmetric Measurement Equations,” *IEEE Transactions on Automatic Control*, vol. 37, no. 3, pp. 371 –374, March 1992.

[11] E. Kamen and C. Sastry, “Multiple Target Tracking using Products of Position Measurements,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 2, pp. 476 –493, April 1993.

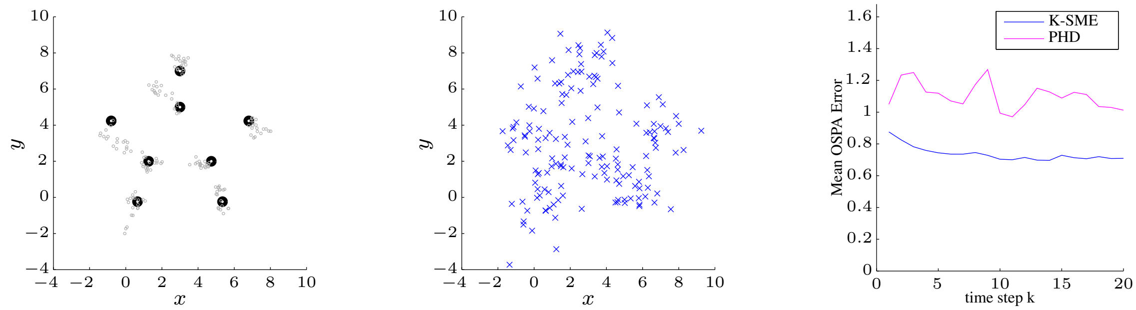
[12] M. Baum and U. D. Hanebeck, “The Kernel-SME Filter for Multiple Target Tracking,” *arXiv preprint: Systems and Control (cs.SY)*, 2012. [Online]. Available: http://arxiv.org/abs/1212.5882

[13] W. Leven and A. Lanterman, “Multiple Target Tracking with Symmetric Measurement Equations using Unscented Kalman and Particle Filters,” in *Proceedings of the Thirty-Sixth Southeastern Symposium on System Theory*, 2004, pp. 195 – 199.

[14] ——, “Unscented Kalman Filters for Multiple Target Tracking With Symmetric Measurement Equations,” *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 370 –375, February 2009.

[15] D. J. Muder and S. D. O’Neil, “Multidimensional SME Filter for Multitarget Tracking,” *Signal and Data Processing of Small Targets 1993*, vol. 1954, no. 1, pp. 587–599, 1993.

[16] Swati and S. Bhaumik, “Multiple Target Tracking based on Homogeneous Symmetric Transformation of Measurements,” *Aerospace Science and Technology*, 2012.

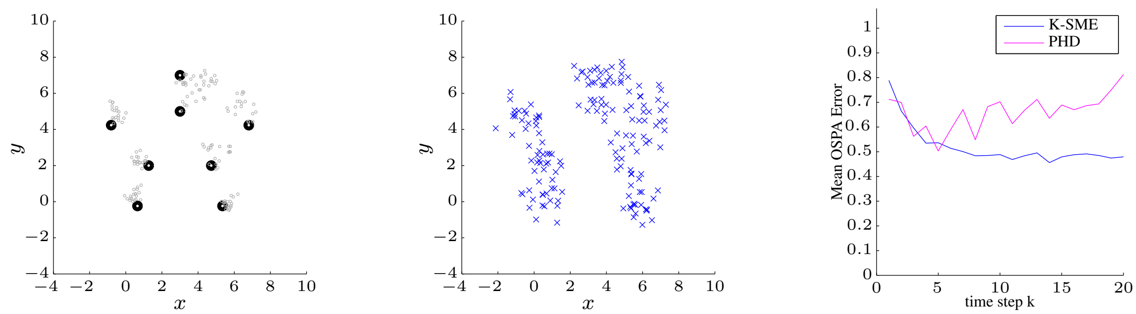


(a) Initial target positions (big dots) and an example random walk (small dots).

(b) Measurements from the example random walk.

(c) Mean OSPA error (for 30 runs).

Fig. 3: Scenario 1: Setting and results.

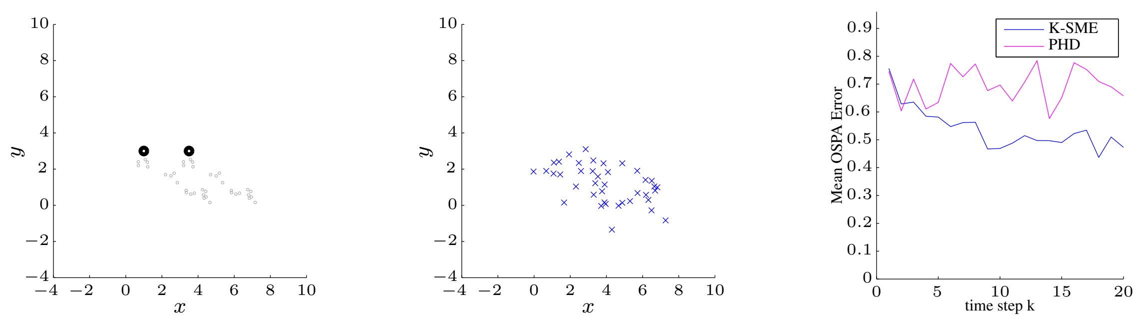


(a) Initial target positions (big dots) and an example random walk (small dots).

(b) Measurements from the example random walk.

(c) Mean OSPA error (for 30 runs).

Fig. 4: Scenario 2: Setting and results.



(a) Initial target positions (big dots) and an example random walk (small dots).

(b) Measurements from the example random walk.

(c) Mean OSPA error (for 30 runs).

Fig. 5: Scenario 3: Setting and results.

[17] M. Baum, B. Noack, F. Beutler, D. Itte, and U. D. Hanebeck, "Optimal Gaussian Filtering for Polynomial Systems Applied to Association-free Multi-Target Tracking," in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, Chicago, Illinois, USA, Jul. 2011.

[18] S. J. Yakowitz and J. D. Spragins, "On the Identifiability of Finite Mixtures," *The Annals of Mathematical Statistics*, vol. 39, no. 1, pp. 209–214, 1968.

[19] M. Baum, P. Ruoff, D. Itte, and U. D. Hanebeck, "Optimal Point Estimates for Multi-target States based on Kernel Distances," in *Proceedings of the 51st IEEE Conference on Decision and Control (CDC 2012)*, Maui, Hawaii, USA, Dec. 2012.

[20] R. L. Streit, M. L. Graham, and M. J. Walsh, "Multitarget Tracking of Distributed Targets Using Histogram-PMHT," *Digital Signal Processing*, vol. 12, no. 23, pp. 394 – 404, 2002.

[21] Y. Boers, H. Driessen, J. Torstensson, M. Trieb, R. Karlsson, and F. Gustafsson, "Track-Before-Detect Algorithm for Tracking Extended Targets," *IEE Proceedings on Radar, Sonar and Navigation*, vol. 153, no. 4, pp. 345–351, Aug. 2006.

[22] S. J. Davey, M. G. Rutten, B. Cheung, and B. Cheung, "A Comparison of Detection Performance for Several Track-Before-Detect Algorithms," in *11th International Conference on Information Fusion*, July 2008, pp. 1 –8.

[23] W. Yi, M. Morelande, L. Kong, and J. Yang, "A computationally efficient particle filter for multitarget tracking using an independence approximation," *IEEE Transactions on Signal Processing*, vol. 61, no. 4, pp. 843–856, 2013.

[24] M. Morelande, C. Kreucher, and K. Kastella, “A Bayesian Approach to Multiple Target Detection and Tracking,” *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 1589–1604, May 2007.

[25] S. J. Julier and J. K. Uhlmann, “Unscented Filtering and Nonlinear Estimation,” in *Proceedings of the IEEE*, vol. 92, no. 3, 2004, pp. 401–422.

[26] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: John Wiley & Sons, Inc., 2002.

[27] J. M. Phillips and S. Venkatasubramanian, “A Gentle Introduction to the Kernel Distance,” *CoRR*, vol. abs/1103.1625, 2011.

[28] M. Baum, “Kernel-SME Filter Matlab Implementation,” CloudRunner Wep Page, April 2013. [Online]. Available: http://www.cloudrunner.eu/algorithm/117/kernel-sme-filter/

[29] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, “A Consistent Metric for Performance Evaluation of Multi-Object Filters,” *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447 –3457, 2008.

Appendix
Closed-Form Expressions of the Moments for the LMMSE Estimator

In the following, analytic expressions for the moments in the measurement update step (15) and (16) are derived (see also the pseudo-code in Fig. 2). Essentially, the derivations are straightforward as they all can be performed with the help of the Kalman filtering formulas. First, we define the helper function

$$P_l^\Gamma(\underline{z}) := \mathcal{N}\left(\underline{z}; \mathbf{H}_k^l \underline{\mu}_k^{x_l}, \mathbf{H}_k^l \Sigma_{k|k-1}^{x_l} (\mathbf{H}_k^l)^T + \Sigma_k^{v_l} + \Gamma\right) \quad .$$

depending $\underline{z} \in \mathbb{R}^n$ that allows to abbreviate the resulting expressions.

A. Mean of Predicted Pseudo-Measurement

The mean of the predicted pseudo-measurement $\underline{\mu}_k^s = \left[\underline{\mu}_k^{s_1}, \ldots, \underline{\mu}_k^{s_{N_a}}\right]^T$ is

$$\begin{aligned}
\underline{\mu}_k^{s_i} &= \mathrm{E}\Big\{ F_{\underline{\boldsymbol{y}}_k}(\underline{a}_k^i) \,|\, \mathcal{Y}_{k-1} \Big\} && (17)\\
&= \sum_{l=1}^{N} \int \mathcal{N}\left(\underline{a}_k^i; \mathbf{H}_k^l \underline{x}_k^l + \underline{v}_k^l, \Gamma\right) \cdot && (18)\\
&\quad \mathcal{N}\left(\underline{x}_k; \underline{\mu}_{k|k-1}^x, \Sigma_{k|k-1}^x\right) \cdot \mathcal{N}\left(\underline{v}_k; \underline{0}, \Sigma_k^v\right) \,\mathrm{d}\underline{x}_k^l \,\mathrm{d}\underline{v}_k^l \\
&= \sum_{l=1}^{N} P_l^\Gamma(\underline{a}_k^i) && (19)
\end{aligned}$$

The simplification from (18) to (19) follows from the fact that the test vector $\underline{a}_k^i$ in (18) can be interpreted as a measurement of the state vector $\underline{x}_k$ under the measurement model $\mathbf{H}_k^l \underline{x}_k^l + \bar{\underline{v}}_k^l$, where $\bar{\underline{v}}_k^l$ is Gaussian noise with covariance matrix $\Gamma + \Sigma_k^{v_l}$. With this interpretation in mind, the summands in (19) result from evaluating the corresponding measurement probability density at $\underline{a}_k^i$.

B. Cross-Covariance

The cross-covariance matrix between the multi-target state vector and the pseudo-measurement becomes $\Sigma_k^{xs} = \left[\Sigma_k^{xs_1}, \ldots, \Sigma_k^{xs_{N_a}}\right]$ with

$$\Sigma_k^{xs_i} = \underbrace{\mathrm{E}\Big\{ \underline{\boldsymbol{x}}_k \cdot F_{\underline{\boldsymbol{y}}_k}(\underline{a}_k^i) \,|\, \mathcal{Y}_{k-1} \Big\}}_{(*)} - \underline{\mu}_k^x \cdot \underline{\mu}_{k,i}^s \ ,$$

where

$$\begin{aligned}
(*) &= \sum_{l=1}^{N} \int \underline{x}_k \cdot \mathcal{N}\left(\underline{a}_k^i; \mathbf{H}_k^l \underline{x}_k^l + \underline{v}_k^l, \Gamma\right) \cdot && (20)\\
&\quad \mathcal{N}\left(\underline{x}_k; \underline{\mu}_{k|k-1}^x, \Sigma_{k|k-1}^x\right) \cdot \mathcal{N}\left(\underline{v}_k; \underline{0}, \Sigma_k^v\right) \,\mathrm{d}\underline{x}_k \,\mathrm{d}\underline{v}_k^l \\
&= \sum_{l=1\ldots N} P_l^\Gamma(\underline{a}_k^i) \cdot \left(\underline{\mu}_{k|k-1}^x + \mathbf{K}_k^l (\underline{a}_k^i - \mathbf{H}_k^l \underline{\mu}_k^{x_l})\right) && (21)
\end{aligned}$$

and

$$\begin{aligned}
\mathbf{K}_k^l &= \left[\Sigma_{k|k-1}^{x_1 x_l}, \ldots, \Sigma_{k|k-1}^{x_N x_l}\right]^T \cdot \mathbf{H}_k^l \cdot \\
&\quad \left(\mathbf{H}_k^l \Sigma_{k|k-1}^{x_l} (\mathbf{H}_k^l)^T + \Gamma + \Sigma_k^{v_l}\right)^{-1} \quad . && (22)
\end{aligned}$$

The result (21) follows from (20) the Kalman filter update equations by interpreting again $\underline{a}_k^i$ as a measurement of the state vector $\underline{x}_k$ under the measurement model $\mathbf{H}_k^l \underline{x}_k^l + \bar{\underline{v}}_k^l$. The corresponding Kalman gain is given in (22). The term $P_l^\Gamma(\underline{a}_k^i)$ results from normalizing the product of Gaussians.

C. Covariance Matrix of Pseudo-Measurement

The covariance matrix of the predicted pseudo-measurement $\Sigma_k^{ss} = (\Sigma_k^{s_i s_j})_{i,j=1,\ldots,N_a}$ can be calculated with

$$\Sigma_k^{s_i s_j} = \underbrace{\mathrm{E}\Big\{ F_{\underline{\boldsymbol{y}}_k}(\underline{a}_k^i) \cdot F_{\underline{\boldsymbol{y}}_k}(\underline{a}_k^j) \,|\, \mathcal{Y}_{k-1} \Big\}}_{(**)} - \underline{\mu}_{k,i}^s \cdot \underline{\mu}_{k,j}^s \ , \text{ where}$$

$$\begin{aligned}
(**) &= \sum_{l=1}^{N} \sum_{m=1}^{N} \int \mathcal{N}\left(\underline{a}_k^i; \mathbf{H}_k^l \underline{x}_k^l + \underline{v}_k^l, \Gamma\right) \cdot \\
&\quad \mathcal{N}\left(\underline{a}_k^j; \mathbf{H}_k^m \underline{x}_k^m + \underline{v}_k^m, \Gamma\right) \cdot \mathcal{N}\left(\underline{x}_k; \underline{\mu}_{k|k-1}^x, \Sigma_{k|k-1}^x\right) \cdot \\
&\quad \mathcal{N}\left(\underline{v}_k; \underline{0}, \Sigma_k^v\right) \,\mathrm{d}\underline{x}_k \,\mathrm{d}\underline{v}_k && (23)\\
&= \left(\sum_{l=1}^{N} P_l^\Gamma(\underline{a}_k^i) \sum_{m=1, m\neq l}^{N} P_l^\Gamma(\underline{a}_k^j) \right) + \\
&\quad \mathcal{N}\left(\underline{a}_k^i; \underline{a}_k^j, 2\Gamma\right) \cdot \sum_{l=1\ldots N} P_l^{0.5\Gamma}\left(\tfrac{1}{2}(\underline{a}_k^i + \underline{a}_k^j)\right) \ . && (24)
\end{aligned}$$

In analogy to (19) for the mean, the reformulation from (23) to (24) results from the measurement pdf for $\underline{a}_k^i$ and $\underline{a}_k^j$. In case $l = m$ in (23), the identity $\mathcal{N}\left(\underline{a}_k^i; \mathbf{H}_k^l \underline{x}_k^l + \underline{v}_k^l, \Gamma\right) \cdot \mathcal{N}\left(\underline{a}_k^j; \mathbf{H}_k^l \underline{x}_k^l + \underline{v}_k^l, \Gamma\right) = \mathcal{N}\left(\underline{a}_k^i; \underline{a}_k^j, 2\Gamma\right) \cdot \mathcal{N}\left(\frac{1}{2}(\underline{a}_k^i + \underline{a}_k^j); \mathbf{H}_k^l \underline{x}_k^l + \underline{v}_k^l, \frac{1}{2}\Gamma\right)$ has to be exploited (can be proven using again the Kalman filter update formulas).