Distributed Kalman Filter Fusion at Arbitrary Instants of Time

Felix Govaers
Sensor Data and Information Fusion
and Institute of Computer Science 4
Fraunhofer FKIE and University of Bonn
Wachtberg and Bonn, Germany
govaers@cs.uni-bonn.de

Wolfgang Koch
Sensor Data and Information Fusion
Fraunhofer FKIE
Wachtberg
wolfgang.koch@fkie.fraunhofer.de

Abstract – *Track-to-track fusion aims at combining locally preprocessed information of individual sensors optimally, i.e. in a way that is equivalent to fusing all measurements of all sensors directly. It is well known that this can be achieved if the local sensor tracks produced at all individual scan times are available in the fusion center. Full-rate communication in this sense, however, is impractical in certain applications. In [1] W. Koch thus proposes a distributed Kalman-type processing scheme, which provides optimal track-to-track fusion results at* arbitrarily chosen *instants of time by communicating and combining the local sensor 'tracks' referring to this time. However, this scheme needs an exchange of sensor data for the track prediction and retrodiction, if the number of sensors exceeds two. Therefore, we present an improvement, which extends the algorithm to arbitrary sensor count.*

Keywords: track-to-track fusion, distributed Kalman filtering, distributed retrodiction, reduced-rate communication

1 Introduction

Practical needs as well as the rapid development of information and communications technology are driving factors of ever increasing networks of surveillance sensors. To a degree never known before, decision makers have access to vast amounts of sensor data. To use this information potential appropriately, the data streams must not overwhelm the human beings involved. On the contrary, the data are to be condensed ("fused") in such a way that high-quality information results, serving as a basis for decision support in many applications.

Among the products of sensor data fusion, so-called 'tracks' are of particular importance. Tracks represent the available knowledge on relevant, time varying state quantities of individual targets. Quantitative measures describing the quality of this knowledge are part of the tracks. The information obtained by 'tracking' algorithms [2, 5, 15] also includes the history of the targets. If possible, a one-to-one association between the targets in the sensors' field of view and the produced tracks is to be established and has to be preserved as long as possible (track continuity). The achievable track quality does not only depend on the performance of the sensors used, but also on the target properties and the operational conditions within the scenario to be observed.

1.1 Bayesian Target Tracking

Let all time-varying target properties of interest at a given time t_k be collected by a state vector $\mathbf{x}_k$ at time t_k, whose temporal evolution is modeled by a Markovian transition density $p(\mathbf{x}_k|\mathbf{x}_{k-1})$. For the sake of simplicity, we here assume conditions where standard Kalman filtering is applicable, i.e. in the case of well-separated targets, assuming perfect detection, and in absence of false measurements. For notational simplicity let us assume S synchronized sensors produce measurements at the same instants of time t_l, $l = 1, \ldots, k$ denoted by $Z_l = \{\mathbf{z}_l^s\}_{s=1}^S$. The proposed methodology can be directly extended to asynchronous sensors.

The accumulation of the sensor data Z_l up to and including the time t_k, typically the present time, is a time series recursively defined by $\mathcal{Z}^k = \{Z_k, \mathcal{Z}^{k-1}\}$. The time series produced by the measurements of the individual sensor $s = 1, \ldots, S$ only is denoted by $\mathcal{Z}_s^k$. The statistical properties of an individual sensor measurement $\mathbf{z}_l^s$ is described by a probability density function $p(\mathbf{z}_l^s|\mathbf{x}_l)$, also called individual sensor likelihood function, which needs to be known up to a constant factor only: $p(\mathbf{z}_l^s|\mathbf{x}_l) \propto \ell_l^s(x_l; \mathbf{z}_l^s)$.

A Bayesian tracking algorithm is an iterative updating scheme for conditional probability density functions $p(\mathbf{x}_l|\mathcal{Z}^k)$ representing all available knowledge on the state vectors $\mathbf{x}_l$ of the targets of interest at discrete instants of time t_l. Depending on the time t_l at which estimates for the state vectors $\mathbf{x}_l$ are required, the related estimation process is referred to as prediction ($t_l > t_k$), filtering ($t_l = t_k$), or retrodiction ($t_l < t_k$) [9, 12, 13]. We refer to $p(\mathbf{x}_k|\mathcal{Z}^k)$ also as to the 'track' of the target at time t_k. Schematically, the iterative calculation can be illustrated by:

Prediction:

$$p(\mathbf{x}_{k-1}|\mathcal{Z}^{k-1}) \xrightarrow[\text{model}]{\text{dynamics}} p(\mathbf{x}_k|\mathcal{Z}^{k-1}) \qquad (1)$$

Filtering:

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) \xrightarrow[\text{sensor model}]{\text{current data } Z_k} p(\mathbf{x}_k|\mathcal{Z}^k) \tag{2}$$

Retrodiction:

$$p(\mathbf{x}_{l-1}|\mathcal{Z}^k) \xleftarrow[\text{dynamics model}]{\text{filtering output}} p(\mathbf{x}_l|\mathcal{Z}^k). \tag{3}$$

The conditional probability densities $p(\mathbf{x}_i|\mathcal{Z}_s^k)$ denote the local predictions, tracks, and retrodictions based on measurements of sensor $s = 1, \ldots, S$ alone.

1.2 Optimal Track-to-track Fusion

In certain applications, decentralized track-to-track fusion (see e.g. [6–8, 14, 16]) has considerable advantages: Firstly, the communication channels are less overloaded with false tracks suppressed by local data processing. Secondly, we may profit from reduced sensibility to sensor registration errors as local tracking is inherently robust regarding these effects. Thirdly, disturbances of individual sensor sites and their corresponding local processors do not lead to the loss of the total system function.

Disadvantages result from suboptimal performance with respect to reaction time, track quality, lacking profit from redundancy, and the lower data rate for sensor individual tracking affecting track initiation, e.g. Moreover, track-to-track fusion is critical if active and passive sensor data have to be fused (e.g. plots and bearings), as the sensor individual production of tracks may be difficult in non-trivial situations.

We speak of *optimal* track-to-track fusion in a Bayesian sense if the conditional probability density functions $p(\mathbf{x}_k|\mathcal{Z}^k) = p(\mathbf{x}_k|\{\mathcal{Z}_s^k\}_{s=1}^S)$, conditioned by all measurements of all sensors, can be correctly reconstructed from the locally produced tracks $p(\mathbf{x}_k|\mathcal{Z}_s^k)$, obtained by processing the data of the sensors $s = 1, \ldots, S$ individually:

$$\{p(\mathbf{x}_l|\mathcal{Z}_s^l)\}_{s,l=1}^{S,k} \xrightarrow[\text{fusion}]{\text{track-to-track}} p(\mathbf{x}_k|\{\mathcal{Z}_s^k\}_{s=1}^S). \tag{4}$$

It is well-known that this reconstruction is possible if the local tracks produced at all instants of time t_l, $l = 1, \ldots, k$ of all sensors are available in the fusion center [3]. This can easily be seen by considering the joint probability density:

$$p(Z_k^s, \mathbf{x}_k|\mathcal{Z}_s^{k-1}) = p(\mathbf{x}_k|\mathcal{Z}_s^k)\, p(Z_k^s|\mathcal{Z}_s^{k-1}) \tag{5}$$

$$= p(Z_k^s|\mathbf{x}_k)\, p(\mathbf{x}_k|\mathcal{Z}_s^{k-1}), \tag{6}$$

which allows the representation of the local likelihood function $p(Z_k^s|\mathbf{x}_k) \propto \ell_k^s(\mathbf{x}_k; Z_k^s)$ of the sensor s at each time t_k by the results of prediction and filtering up to a factor independent of the target state:

$$\ell_k^s(\mathbf{x}_k; Z_k^s) \propto \frac{p(\mathbf{x}_k|\mathcal{Z}_s^k)}{p(\mathbf{x}_k|\mathcal{Z}_s^{k-1})}. \tag{7}$$

Under the conditions where Kalman filtering is applicable, (7) is a quotient of Gaußians, which can be calculated up to a factor independent of $\mathbf{x}_k$ according to a product formula for Gaußians (see appendix, (94), 2nd version):

$$\ell_k^s(\mathbf{x}_k; \mathbf{z}_k^s) \propto \frac{\mathcal{N}\big(\mathbf{x}_k;\, \mathbf{x}_{k|k}^s,\, \mathbf{P}_{k|k}^s\big)}{\mathcal{N}\big(\mathbf{x}_k;\, \mathbf{x}_{k|k-1}^s,\, \mathbf{P}_{k|k-1}^s\big)} \tag{8}$$

$$\propto \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{x}_{k|k}^{*s},\, \mathbf{P}_{k|k}^{*s}\big) \tag{9}$$

with an expectation vector $\mathbf{x}_{k|k}^{*s}$ and a covariance matrix $\mathbf{P}_{k|k}^{*s}$ given by:

$$\mathbf{P}_{k|k-1}^{*s\ -1} = \mathbf{P}_{k|k}^{s\ -1} - \mathbf{P}_{k|k-1}^{s\ -1} \tag{10}$$

$$\mathbf{x}_{k|k-1}^{*s} = \mathbf{P}_{k|k}^{*s}\big(\mathbf{P}_{k|k}^{s\ -1}\mathbf{x}_{k|k}^s - \mathbf{P}_{k|k-1}^{s\ -1}\mathbf{x}_{k|k-1}^s\big). \tag{11}$$

This representation of the local likelihood function $\ell_k^s(\mathbf{x}_k; \mathbf{z}_k^s)$ has also been called a 'tracklet' according to G. Frenkel (see e.g. [10]).

Along this line of argumentation and with a repeated use of the product formula, the joint likelihood function of the measurements of S independent sensors producing data at the same time t_k can be written as:

$$\ell_k(\mathbf{x}_k; Z_k) = \prod_{s=1}^S \ell_k^s(\mathbf{x}_k; \mathbf{z}_k^s) \tag{12}$$

$$\propto \prod_{s=1}^S \mathcal{N}\big(\mathbf{x}_{k|k-1}^{*s};\, \mathbf{x}_k,\, \mathbf{P}_{k|k-1}^{*s}\big) \tag{13}$$

$$\propto \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{x}_{k|k-1}^*,\, \mathbf{P}_{k|k-1}^*\big) \tag{14}$$

with $\mathbf{x}_{k|k}^*$ and $\mathbf{P}_{k|k}^*$ given by

$$\mathbf{P}_{k|k-1}^{*\ -1} = \sum_{s=1}^S \big(\mathbf{P}_{k|k}^{s\ -1} - \mathbf{P}_{k|k-1}^{s\ -1}\big) \tag{15}$$

$$\mathbf{x}_{k|k-1}^* = \mathbf{P}_{k|k-1}^* \sum_{s=1}^S \big(\mathbf{P}_{k|k}^{s\ -1}\mathbf{x}_{k|k}^s - \mathbf{P}_{k|k-1}^{s\ -1}\mathbf{x}_{k|k-1}^s\big). \tag{16}$$

With this representation of the joint likelihood function $\ell_k(\mathbf{x}_k; Z_k)$ built up by using the local sensor tracks, formulae for optimal track-to-track fusion directly result from Bayes' rule and the product formula (94):

$$p(\mathbf{x}_k|\mathcal{Z}^k) = \frac{\ell_k(\mathbf{x}_k; Z_k)\, p(\mathbf{x}_k|\mathcal{Z}^{k-1})}{\int d\mathbf{x}_k\, \ell_k(\mathbf{x}_k; Z_k)\, p(\mathbf{x}_k|\mathcal{Z}^{k-1})} \tag{17}$$

$$= \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{x}_{k|k},\, \mathbf{P}_{k|k}\big) \tag{18}$$

with $\mathbf{x}_{k|k}$ and $\mathbf{P}_{k|k}$ given by

$$\mathbf{P}_{k|k}^{-1} = \mathbf{P}_{k|k-1}^{-1} + \mathbf{P}_{k|k-1}^{*\ -1} \tag{19}$$

$$\mathbf{x}_{k|k} = \mathbf{P}_{k|k}^{-1}\big(\mathbf{P}_{k|k-1}^{-1}\mathbf{x}_{k|k-1} + \mathbf{P}_{k|k-1}^{*\ -1}\mathbf{x}_{k|k-1}^*\big). \tag{20}$$

1.3 Arbitrary Communication Rate Fusion

In many applications, the full communication of all local sensor tracks produced at all scan times of the individual

sensors and fusing them as previously sketched is impractical. We therefore propose a distributed Kalman-type processing scheme which provides optimal track-to-track fusion results at an *arbitrarily chosen* instant of time by communicating the local sensor tracks referring to this time only and fusing them in the fusion center. Then, the obtained global state is optimal in sense of the expected estimation error and equivalent to an imaginary Kalman filter receiving all available measurements from all sensors every time.

More concretely speaking, we derive fusion formulas to correctly reconstruct the probability densities $p(\mathbf{x}_l|\mathcal{Z}^k)$ for an arbitrary instant of time t_l from local state estimates $\mathbf{x}^s_{l|k}$, $s = 1, \ldots, S$, which are calculated by processing data from sensor s only:

$$\{\mathbf{x}^s_{l|k}\}^S_{s=1} \xrightarrow[\text{fusion}]{\text{track-to-track}} p(\mathbf{x}_l|\{\mathcal{Z}^k_s\}^S_{s=1}). \tag{21}$$

For the sake of notational simplicity, let all sensors be equally aligned and synchronized with the same data update rate. These assumptions, however, are not essential and can well be relaxed. We furthermore assume that the measurement error covariance matrices of all individual sensors are known for the local and processors (e.g. the sensor-specific standard deviations of range and azimuth measurements σ^s_r, σ^s_φ). The detection probability is assumed to be one. If missing detections have to be taken into account, detection and non-detection flags have to be communicated with the vectors $\mathbf{x}^s_{l|k}$ as well (i.e. a few additional bits).

In calculating the local tracks $\mathbf{x}^s_{l|k}$ the filtering update with the local sensor data is identical with the standard Kalman filtering update step, while the prediction and the retrodiction steps are to be modified in order to be able to reconstruct $p(\mathbf{x}_l|\{\mathcal{Z}^k_s\}^S_{s=1})$ after fusing them according to (30). This subject matter is exactly the same as in [1]. In contrast to it, the normalizing functions we use in this paper (e.g. at (56) and (85)) do not depend on the state variable. This can only be achieved by updating the local tracks $\mathbf{x}^s_{l|k}$ using the global covariance $\overline{\mathbf{P}}_{l|k}$ at each sensor node. In consequence, these are not identical with the results of local Kalman filtering, thus local optimality suffers for global optimality. Details and reasoning are given in section 3 and 5. In case of these functions being dependent on the state, it is necessary to calculate them in order to reconstruct the correct global track $\mathbf{x}_{l|k}$. By a closer look at the equations in [1, section III-C] one can see that the measurements of several sensors are required for this purpose. Therefore, we did some further investigations in order to overcome this problem. Note that the special case of two sensors (i.e. $S = 2$), which is discussed in [1, section III-A] and evaluated in [1, section VI] works fine without exchanging data messages between the sensors. This becomes clear by (57) and (89) in the sections below.

The proposed distributed Kalman-type processing scheme essentially makes use of the fact that the sensor measurements do *not* enter into the update equation for the estimation error covariance matrices. This means in particular that the covariance matrices of *all* sensors can be calculated at each individual sensor site without any further need of communication (given the relevant parameters of all sensors are known at each sensor site). A consequence of this fact is, however, that the proposed optimal processing scheme cannot directly be generalized to IMM- or PDA-type tracking, where the covariance matrices are explicitly data-dependent.

2 Statement of the Problem

Let us *consider* probability density functions $p(\mathbf{x}_l|\mathcal{Z}^k)$ that are proportional to a *product* of S Gaussians:

$$p(\mathbf{x}_l|\mathcal{Z}^k) = c_{l|k} \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_l;\, \mathbf{x}^s_{l|k},\, \mathbf{P}^s_{l|k}\big), \tag{22}$$

where $\mathcal{Z}^k = \{\mathcal{Z}^k_s\}^S_{s=1}$ is the set of all measurements from all sensors. The normalizing constant $c_{l|k}$ directly results from a repeated use of the product formula for Gaussians (94):

$$c^{-1}_{l|k} = \prod_{s=1}^{S} \int d\mathbf{x}_l\; \mathcal{N}\big(\mathbf{x}_l;\, \mathbf{x}^s_{l|k},\, \mathbf{P}^s_{l|k}\big) \tag{23}$$

$$= \prod_{s=1}^{S-1} \mathcal{N}\big(\mathbf{x}^{*s}_{l|k};\, \mathbf{x}^{s+1}_{l|k},\, \mathbf{P}^{*s}_{l|k} + \mathbf{P}^{s+1}_{l|k}\big), \tag{24}$$

where the parameters $\mathbf{x}^{*s}_{l|k}$ and $\mathbf{P}^{*s}_{l|k}$ are given by

$$\mathbf{P}^{*s\ -1}_{l|k} = \sum_{i=1}^{s-1} \mathbf{P}^{i\ -1}_{l|k} \tag{25}$$

$$\mathbf{x}^{*s}_{l|k} = \mathbf{P}^{*s\ -1}_{l|k} \sum_{i=1}^{s-1} \mathbf{P}^{i\ -1}_{l|k} \mathbf{x}^i_{l|k}. \tag{26}$$

In the sequel, as well as in most applications, it is unnecessary to calculate the normalization constant $c_{l|k}$ explicitly.

By virtue of the same product formula, probability densities defined by Gaussian products as in (22) can be transformed into a single Gaussian:

$$p(\mathbf{x}_l|\mathcal{Z}^k) = c_{l|k} \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_l;\, \mathbf{x}^s_{l|k},\, \mathbf{P}^s_{l|k}\big) \tag{27}$$

$$= \mathcal{N}\big(\mathbf{x}_l;\, \mathbf{x}_{l|k},\, \mathbf{P}_{l|k}\big), \tag{28}$$

with an expectation vector $\mathbf{x}_{l|k}$ and a covariance matrix $\mathbf{P}_{l|k}$ obtained by 'fusing' $\mathbf{x}^s_{l|k}$ and $\mathbf{P}^s_{l|k}$, $s = 1, \ldots, S$ according to the formulae:

$$\mathbf{P}^{-1}_{l|k} = \sum_{s=1}^{S} \mathbf{P}^{s\ -1}_{l|k} \tag{29}$$

$$\mathbf{x}_{l|k} = \mathbf{P}_{l|k} \Big(\sum_{s=1}^{S} \mathbf{P}^{s\ -1}_{l|k} \mathbf{x}^s_{l|k} \Big). \tag{30}$$

'Convex combinations' of this type are fundamental in almost all data fusion applications (see e.g. [14, Chapter 12]).

However, the challenge is to keep the product representation of (22) during the prediction and retrodiction step of the locals data processing. These involve the Markovian transition density $p(\mathbf{x}_k|\mathbf{x}_{k-1})$, which describes the same stochastic process for every local track. This leads cross correlations between the tracks which prevents a product representation as in (22). Therefore, we have to alter its utilization such that it is used in a more global way. As previously stated, under conditions where Kalman filtering is appropriate for tracking, the covariance matrices $\mathbf{P}^s_{l|k}$ can be calculated locally for all sensors without exchanging sensor data, provided the measurement error covariance matrices of each individual sensor are known. This enables us to let each node work with the *global covariance* $\tilde{\mathbf{P}}_{l|k}$.

The proof of the proposed methodology is organized as follows. As we assume the sensors to be synchronized, it suffices to show the product representation is valid after every update step. To this end, we state an induction argument, which starts with the initialization of the distributed Kalman filters. This has to be done such that a product representation of $p(\mathbf{x}_1|\mathcal{Z}^1)$ is equivalent to a single Gaussian representation. For the latter, one might choose the *One-Point-Initialization* (*OPI*) [2], which is given by an initial ignorance covariance $\mathbf{P}_{1|0}$ and an initial measurement. While $\mathbf{P}_{1|0}$ is usually part of the modeling assumption, we take e.g. $\mathbf{z}^1_1$ in order to start the initialization of $\mathbf{x}_1$. All other measurements $\mathbf{z}^2_1, \ldots, \mathbf{z}^S_1$ enter into the initial density according to the standard filtering equations:

$$\mathbf{x}_{1|1} = \mathbf{P}_{1|1} \sum_{s=1}^{S} \tilde{\mathbf{H}}^{s\top} \tilde{\mathbf{R}}^{s\,-1} \tilde{\mathbf{z}}^s_1 \tag{31}$$

$$\mathbf{P}_{1|1} = \left(\sum_{s=1}^{S} \tilde{\mathbf{H}}^{s\top} \tilde{\mathbf{R}}^{s\,-1} \tilde{\mathbf{H}}^s \right)^{-1} \tag{32}$$

$$\tilde{\mathbf{R}}^s = \begin{cases} \left(\mathbf{H}^{1\top} \mathbf{R}^{1\,-1} \mathbf{H}^1 + \mathbf{P}^{-1}_{1|0} \right)^{-1}, & \text{if } s = 1 \\ \mathbf{R}^s, & \text{else,} \end{cases} \tag{33}$$

$$\tilde{\mathbf{H}}^s = \begin{cases} \mathbf{1}_{\mathbf{x}}, & \text{if } s = 1 \\ \mathbf{H}^s, & \text{else,} \end{cases} \tag{34}$$

$$\tilde{\mathbf{z}}^s_1 = \begin{cases} \mathbf{H}^{1\top} \mathbf{z}^1_1, & \text{if } s = 1 \\ \mathbf{z}^s_1, & \text{else,} \end{cases} \tag{35}$$

which finally gives us the optimal OPI under consideration of all first measurements $\mathcal{Z}^1$:

$$p(\mathbf{x}_1|\mathcal{Z}^1) = \mathcal{N}\big(\mathbf{x}_1;\, \mathbf{x}_{1|1},\, \mathbf{P}_{1|1}\big) \tag{36}$$

It can easily be seen that this Gaussian density is proportional to the following product representation

$$p(\mathbf{x}_1|\mathcal{Z}^1) \propto \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_1;\, \mathbf{x}^s_{1|1},\, \mathbf{P}^s_{1|1}\big), \tag{37}$$

where the distributed parameters $\mathbf{x}^s_{1|1}$ and $\mathbf{P}^s_{1|1}$ are given by

$$\mathbf{x}^s_{1|1} = \mathbf{P}_{1|1} \tilde{\mathbf{H}}^{s\top} \tilde{\mathbf{R}}^{s\,-1} \tilde{\mathbf{z}}^s_1 \tag{38}$$

$$\mathbf{P}^s_{1|1} = S \mathbf{P}_{1|1}. \tag{39}$$

The following section now presents the distributed prediction step which is the tricky part as we described above. Then, in section 4 the processing of measurements in the distributed scheme is described.

3 Distributed Track Prediction

Due to the induction assumption we start with a product representation at time t_{k-1} where $k - 1 \geq 0$:

$$p(\mathbf{x}_{k-1}|\mathcal{Z}^{k-1}) \propto \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_{k-1};\, \mathbf{x}^s_{k-1|k-1},\, \mathbf{P}^s_{k-1|k-1}\big) \tag{40}$$

This density can be rewritten in the following way.

$$p(\mathbf{x}_{k-1}|\mathcal{Z}^{k-1}) \propto \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_{k-1};\, \mathbf{x}^s_{k-1|k-1},\, \mathbf{P}^s_{k-1|k-1}\big) \tag{41}$$

$$= \mathcal{N}\big(\mathbf{x}_{k-1};\, \frac{1}{S} \sum_{s=1}^{S} \overline{\mathbf{x}}^s_{k-1|k-1},\, \overline{\mathbf{P}}_{k-1|k-1}\big), \tag{42}$$

$$= \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_{k-1};\, \overline{\mathbf{x}}^s_{k-1|k-1},\, S\overline{\mathbf{P}}_{k-1|k-1}\big), \tag{43}$$

where the parameters $\overline{\mathbf{x}}^s_{k-1|k-1}$ and the global filtering covariance $\overline{\mathbf{P}}_{k-1|k-1}$ are given by:

$$\overline{\mathbf{x}}^s_{k-1|k-1} = S\overline{\mathbf{P}}_{k-1|k-1} \mathbf{P}^{s\,-1}_{k-1|k-1} \mathbf{x}^s_{k-1|k-1} \tag{44}$$

$$\overline{\mathbf{P}}_{k-1|k-1} = \left(\sum_{s=1}^{S} \mathbf{P}^{s\,-1}_{k-1|k-1} \right)^{-1}. \tag{45}$$

If we now use the fact that

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{F}_{k|k-1}\mathbf{x}_{k-1},\, \mathbf{D}_{k|k-1}\big) \tag{46}$$

$$\propto \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{F}_{k|k-1}\mathbf{x}_{k-1},\, S\mathbf{D}_{k|k-1}\big)^S, \tag{47}$$

this enables us to express the prediction density as

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

$$\propto \int d\mathbf{x}_{k-1} \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{F}_{k|k-1}\mathbf{x}_{k-1},\, S\mathbf{D}_{k|k-1}\big) \cdot \mathcal{N}\big(\mathbf{x}_{k-1};\, \overline{\mathbf{x}}^s_{k-1|k-1},\, S\overline{\mathbf{P}}_{k-1|k-1}\big). \tag{48}$$

A repeated use of the product formula for Gaussians now yields

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) \propto \int d\mathbf{x}_{k-1} \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_k;\, \tilde{\mathbf{x}}^s_{k|k-1},\, \tilde{\mathbf{P}}_{k|k-1}\big) \cdot \mathcal{N}\big(\mathbf{x}_{k-1};\, \tilde{\mathbf{W}}_{k-1}\mathbf{x}_k + \tilde{\mathbf{z}}^s_{k-1},\, \tilde{\mathbf{R}}_{k-1}\big), \tag{49}$$

where the following abbreviations were used:

$$\tilde{\mathbf{x}}^s_{k|k-1} = \mathbf{F}_{k|k-1}\overline{\mathbf{x}}^s_{k-1|k-1} \tag{50}$$

$$\tilde{\mathbf{P}}_{k|k-1} = S\left(\mathbf{F}_{k|k-1}\overline{\mathbf{P}}_{k-1|k-1}\mathbf{F}^T_{k|k-1} + \mathbf{D}_{k|k-1}\right) \tag{51}$$

$$\tilde{\mathbf{z}}^s_{k-1} = \overline{\mathbf{x}}^s_{k-1|k-1} - \tilde{\mathbf{W}}_{k-1}\tilde{\mathbf{x}}^s_{k|k-1} \tag{52}$$

$$\tilde{\mathbf{W}}_{k-1} = S\overline{\mathbf{P}}_{k-1|k-1}\mathbf{F}^T_{k|k-1}\tilde{\mathbf{R}}^{-1}_{k-1}. \tag{53}$$

$$\tilde{\mathbf{R}}_{k-1} = S\overline{\mathbf{P}}_{k-1|k-1} - \tilde{\mathbf{W}}_{k-1}\tilde{\mathbf{P}}_{k|k-1}\tilde{\mathbf{W}}^T_{k-1} \tag{54}$$

A bit literally speaking we might say that $\overline{\mathbf{P}}_{k-1|k-1}$ is the global filtering covariance, while $\tilde{\mathbf{P}}_{k|k-1}$ may be seen as the *global prediction covariance*. This equation can be written as

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) = \ell_{k|k-1}\prod_{s=1}^{S}\mathcal{N}\big(\mathbf{x}_k;\,\tilde{\mathbf{x}}^s_{k|k-1},\,\tilde{\mathbf{P}}_{k|k-1}\big) \tag{55}$$

by introducing the function $\ell_{k|k-1}$ defined by

$$\ell_{k|k-1} = \int d\mathbf{x}_{k-1}\prod_{s=1}^{S}\mathcal{N}\big(\mathbf{x}_{k-1} - \tilde{\mathbf{W}}_{k-1}\mathbf{x}_k;\,\tilde{\mathbf{z}}^s_{k-1},\,\tilde{\mathbf{R}}_{k-1}\big) \tag{56}$$

Applying the formula (94) along the product within $l_{k|k-1}$ yields

$$\ell_{k|k-1} = \prod_{s=1}^{S-1}\mathcal{N}\big(\tilde{\mathbf{z}}^{s+1}_{k-1};\,\frac{1}{s}\sum_{i=1}^{s}\tilde{\mathbf{z}}^i_{k-1},\,\left(1+\frac{1}{s}\right)\tilde{\mathbf{R}}_{k-1}\big) \cdot \underbrace{\int d\mathbf{x}_{k-1}\,\mathcal{N}\big(\mathbf{x}_{k-1} - \tilde{\mathbf{W}}_{k-1}\mathbf{x}_k;\,\frac{1}{S}\sum_{i=1}^{S}\tilde{\mathbf{z}}^i_{k-1},\,\frac{1}{S}\tilde{\mathbf{R}}_{k-1}\big)}_{=1,\text{ independent from }\mathbf{x}_k!} \tag{57}$$

Thus, the function $\ell_{k|k-1}$ is independent from $\mathbf{x}_k$ and we obtain the desired product representation:

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) \propto \prod_{s=1}^{S}\mathcal{N}\big(\mathbf{x}_k;\,\tilde{\mathbf{x}}^s_{k|k-1},\,\tilde{\mathbf{P}}_{k|k-1}\big) \tag{58}$$

4 Distributed Track Filtering

In this section, we estimate the posterior density $p(\mathbf{x}_k|\mathcal{Z}^k)$ which additionally involves the set of measurements $Z_k = \{\mathbf{z}^s_k\}_{s=1,\ldots,S}$ from all sensors at time t_k. To this end, it is important to notice that the sensors act independently from each other:

$$p(Z_k|\mathbf{x}_k) \propto \prod_{s=1}^{S} p(\mathbf{z}^s_k|x_k), \tag{59}$$

where $p(\mathbf{z}^s_k|\mathbf{x}_k) = \mathcal{N}\big(\mathbf{z}^s_k;\,\mathbf{H}^s_k\mathbf{x}_k,\,\mathbf{R}^s_k\big)$ is the likelihood function of sensor s. By a use of the Bayes' rule, we obtain

$$p(\mathbf{x}_k|\mathcal{Z}^k) = \frac{p(Z_k|\mathbf{x}_k)\,p(\mathbf{x}_k|Z^{k-1})}{\int \mathrm{d}\mathbf{x}_k\,p(Z_k|\mathbf{x}_k)\,p(\mathbf{x}_k|Z^{k-1})}. \tag{60}$$

We can now fill in the prediction (58) and the likelihood function (59) and then have

$$p(\mathbf{x}_k|\mathcal{Z}^k) \propto \prod_{s=1}^{S}\mathcal{N}\big(\mathbf{z}^s_k;\,\mathbf{H}^s_k\mathbf{x}_k,\,\mathbf{R}^s_k\big)\cdot \mathcal{N}\big(\mathbf{x}_k;\,\tilde{\mathbf{x}}^s_{k|k-1},\,\tilde{\mathbf{P}}_{k|k-1}\big). \tag{61}$$

A final use of the product formula now yields the product representation for the filtering update:

$$p(\mathbf{x}_k|\mathcal{Z}^k) \propto \prod_{s=1}^{S}\mathcal{N}\big(\mathbf{x}_k;\,\mathbf{x}^s_{k|k},\,\mathbf{P}^s_{k|k}\big), \tag{62}$$

where the following abbreviations in analogy to the Kalman equations were used:

$$\mathbf{x}^s_{k|k} = \tilde{\mathbf{x}}^s_{k|k-1} + \mathbf{W}^s_{k|k-1}\left(\mathbf{z}^s_k - \mathbf{H}^s_k\tilde{\mathbf{x}}^s_{k|k-1}\right) \tag{63}$$

$$\mathbf{W}^s_{k|k-1} = \tilde{\mathbf{P}}_{k|k-1}\mathbf{H}^{s\,T}_k\mathbf{S}^{s\,-1}_{k|k-1} \tag{64}$$

$$\mathbf{S}^s_{k|k-1} = \mathbf{H}^s_k\tilde{\mathbf{P}}_{k|k-1}\mathbf{H}^{s\,T}_k + \mathbf{R}^s_k \tag{65}$$

$$\mathbf{P}^s_{k|k} = \tilde{\mathbf{P}}_{k|k-1} - \mathbf{W}^s_{k|k-1}\mathbf{S}^s_{k|k-1}\mathbf{W}^{s\,T}_{k|k-1}. \tag{66}$$

5 Distributed Retrodiction Update

Essentially the same considerations are valid for distributed retrodiction, i.e. if for the retrodicted probability density $p(\mathbf{x}_l|\mathcal{Z}^k)$ a Gaussian product representation exists,

$$p(\mathbf{x}_l|\mathcal{Z}^k) \propto \prod_{s=1}^{S}\mathcal{N}\big(\mathbf{x}_l;\,\mathbf{x}^s_{l|k},\,\mathbf{P}^s_{l|k}\big), \quad l < k, \tag{67}$$

where the retrodicted local state estimates $\mathbf{x}^s_{l|k}$ can be calculated by processing the measurements of sensor s only. To this end, we assume such a representation for step $l+1$, where $l+1 \le k$ and derive it for step l.

At first, standard probability reasoning yields:

$$p(\mathbf{x}_l|\mathcal{Z}^k) = \int d\mathbf{x}_{l+1}\,p(\mathbf{x}_l,\mathbf{x}_{l+1}|\mathcal{Z}^k) \tag{68}$$

$$= \int d\mathbf{x}_{l+1}\,p(\mathbf{x}_l|\mathbf{x}_{l+1},\mathcal{Z}^k)\,p(\mathbf{x}_{l+1}|\mathcal{Z}^k) \tag{69}$$

For the integration kernel $p(\mathbf{x}_l|\mathbf{x}_{l+1},\mathcal{Z}^k)$ we obtain after applying Bayes' rule:

$$p(\mathbf{x}_l|\mathbf{x}_{l+1},\mathcal{Z}^k) = p(\mathbf{x}_l|\mathbf{x}_{l+1},\mathcal{Z}^l) \tag{70}$$

$$= \frac{p(\mathbf{x}_{l+1}|\mathbf{x}_l)\,p(\mathbf{x}_l|\mathcal{Z}^l)}{\int d\mathbf{x}_l\,p(\mathbf{x}_{l+1}|\mathbf{x}_l)\,p(\mathbf{x}_l|\mathcal{Z}^l)} \tag{71}$$

As in section 3, we rewrite the density $p(\mathbf{x}_l|\mathcal{Z}^l)$. Because of section 4, a product representation for may be assumed for

it. Therefore, by the product formula it holds

$$p(\mathbf{x}_l|\mathcal{Z}^l) \propto \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_l;\, \mathbf{x}^s_{l|l},\, \mathbf{P}^s_{l|l}\big) \tag{72}$$

$$\propto \mathcal{N}\big(\mathbf{x}_l;\, \frac{1}{S}\sum_{s=1}^{S} \overline{\mathbf{x}}^s_{l|l},\, \overline{\mathbf{P}}_{l|l}\big) \tag{73}$$

$$= \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_l;\, \overline{\mathbf{x}}^s_{l|l},\, S\overline{\mathbf{P}}_{l|l}\big), \tag{74}$$

where we used the following abbreviations:

$$\overline{\mathbf{x}}^s_{l|l} = S\overline{\mathbf{P}}_{l|l}\mathbf{P}^{s\,-1}_{l|l}\mathbf{x}^s_{l|l} \tag{75}$$

$$\overline{\mathbf{P}}_{l|l} = \left(\sum_{s=1}^{S} \mathbf{P}^{s\,-1}_{l|l}\right)^{-1}. \tag{76}$$

In the same way, we modify the product representation for time t_{l+1}, which then looks like

$$p(\mathbf{x}_{l+1}|\mathcal{Z}^k) \propto \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_{l+1};\, \overline{\mathbf{x}}^s_{l+1|k},\, S\overline{\mathbf{P}}_{l+1|k}\big). \tag{77}$$

Now, we insert (76) into (71). This enables us to apply the product formula again:

$$p(\mathbf{x}_l|\mathbf{x}_{l+1}, \mathcal{Z}^l) \propto \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_l;\, \overline{\mathbf{x}}^s_{l|l},\, S\overline{\mathbf{P}}_{l|l}\big)$$

$$\cdot\, \mathcal{N}\big(\mathbf{x}_{l+1};\, \mathbf{F}_{l+1|l}\mathbf{x}_l,\, S\mathbf{D}_{l+1|l}\big) \tag{78}$$

$$\propto \mathcal{N}\big(\mathbf{x}_l;\, \mathbf{h}^s_{l|l}(\mathbf{x}_{l+1}),\, \mathbf{R}_{l|l}\big) \tag{79}$$

by using the abbreviations:

$$\mathbf{h}^s_{l|l}(\mathbf{x}_{l+1}) = \overline{\mathbf{x}}^s_{l|l} + \overline{\mathbf{W}}_{l|l+1}\left(\mathbf{x}_{l+1} - \tilde{\mathbf{x}}^s_{l+1|l}\right) \tag{80}$$

$$\mathbf{R}_{l|l} = S\left(\overline{\mathbf{P}}_{l|l} - \overline{\mathbf{W}}_{l|l+1}\overline{\mathbf{P}}_{l+1|l}\overline{\mathbf{W}}^{\top}_{l|l+1}\right) \tag{81}$$

$$\overline{\mathbf{W}}_{l|l+1} = S\overline{\mathbf{P}}_{l|l}\mathbf{F}^{\top}_{l+1|l}\overline{\mathbf{P}}_{l+1|l} \tag{82}$$

Inserting $p(\mathbf{x}_l|\mathbf{x}_{l+1}, \mathcal{Z}^k)$ into (69) and applying the product formula once again yields

$$p(\mathbf{x}_l|\mathcal{Z}^k) \propto\ \ell_{l|l+1} \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_l;\, \tilde{\mathbf{x}}^s_{l|k},\, \tilde{\mathbf{P}}_{l|k}\big), \tag{83}$$

where parameters $\tilde{\mathbf{x}}^s_{l|k}$, $\tilde{\mathbf{P}}_{l|k}$ are given by the slightly varied Rauch-Tung-Striebel update equations [11] for Kalman retrodiction

$$\tilde{\mathbf{x}}^s_{l|k} = \overline{\mathbf{x}}^s_{l|l} + \overline{\mathbf{W}}_{l|l+1}(\mathbf{x}^s_{l+1|k} - \tilde{\mathbf{x}}^s_{l+1|l}) \tag{84}$$

$$\tilde{\mathbf{P}}_{l|k} = S\left(\overline{\mathbf{P}}_{l|l} + \overline{\mathbf{W}}_{l|l+1}(\overline{\mathbf{P}}_{l+1|k} - \overline{\mathbf{P}}_{l+1|l})\overline{\mathbf{W}}^{s\top}_{l|l+1}\right)$$

and a normalizing function $\ell_{l|k}$, which is defined by:

$$\ell_{l|k}(\mathbf{x}_l) = \int \mathrm{d}\mathbf{x}_{l+1}$$

$$\prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_{l+1} - \mathbf{V}_{l|l+1}\mathbf{x}_l;\, \mathbf{y}^s_{l|l+1},\, \mathbf{Y}_{l|l+1}\big) \tag{85}$$

with the abbreviations:

$$\mathbf{y}^s_{l|l+1} = \mathbf{x}^s_{l+1|k} - \mathbf{V}_{l|l+1}\tilde{\mathbf{x}}^s_{l|k} \tag{86}$$

$$\mathbf{Y}_{l|l+1} = S\left(\overline{\mathbf{P}}_{l+1|k} - \mathbf{V}_{l|l+1}\tilde{\mathbf{P}}_{l|k}\mathbf{V}^{\top}_{l|l+1}\right) \tag{87}$$

$$\mathbf{V}_{l|l+1} = S\overline{\mathbf{P}}_{l+1|k}\overline{\mathbf{W}}^{\top}_{l|l+1}\tilde{\mathbf{P}}^{-1}_{l|k}. \tag{88}$$

If we use the product formula along the terms in $\ell_{l|k}$, we obtain

$$\ell_{l|k} = \prod_{s=1}^{S-1} \mathcal{N}\big(\mathbf{y}^{s+1}_{l|l+1};\, \frac{1}{s}\sum_{i=1}^{s} \mathbf{y}^i_{l|l+1},\, (1+\frac{1}{s})\mathbf{Y}_{l|l+1}\big)$$

$$\int \mathrm{d}\mathbf{x}_{l+1}\, \mathcal{N}\big(\mathbf{x}_{l+1} - \mathbf{V}_{l|l+1}\mathbf{x}_l;\, \frac{1}{S}\sum_{i=1}^{S} \mathbf{y}^i_{l|l+1},\, \frac{1}{S}\mathbf{Y}_{l|l+1}\big). \tag{89}$$

As in section 3, the integration term becomes trivial, as it is over a single Gaussian distribution. Therefore, the normalizing function $\ell_{l|k}$ does not depend on $\mathbf{x}_l$ and we have

$$p(\mathbf{x}_l|\mathcal{Z}^k) \propto \prod_{s=1}^{S} \mathcal{N}\big(\mathbf{x}_l;\, \tilde{\mathbf{x}}^s_{l|k},\, \tilde{\mathbf{P}}_{l|k}\big). \tag{90}$$

6 Evaluation

This section presents simulation results, which verify the equivalence of the presented distributed scheme to a central Kalman filter receiving all measurements. Note that if all involved sensors are of the same type, i.e. $\mathbf{H}^1 = \ldots = \mathbf{H}^S =: \mathbf{H}$ and $\mathbf{R}^1 = \ldots = \mathbf{R}^S =: \mathbf{R}$, the distributed prediction and retrodiction is almost identical with the central scheme:

$$\overline{\mathbf{x}}^s_{k-1|k-1} = \mathbf{x}^s_{k-1|k-1} \tag{91}$$

$$\overline{\mathbf{P}}_{k-1|k-1} = \frac{1}{S}\mathbf{P}^s_{k-1|k-1}. \tag{92}$$

Thus, both schemes are identical except for the distributed nodes use a '*relaxed*' evolution model given by the transition density

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{F}_{k|k-1}\mathbf{x}_{k-1},\, S\mathbf{D}_{k|k-1}\big). \tag{93}$$

Therefore, we chose a simulation scenario, which utilizes ten virtual sensors observing the same object moving around. The quality of the sensors varies with the simulated normal distributed error $\mathbf{R}^s = \lambda \cdot \mathbf{1}_z$, where $\lambda = 50, 100, \ldots, 500$ and $\mathbf{1}_z$ the identity matrix in the dimension of the position measurements $\mathbf{z}_k$. Each sensor is a fusion node, i.e. processes locally its own measurements in the way described above. As the presented scheme for distributed Kalman filtering does not use any approximations, we expect the results to match an optimal Kalman filter, which receives all data immediately. Therefore, we compare such an optimal Kalman filter to the presented distributed scheme at each step. In figure 1, the distance between the tracks of both algorithms is shown. An increase of the distance can be observed. This is due to rounding errors in the numerical function for inverting matrices of the

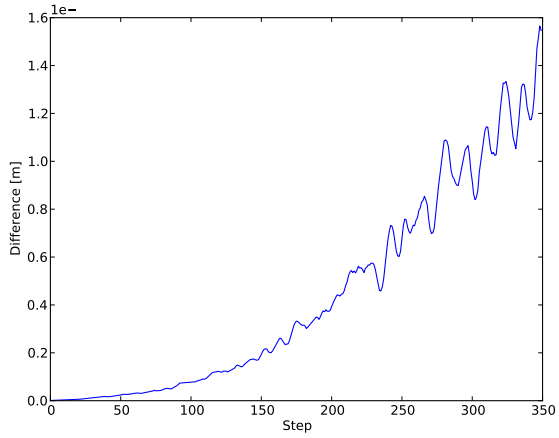

Figure 1: Distance of the distributed track to an optimal Kalman track

simulation environment. As such an error is additive, the distance increases over time. Of course, it can be neglected as the accumulated error in the distance is about $1.6 \cdot 10^{-8} m$ in the maximum. Therefore, the tracks of both algorithms go together. The mentioned rounding error has the same effect

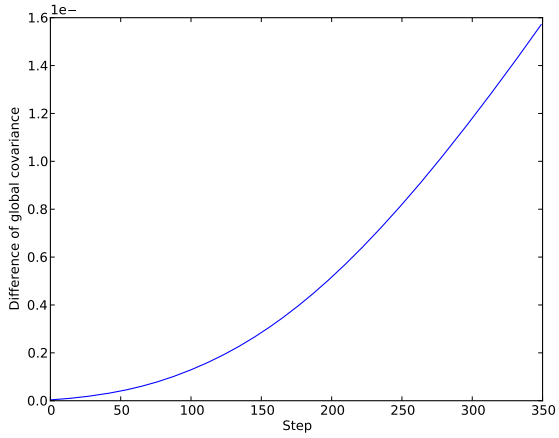

Figure 2: Distance of the covariances in the matrix-2-norm for standard Kalman filter and distributed Kalman filter

on the covariance matrices. As both algorithms are equivalent, we also expect the corresponding global covariance matrices to be the same at each step. In figure 2 the difference between both in the matrix-2-norm is shown. As one can see, the curve is quite similar to the difference in both tracks (see figure 1). The maximum difference is about $1.6 \cdot 10^{-9}$, therefore one might say the error covariance matrices are identical for both algorithms in terms of numerical precision. However, this reasoning is confirmed by a use of the *Joseph form*, see (99) in the appendix, in both, the optimal filter and the distributed scheme. This form is less prone to numerical errors, which leads to a difference in track and covariance of both algorithms at a constantly low level. This

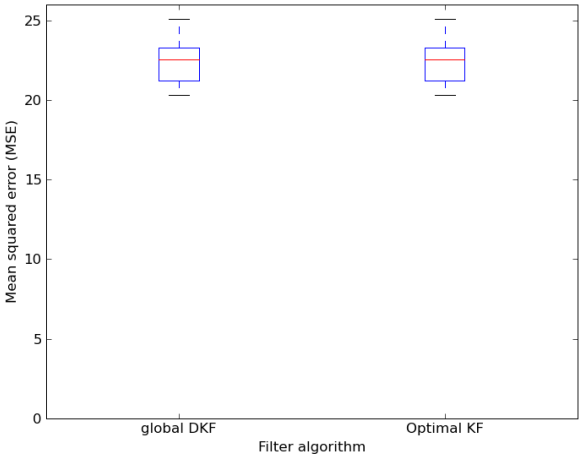

Figure 3: Mean Squared Error (*MSE*) of both algorithms for different configuration runs

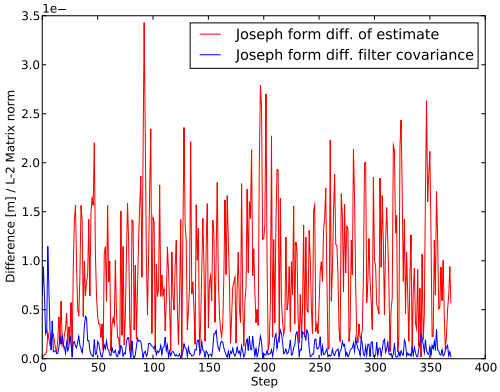

Figure 4: Difference of covariance and track when using the Joseph form

is depicted in figure 4. In order to show this is true for different configurations, we simulated a series of different covariances $\mathbf{D}_{k|k-1}$ with multiple Monte-Carlo runs each. Figure 3 shows the Mean-Squared-Error (*MSE*) of both algorithms in these scenarios. It is obvious that the MSEs coincide.

Appendix

Product formula

For matrices of suitable dimensions the following formula for products of Gaussians holds:

$$\begin{aligned} &\mathcal{N}(\mathbf{z};\, \mathbf{Hx},\, \mathbf{R})\, \mathcal{N}(\mathbf{x};\, \mathbf{y},\, \mathbf{P}) = \\ &\mathcal{N}(\mathbf{z};\, \mathbf{Hy},\, \mathbf{S}) \begin{cases} \mathcal{N}(\mathbf{x};\, \mathbf{y} + \mathbf{W}\boldsymbol{\nu},\, \mathbf{P} - \mathbf{WSW}^\top) \\ \mathcal{N}(\mathbf{x};\, \mathbf{Q}(\mathbf{P}^{-1}\mathbf{y} + \mathbf{H}^\top \mathbf{R}^{-1}\mathbf{z}),\, \mathbf{Q}) \end{cases} \end{aligned} \tag{94}$$

with the following abbreviations:

$$\boldsymbol{\nu} = \mathbf{z} - \mathbf{H}\mathbf{y} \tag{95}$$

$$\mathbf{S} = \mathbf{H}\mathbf{P}\mathbf{H}^\top + \mathbf{R} \tag{96}$$

$$\mathbf{W} = \mathbf{P}\mathbf{H}^\top\mathbf{S}^{-1} \tag{97}$$

$$\mathbf{Q}^{-1} = \mathbf{P}^{-1} + \mathbf{H}^\top\mathbf{R}^{-1}\mathbf{H}. \tag{98}$$

Sketch of a proof: Interpret $\mathcal{N}\big(\mathbf{z};\, \mathbf{H}\mathbf{x},\, \mathbf{R}\big)\,\mathcal{N}\big(\mathbf{x};\, \mathbf{y},\, \mathbf{P}\big)$ as a joint density $p(\mathbf{z}, \mathbf{x}) = p(\mathbf{z}|\mathbf{x})p(\mathbf{x})$. It can be written as a Gaussian, from which the marginal and conditional densities $p(\mathbf{z})$, $p(\mathbf{x}|\mathbf{z})$ can be derived. In the calculations we make use of known formulae for the inverse of a partitioned matrix (see [2, p. 22], e.g.). From $p(\mathbf{z}, \mathbf{x}) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z})$ the formula results.

Joseph form

The Joseph form states the filter covariance update in a numerical stable way. Given a prediction covariance $\mathbf{P}_{k|k-1}$, a measurement error covariance $\mathbf{R}_k$, and a measurement matrix $\mathbf{H}_k$ at time t_k, the filter covariance $\mathbf{P}_{k|k}$ can be calculated in the following way (see [2, (7.3.2-13)]):

$$\begin{aligned}\mathbf{P}_{k|k} = {} & (\mathbf{1}_\mathbf{x} - \mathbf{W}_{k|k-1}\mathbf{H}_k)\mathbf{P}_{k|k-1}(\mathbf{1}_\mathbf{x} - \mathbf{W}_{k|k-1}\mathbf{H}_k)^\top \\ & + \mathbf{W}_{k|k-1}\mathbf{R}_k\mathbf{W}_{k|k-1}^\top,\end{aligned} \tag{99}$$

where an arbitrary gain matrix $\mathbf{W}_{k|k-1}$ might be used.

References

[1] W. Koch. *Exact update formulae for distributed Kalman filtering and retrodiction at arbitrary communication rates.* In Proc. of the *12th ISIF International Conference on Information Fusion, FUSION 2009*, Seattle, USA, July 6-9, 2009.

[2] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation.* Wiley & Sons, 2001.

[3] X. Tian and Y. Bar-Shalom. *Sequential track-to-track fusion algorithm: exact solution and approximate implementation.* In Proc. of the *SPIE, the International Society for Optical Engineering*, Orlando, Florida, USA, 18-20 March, 2008.

[4] S. Blackman. *Multiple Hypothesis Methods.* In *IEEE AES Magazine*, **19**, 1, p. 41–52, 2004.

[5] S. Blackman and R. Populi. *Design and Analysis of Modern Tracking Systems.* Artech House, 1999.

[6] K.-C. Chang, T. Zhi, R.K. Saha. *Performance Evaluation of Track Fusion with Information Matrix Filter.* IEEE AES, Vol. 38, No. 2, pp. 455, April 2002.

[7] H. Chen, T. Kirubarajan, Y. Bar-Schalom. *Performance Limits of Track-to-Track Fusion versus Centralized Fusion: Theory and Applications.* IEEE AES, Vol. 39, No. 2, pp. 386, April 2003.

[8] C.-Y. Chong, S. Mori, H. Barker, K.-C. Chang. *Architectures and Algorithms for Track Association and Fusion.* IEEE AES Systems Magazine, January 2000.

[9] O. E. Drummond. *Target Tracking with Retrodicted Discrete Probabilities.* In *SPIE **3163**, Signal and Data Processing of Small Targets*, 249 (1997).

[10] O. E. Drummond. *Tracklets and a Hybrid Fusion with Process Noise.* In *SPIE **3163**, Signal and Data Processing of Small Targets* (1997).

[11] H. E. Rauch, F. Tung, and C. T. Striebel. *Maximum Likelihood Estimates of Linear Dynamic Systems.* In *American Institute of Aeronautics and Astronautics Journal*, Vol. 3, No. 8, pp. 1445, August 1965

[12] A. Gelb (Ed.). *Applied Optimal Estimation*, MIT Press (1974).

[13] A. P. Sage and J. L. Melsa (Ed.). *Estimation theory with applications to communications and control*, McGraw-Hill (1979).

[14] D.L. Hall and J. Llinas (Eds.). *Handbook of Multisensor Data Fusion.* CRC Press, 2001.

[15] W. Koch. *Target Tracking.* Chapter 8 in *S. Stergiopoulos (Ed.), Signal Processing Handbook.* CRC Press, 2001.

[16] S. Mori, W.H. Barker, C.-Y. Chong, K.-C. Chang. *Track Association and Track Fusion with Nondeterministic Target Dynamics.* IEEE AES, Vol. 38, No. 2, pp. 659, April 2002.

[17] P. L. Ainsleigh, N. Kehtarnavaz, R. L. Streit. *Hidden Gauss-Markov models for signal classification*, Signal Processing, IEEE Transactions on , vol.50, no.6, pp.1355-1367, Jun 2002

Felix Govaers received his Diploma in Mathematics at the University of Bonn, Germany in 2007. Since then, he has been working at the University of Bonn at the Institute Computer Science 4, department for Communication Systems. Since 2009 he additionally works at the Fraunhofer Institute for Communication, Information and Ergonomics (FKIE), department for Sensor data and Information Fusion (SDF). His research interests are focused on aspects of information fusion in distributed sensor applications.

Wolfgang Koch studied Physics and Mathematics at Aachen Technical University (RWTH), Germany, where he earned a "Dr. rer. nat." degree in Theoretical Physics. He is head of the dept. "Sensor Data and Information Fusion" at Fraunhofer FKIE, a German defence research lab. At Bonn University, he is teaching Applied Computer Science as a "Privatdozent". Furthermore he serves the IEEE AESS as an Associate Editor-in-Chief and Technical Editor "Target Tracking and Multisensor Systems" for the IEEE Trans. on AES.