Closed-form Optimization of Covariance Intersection for Low-dimensional Matrices

Marc Reinhardt, Benjamin Noack, and Uwe D. Hanebeck
Intelligent Sensor-Actuator-Systems Laboratory (ISAS),
Institute for Anthropomatics,
Karlsruhe Institute of Technology (KIT), Germany.
Email: marc.reinhardt@ieee.org, benjamin.noack@ieee.org, uwe.hanebeck@ieee.org

***Abstract*—The fusion under unknown correlations is an important technique in sensor-network information processing as the cross-correlations between different estimates remain often unknown to the nodes. Covariance intersection is a widespread and efficient algorithm to fuse estimates under such uncertain conditions. Although different optimization criteria have been developed, the trace or determinant minimization of the fused covariance matrix seems to be most meaningful. However, this minimization requires numeric solutions of a convex optimization problem. We derive an algorithm to reduce this nonlinear optimization to the well-known polynomial root-finding problem. This allows us to present closed-form solutions for the determinant criterion when the dimension of the occurring covariance matrices is at most 4 and for the trace criterion when the dimension of the covariance matrices is at most 3. We demonstrate the effectiveness of the approach by means of a speed evaluation.**

I. Introduction

With the development of small sensor systems, decentralized data fusion has become an important topic in estimation theory. Due to the progress in hardware development over the last two decades, the costs for small systems have dropped considerably while the computational power has been improved and the energy consumption has been lowered. This enables system designers to apply decentralized estimation techniques not only in wide-area observation scenarios with massive nodes such as the tsunami warning system, but also in applications where data from multiple small sensor sources is combined in a decentralized fashion without the need for a central processing system.

Estimating the state of one phenomenon in multiple nodes requires further considerations compared to a central Kalman filter processing. When all nodes utilize the same system model, the same process noise is modeled at all nodes, which, in turn, leads to dependencies between the local estimates that make it difficult to estimate linearly optimal in a global sense [1].

Besides this source of correlations, which is often referred to as common process noise in literature [2], the estimates are typically exchanged and combined regularly, which again implies a correlation between the local estimates that is caused by common prior information [3], [4]. In sensor networks with a tree topology, the common prior information can be cached with the Channel Filter [4]. In more complex network topologies or when the network topology is unknown, common prior information in general comes not from estimates of direct neighbours and thus, can only be identified by a complex communication history and with a significantly higher communication effort.

Due to these effects, the estimates of the local nodes are correlated, which, in turn, prevents the nodes from applying a joint Kalman filter as the global optimal filter step would require the remote nodes to be updated when measurements are incorporated into local estimates. Although there exists a solution to bypass this problem when global knowledge about the utilized models is available to all nodes [5], the decentralized estimation under consideration of the correct correlations is often challenging. This motivates the application of suboptimal methods such as the Channel filter [4] or the Bar-Shalom/Campo formula [2] that take only one of the correlation sources each into account.

Alternatively, the estimates from different nodes are fused under unknown dependencies. In this research area, different approaches have been proposed that optimize the fusion of two estimates according to different criteria. Ellipsoidal intersection [6], [7] calculates the fusion under the assumption that the common information is maximized, the uniform distribution approach [8] considers the possible correlations and takes an average over all potential estimates, and covariance intersection(CI) [9], [10] determines a consistent estimate[1] with a tight covariance bound.

The most popular of them is CI for what different optimization criteria have been developed and theoretic attributes have been derived. In [11] it was proven that CI with trace minimization calculates the trace-minimizing covariance matrix in a family of consistent linear combinations. Although it remains to show that CI yields the fused covariance matrix with minimum trace – respectively minimum determinant – when the optimization criterion is chosen accordingly, this seems to be more a theoretic problem than an open question.

However, as CI is especially meaningful when the trace or the determinant of the fused covariance matrix is minimized[2], the application of the algorithm requires the solution of a (convex) nonlinear optimization problem. As this is computationally expensive, extensions to CI have been proposed that calculate

[1]The difference between calculated covariance matrix and real error covariance matrix is positive-semidefinite.

[2]The decision which of the criteria is best to apply is problem specific and is not treated in this paper.

suboptimal optimizations in closed-form [12], [13] or optimize other criteria than the trace or determinant [14], [15].

In this paper we present closed-form solutions for the optimization of CI for covariance matrices with dimension less than 5 when the determinant is minimized and for less then 4-dimensional covariance matrices when the trace is minimized. We furthermore propose a strategy to solve the optimization problems in higher dimensions more effectively than the naïve procedure.

The basic idea of this paper as well as some interesting attributes concerning the tight intersection of ellipsoids have been presented by Kahan in 2006 in an update [16] of the article [17] with the same name. As the authors derived the algorithm independently of Kahan and it seems to be worthwhile to present the application of the idea to CI, we further extended and evaluated the algorithm.

The structure of the paper is as follows. In Sec. II we shortly present CI and a technique to jointly diagonalize two covariance matrices that is utilized in the proposed approach. We derive the closed-form solutions for both optimization criteria in Sec. III and discuss the results in Sec. IV. The performance of the proposed approach is then demonstrated by means of a runtime evaluation in comparison to two different naïve CI implementations.

II. Problem Statement

A. Covariance Intersection

CI is an algorithm to efficiently combine two estimates under unknown correlations. Assume the estimates $\underline{x}_1$ and $\underline{x}_2$ with error covariance matrix bounds $\mathbf{C}_1$ and $\mathbf{C}_2$ to be given. We assume the estimates to be consistent, i.e., $\mathrm{E}\left\{\left(\underline{\boldsymbol{x}} - \underline{x}_i\right)\left(\underline{\boldsymbol{x}} - \underline{x}_i\right)^T\right\} \leq \mathbf{C}_i \;\forall i \in 1,2$ where $\underline{\boldsymbol{x}}$ denotes the real state and $\leq$ indicates that the difference of the two matrices is positive-semidefinite.

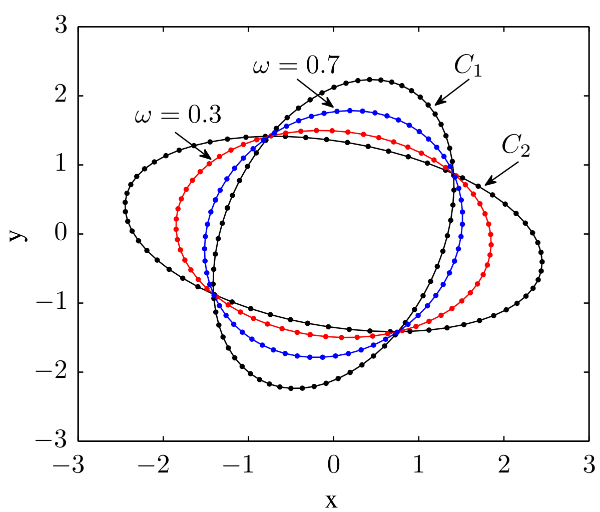

Figure 1. The error ellipses of two covariance matrices $\mathbf{C}_1 = [5\ 1; 1\ 2]$ and $\mathbf{C}_2 = [2\ {-1}; -1\ 6]$ are given as the black lines and the CI fused covariance matrix $\mathbf{C}^\omega$ is given in red for $\omega = 0.3$ in and in blue for $\omega = 0.7$.

The CI combination formulas are given by

$$\begin{aligned}\mathbf{C}^\omega &= \left(\omega\left(\mathbf{C}_1\right)^{-1} + (1-\omega)\left(\mathbf{C}_2\right)^{-1}\right)^{-1} \\ \underline{x}^\omega &= \mathbf{C}^\omega\left(\left(\mathbf{C}_1\right)^{-1}\underline{x}_1 + \left(\mathbf{C}_2\right)^{-1}\underline{x}_2\right)\ ,\end{aligned} \tag{1}$$

where $\omega \in [0,1]$ is a design parameter that is optimized according to some criteria. Typically, the determinant or trace of the fused covariance matrix $\mathbf{C}^\omega$ is minimized. The motivation for (and the name of) CI comes from geometry as the fused covariance error ellipsoids enclose the intersection of the input covariance error ellipsoids, which can be seen in Figure 1 for two different parameter values.

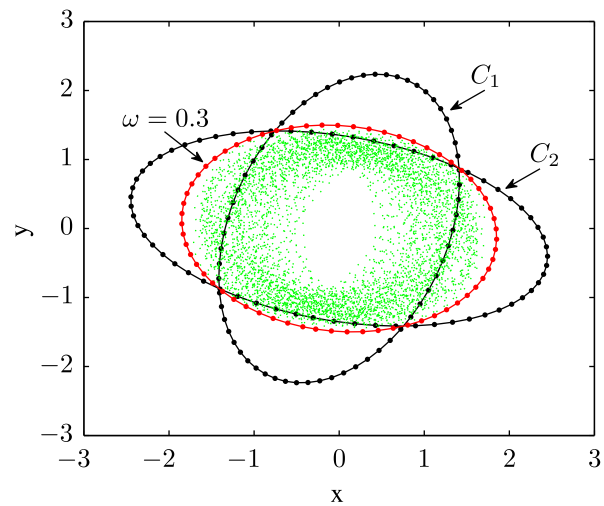

Figure 2. The same error ellipsoids as in Figure 1 without the error ellipsoid for $\omega = 0.7$. The green dotted lines mark the error ellipsoids of the true error covariance matrices that are calculated with randomly chosen cross-covariances.

From a theoretic point of view more important is the fact that the difference between calculated covariance matrix $\mathbf{C}^\omega$ and real covariance matrix $\mathrm{E}\left\{\left(\underline{\boldsymbol{x}} - \underline{x}^\omega\right)\left(\underline{\boldsymbol{x}} - \underline{x}^\omega\right)^T\right\}$ is positive-semidefinite for all valid cross-correlations [9]. This is graphically illustrated in Figure 2 where the error ellipsoid of $\mathbf{C}^\omega$ encapsulates the true error ellipsoids of the fused estimate for randomly selected cross-correlations. This attribute does not only provide a reliable error bound, but also prevents the estimates from diverging.

B. Joint Diagonalization of Two Covariance Matrices

It is well known in literature that two covariance matrices can be jointly diagonalized by transforming one of the covariance matrices to an identity matrix while the other matrix keeps its covariance matrix structure with an orthogonal eigenvector decomposition [6], [18], [19]. This facilitates the application of a second, orthogonal transformation that does not affect the identic matrix and diagonalizes the other matrix.

Assume the covariance matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ to be given. Let $\mathbf{E}_1$ be the diagonal matrix with the eigenvalues of $\mathbf{C}_1$ on its diagonal and let $\mathbf{V}_1$ denote the corresponding eigenvector matrix. The eigenvector matrices of symmetric matrices are orthogonal $\mathbf{V}_1\left(\mathbf{V}_1\right)^T = \mathbf{I}$ and therefore

$$\mathbf{C}_1 = \mathbf{V}_1\mathbf{E}_1\left(\mathbf{V}_1\right)^T = \mathbf{V}_1\sqrt{\mathbf{E}_1}\left(\mathbf{V}_1\right)^T\mathbf{V}_1\sqrt{\mathbf{E}_1}\left(\mathbf{V}_1\right)^T$$

holds. We set $\mathbf{T}_1 = \left(\mathbf{V}_1\sqrt{\mathbf{E}_1}\right)^{-1}$ and obtain the transformed matrices

$$\mathbf{C}_1' = \mathbf{T}_1\mathbf{C}_1\left(\mathbf{T}_1\right)^T = \mathbf{I} \text{ and } \mathbf{C}_2' = \mathbf{T}_1\mathbf{C}_2\left(\mathbf{T}_1\right)^T \ .$$

As $\mathbf{C}_2'$ is still hermitian, an eigenvalue decomposition provides an orthogonal eigenvector matrix $\mathbf{V}_2'$. Let $\mathbf{E}_2'$ be the eigenvalue matrix of $\mathbf{C}_2'$. We define

$$\mathbf{T} = \mathbf{V}_2'\mathbf{T}_1 = \mathbf{V}_2'\left(\sqrt{\mathbf{E}_1}\right)^{-1}\left(\mathbf{V}_1\right)^T \tag{2}$$

and obtain with

$$\mathbf{E}_2' = \mathbf{V}_2'\mathbf{C}_2'\left(\mathbf{V}_2'\right)^T = \mathbf{V}_2'\mathbf{T}_1\mathbf{C}_2\left(\mathbf{T}_1\right)^T\left(\mathbf{V}_2'\right)^T = \mathbf{T}\mathbf{C}_2\left(\mathbf{T}\right)^T$$

the transformed matrices

$$\begin{aligned}\mathbf{C}_1'' &= \mathbf{T}\mathbf{C}_1\mathbf{T}^T = \mathbf{I}\\ \mathbf{C}_2'' &= \mathbf{T}\mathbf{C}_2\mathbf{T}^T = \mathbf{E}_2' \ .\end{aligned}$$

The inverse of the transformation matrix $\mathbf{T}$ is given by

$$\left(\mathbf{T}\right)^{-1} = \mathbf{V}_1\sqrt{\mathbf{E}_1}\mathbf{V}_2' \ .$$

III. Derivation of Closed-form Formulas

In this section, we derive closed-form formulas for the determinant as well as the trace optimization criterion of CI. To do that we reduce the nonlinear optimization problems by means of the joint diagonalization from Sec. II-B to scalar-valued polynomial optimizations.

A. Transformation of the Cost Function

Let J denote an arbitrary cost function. The CI optimization is given as

$$\omega^* = \text{argmin}_\omega\left\{J\left(\left(\omega\left(\mathbf{C}_1\right)^{-1} + (1-\omega)\left(\mathbf{C}_2\right)^{-1}\right)^{-1}\right)\right\} \ . \tag{3}$$

While most of the matrix optimization criteria such as the trace or the determinant are invariant for similarity transformations, this argument does not hold for general matrix transformations $\mathbf{A}\mathbf{C}^\omega\mathbf{A}^T$. As the joint diagonalization of two covariance matrices is in general no similarity transformation, we are only allowed to modify (3) according to

$$\omega^* = \text{argmin}_\omega\left\{J\left(\mathbf{A}^{-1}\mathbf{A}\mathbf{C}^\omega\left(\mathbf{A}\right)^T\left(\mathbf{A}\right)^{-T}\right)\right\} \ , \tag{4}$$

with $\mathbf{C}^\omega$ from (1).

Let $\mathbf{A}$ be the transformation matrix from (2). We obtain for

$$\begin{aligned}\mathbf{A}\mathbf{C}^\omega(\mathbf{A})^T &= \mathbf{A}\left(\omega\left(\mathbf{C}_1\right)^{-1} + (1-\omega)\left(\mathbf{C}_2\right)^{-1}\right)^{-1}\left(\mathbf{A}\right)^T\\ &= \left(\omega\left(\mathbf{A}\mathbf{C}_1\left(\mathbf{A}\right)^T\right)^{-1} + (1-\omega)\left(\mathbf{A}\mathbf{C}_2\left(\mathbf{A}\right)^T\right)^{-1}\right)^{-1}\\ &= \left(\omega\mathbf{I} + (1-\omega)\left(\mathbf{D}\right)^{-1}\right)^{-1} \ .\end{aligned}$$

The matrix $\mathbf{D}$ is a diagonal matrix with positive scalars $d_1, \ldots, d_n$. As both matrices within the brackets are diagonal, the inverse can be calculated element-wise. Thus, we obtain

$$\mathbf{A}\mathbf{C}^\omega(\mathbf{A})^T = \text{diag}\left(\frac{1}{\omega + (1-\omega)\bar{d}_1}, \ldots, \frac{1}{\omega + (1-\omega)\bar{d}_n}\right) , \tag{5}$$

with $\bar{d}_i = \frac{1}{d_i} \ \forall i \in 1, \ldots, n$.

In the following, trace and determinant optimization criteria are investigated and closed-form solutions are derived by performing an extreme value search. As the optimization is convex [13], a zero point of the derivation characterizes a minimum when it is from the interval $[0, 1]$. Nevertheless, it is not guaranteed that the optimal ω^* lies within the open interval and so, a manual check whether $\mathbf{C}_1$ or $\mathbf{C}_2$ provides lower costs is necessary when the combination of both matrices does not improve the costs.

B. Determinant Minimization

For $J(\mathbf{C}^\omega) = \det\{\mathbf{C}^\omega\}$, equation (4) is simplified to

$$\begin{aligned}\omega^* &= \text{argmin}_\omega\left\{\det\left\{\mathbf{A}^{-1}\mathbf{A}\mathbf{C}^\omega\left(\mathbf{A}\right)^T\left(\mathbf{A}\right)^{-T}\right\}\right\}\\ &= \text{argmin}_\omega\left\{\det\left\{\mathbf{A}^{-1}\right\}\det\left\{\left(\mathbf{A}\right)^{-T}\right\}\det\left\{\mathbf{A}\mathbf{C}^\omega\left(\mathbf{A}\right)^T\right\}\right\} \ .\end{aligned} \tag{6}$$

With regular $\mathbf{A}$, it follows $\det\left\{\mathbf{A}^{-1}\right\} \neq 0$. As the determinant is invariant to the transposed attribute, it holds $\det\left\{\mathbf{A}^{-1}\right\}\det\left\{\left(\mathbf{A}\right)^{-T}\right\} = \det\left\{\mathbf{A}^{-1}\right\}^2 > 0$ and therefore, the optimization (6) equals

$$\omega^* = \text{argmin}_\omega\left\{\det\left\{\mathbf{A}\mathbf{C}^\omega\left(\mathbf{A}\right)^T\right\}\right\} \ .$$

With (5), the optimization is given as

$$\begin{aligned}\omega^* &= \text{argmin}_\omega\left\{\prod_{i=1}^n \frac{1}{\omega + (1-\omega)\bar{d}_i}\right\}\\ &= \text{argmax}_\omega\left\{\prod_{i=1}^n \omega + (1-\omega)\bar{d}_i\right\} \ .\end{aligned} \tag{7}$$

The derivative of the maximization term in (7) is obtained by utilizing the derivative product rule as

$$\det\left\{\mathbf{A}\mathbf{C}^\omega\left(\mathbf{A}\right)^T\right\}' = \sum_{i=1}^n \left(1-\bar{d}_i\right)\prod_{j\neq i}\left(\omega + (1-\omega)\bar{d}_j\right) \ .$$

With

$$\omega + (1-\omega)\bar{d}_j = (1-\bar{d}_j)\left(\omega + \frac{\bar{d}_j}{1-\bar{d}_j}\right) \ \forall j \in 1, \ldots, n$$

the derivative is simplified up to

$$\det\left\{\mathbf{A}\mathbf{C}^\omega\left(\mathbf{A}\right)^T\right\}' = \sum_{i=1}^n (1-\bar{d}_i)\prod_{j\neq i}(1-\bar{d}_j)\left(\omega + \frac{\bar{d}_j}{1-\bar{d}_j}\right) \ . \tag{8}$$

We set (8) to zero, divide the equation by $\prod_{i=1}^n (1-\bar{d}_i)$ and obtain with the substitution $\tilde{d}_j = \frac{\bar{d}_j}{1-\bar{d}_j} \ \forall j \in 1, \ldots, n$

$$0 = \sum_{i=1}^n \prod_{j\neq i}\left(\omega + \tilde{d}_j\right) \ , \tag{9}$$

which is equivalent to a polynomial derived in [20]. Thus, the extreme-value determination is reduced to the problem of finding the roots of a polynomial of rank $n-1$ that is given as

$$0 = n\cdot\omega^{n-1} + (n-1)\pi_1\omega^{n-2} + (n-2)\pi_2\omega^{n-3} + \cdots + \pi_{n-1} \tag{10}$$

with

$$\pi_i = \sum_{j_1=1}^{n} \sum_{j_2=j_1+1}^{n} \cdots \sum_{j_i=j_{i-1}+1}^{n} \tilde{d}_{j_1} \cdots \tilde{d}_{j_n} \ .$$

From the Abel-Ruffini theorem we know that there is no algebraic solution when the dimension of the polynomial is above 4 ($n > 5$). For polynomials with a degree below 5, however, closed-form solutions can be given. We provide such solutions for $n \leq 3$ below and for $n = 4$ in Appendix A. Note that also a solution for (10) with $n = 5$ can be obtained in closed form. But as the eigenvalue decomposition of the corresponding matrices requires numeric approaches in this case (c.f. Sec. IV) and the equations become lengthy we will not present this formulas here.

For $n = 1$, the problem of finding the best ω is trivial as it holds $\omega = 1$ when $C_1 < C_2$ and $\omega = 0$ otherwise. For $n = 2$, equation (10) is given by

$$\begin{aligned} 0 = 2 \cdot \omega + \tilde{d}_1 + \tilde{d}_2 \Leftrightarrow \omega = -\frac{1}{2}\left(\tilde{d}_1 + \tilde{d}_2\right) \\ \Leftrightarrow \omega = -\frac{1}{2}\left(\frac{1}{1-d_1} + \frac{1}{1-d_2}\right) \ . \end{aligned} \tag{11}$$

For $n = 3$, equation (10) is given as

$$\begin{aligned} 0 = 3 \cdot \omega^2 + 2\left(\tilde{d}_1 + \tilde{d}_2 + \tilde{d}_3\right)\omega + \tilde{d}_1\tilde{d}_2 + \tilde{d}_1\tilde{d}_3 + \tilde{d}_2\tilde{d}_3 \Leftrightarrow \\ \omega_{1/2} = -\frac{1}{3}\Bigg(\left(\tilde{d}_1 + \tilde{d}_2 + \tilde{d}_3\right) \pm \\ \sqrt{\tilde{d}_1^2 + \tilde{d}_2^2 + \tilde{d}_3^2 - \tilde{d}_1\tilde{d}_2 - \tilde{d}_1\tilde{d}_3 - \tilde{d}_2\tilde{d}_3}\Bigg) \ . \end{aligned} \tag{12}$$

As it has been mentioned, the optimization problem (3) is convex on the interval $[0, 1]$ and therefore, there is only one valid solution for $\omega^* \in [0, 1]$. A detailed approach to extract the optimal ω^* from the candidates is given by Algorithm 1.

C. Trace Minimization

In contrast to the determinant, the trace of the product of matrices does not equal the product of the trace of the matrices. However, as the trace is invariant to cyclic permutations

$$\mathrm{tr}\left\{\mathbf{A}^{-1}\mathbf{A}\mathbf{C}^{\omega}\mathbf{A}^T\mathbf{A}^{-T}\right\} = \mathrm{tr}\left\{\mathbf{A}^{-T}\mathbf{A}^{-1}\mathbf{A}\mathbf{C}^{\omega}\mathbf{A}^T\right\}$$

holds. Again, we assume $\mathbf{A}$ to be the transformation matrix from (2) and obtain

$$\omega^* = \mathrm{argmin}_{\omega}\left\{\sum_{i=1}^{n} \frac{a_i}{\omega + (1-\omega)\bar{d}_i}\right\} \ , \tag{13}$$

where $a_i > 0$ denotes the ith diagonal element of $\mathbf{A}^{-T}\mathbf{A}^{-1}$ $\forall i \in 1, \ldots, n$.

The derivation of the trace directly follows as

$$\begin{aligned} \mathrm{tr}\left\{\mathbf{A}\mathbf{C}^{\omega}\mathbf{A}^T\right\}' &= \sum_{i=1}^{n} \frac{a_i\left(1-\bar{d}_i\right)}{\left(\omega + (1-\omega)\bar{d}_i\right)^2} \\ &= \sum_{i=1}^{n} \frac{a_i\left(1-\bar{d}_i\right)}{\left(\omega(1-\bar{d}_i) + \bar{d}_i\right)^2} \ . \end{aligned} \tag{14}$$

We simplify (14) to

$$\sum_{i=1}^{n} a_i\left(1-\bar{d}_i\right)^{-1}\left(\omega + \frac{\bar{d}_i}{1-\bar{d}_i}\right)^{-2} \ , \tag{15}$$

set the equation to zero and obtain

$$0 = \sum_{i=1}^{n} a_i\left(1-\bar{d}_i\right)^{-1}\prod_{j\neq i}\left(\omega + \frac{\bar{d}_j}{1-\bar{d}_j}\right)^2 \ . \tag{16}$$

By substituting $\tilde{d}_i = \frac{\bar{d}_i}{1-\bar{d}_i}$ $\forall i \in 1, \ldots, n$, we obtain

$$0 = \sum_{i=1}^{n} a_i\left(1+\tilde{d}_i\right)\prod_{j\neq i}\left(\omega + \tilde{d}_j\right)^2 \ . \tag{17}$$

Again, equation (17) is a polynomial. As the normal form of this polynomial does not provide new insights and has a complicated and elongated structure, we do not present it here and give only closed-form solutions for $n \leq 2$. Solely, it is worth to mention that the degree of the polynomial is $2(n-1)$ and so – following the argumentation from the determinant minimization – a general algebraic solution is only found for $n \leq 3$.

For $n = 1$, the problem is equivalent to the determinant minimization. For $n = 2$, the polynomial from equation (17) is given as

$$\begin{aligned} 0 =& a_1(1+\tilde{d}_1)(\omega^2 + 2\omega\tilde{d}_2 + \left(\tilde{d}_2\right)^2) + \\ & a_2(1+\tilde{d}_2)(\omega^2 + 2\omega\tilde{d}_1 + \left(\tilde{d}_1\right)^2) \ \Leftrightarrow \\ 0 =& \omega^2 + 2\frac{a_1\tilde{d}_2(1+\tilde{d}_1) + a_2\tilde{d}_1(1+\tilde{d}_2)}{a_1(1+\tilde{d}_1) + a_2(1+\tilde{d}_2)}\omega + \\ & \frac{a_1\left(\tilde{d}_2\right)^2(1+\tilde{d}_1) + a_2\left(\tilde{d}_1\right)^2(1+\tilde{d}_2)}{a_1(1+\tilde{d}_1) + a_2(1+\tilde{d}_2)} \ . \end{aligned} \tag{18}$$

As this is a quadratic equation, we directly obtain the two candidates

$$\omega_1 = -p + \sqrt{p^2 - q} \text{ and } \omega_2 = -p + \sqrt{p^2 - q} \tag{19}$$

with

$$\begin{aligned} p =& \frac{a_1\tilde{d}_2(1+\tilde{d}_1) + a_2\tilde{d}_1(1+\tilde{d}_2)}{a_1(1+\tilde{d}_1) + a_2(1+\tilde{d}_2)} \text{ and} \\ q =& \frac{a_1\left(\tilde{d}_2\right)^2(1+\tilde{d}_1) + a_2\left(\tilde{d}_1\right)^2(1+\tilde{d}_2)}{a_1(1+\tilde{d}_1) + a_2(1+\tilde{d}_2)} \ . \end{aligned}$$

D. Algorithm

We conclude this section with a skeleton algorithm that simplifies the implementation of the proposed closed-form solution of CI. Basically, we show how to choose the correct candidate and describe the handling when the extreme value search does not provide a solution in the specified interval. This occurs – but is not limited to the case – when the eigenvalues of one covariance are all smaller than those of the other covariance.

The notation corresponds basically to that of MATLAB with element wise division '`./`', '%' for comments, 'diag' for the diagonal elements of a matrix and a matrix decomposition in eigenvector- and eigenvalue matrices with 'eig'.

Input: $\mathbf{C}_1, \mathbf{C}_2, J$

% Joint diagonalization
$[\mathbf{V}_1, \mathbf{D}_1] \leftarrow \mathrm{eig}(\mathbf{C}_1)$;
$\mathbf{T}_1 \leftarrow (\sqrt{D_1})^{-1}(\mathbf{V}_1)^T$;
$[\mathbf{V}_2', \mathbf{D}_2'] \leftarrow \mathrm{eig}(\mathbf{T}_1\mathbf{C}_2(\mathbf{T}_1)^T)$;
$\underline{\bar{d}} \leftarrow (\mathrm{diag}(\mathbf{D}_2'))^{-1}$;
$\underline{\tilde{\bar{d}}} \leftarrow \underline{\bar{d}}./(\underline{1} - \underline{\bar{d}})$;

% Find optimal ω
if J == trace **then**
    $\underline{a} \leftarrow \mathrm{diag}(\mathbf{V}_2'\mathbf{D}_1(\mathbf{V}_2')^T)$;
    $\underline{\omega}_{cand} \leftarrow$ candidates from (19);
else if J == det **then**
    $\underline{\omega}_{cand} \leftarrow$ candidates from one of (11), (12) or (20);
end if
$\omega \leftarrow \underline{\omega}_{cand} \cap [0, 1]$;

% Handle ω out of bounds
if $\omega = \varnothing$ **then**
    $\omega \leftarrow (J(\mathbf{C}_1) < J(\mathbf{C}_2))$? 1 : 0;
end if

Output: ω

Algorithm 1: Skeleton algorithm for closed-form CI.

The function J in Algorithm 1 is a place-holder for the optimization criterion and is to be replaced by trace or determinant. It is worth to point out that both optimization criteria are convex on the interval $[0, 1]$ and therefore, a solution obtained from the interval always denotes a minimum.

IV. Discussion

In this section, the proposed closed-form formulas for CI are discussed from different point of views. Before benchmarks are presented, we investigate the effort of the utilized mathematic methods. Besides simple matrix operations, such as the inverse, the multiplications and so on, which can be obtained in closed form, the joint diagonalization of two covariance matrices requires an eigenvalue decomposition. The eigenvalue decomposition of a matrix can be reduced to the problem of finding the roots of the corresponding characteristic polynomial and thus, can be calculated in closed form when the dimension of the matrix is below 5.

When the proposed procedure is used to speed up CI for matrices of dimension above 4, additional computational effort is necessary. While efficient, numerically stable algorithms such as the Cholesky decomposition and the QR-algorithm (and its derivatives) exist for matrix operations on positive-definite symmetric matrices [19], [21], the eigenvalue decomposition and the root-finding of the derivative must be solved by means of numeric algorithms. Nevertheless, as the derivatives of the trace (17) and determinant (10) optimizations are given as simple polynomials, efficient root-finding algorithms such as Laguerre's method can be applied that outperform the direct covariance matrix optimizations of (3). Consequently, a speed improvement is obtained by utilizing the proposed approach even when the dimension of the occurring matrices is to high to provide closed-form solutions.

For a speed evaluation of the closed-form and the naïve CI algorithm, the MATLAB implementation of Julier from [22], an improved variant of this implementation and the skeleton Algorithm 1 have been compared with the MATLAB profiler.

We randomly generated 10^6 joint covariance matrices for each test and extracted the top-left and the bottom-right block matrices as $\mathbf{C}_1$ and $\mathbf{C}_2$ for the evaluation. All approaches were utilized to obtain the optimal ω^*, where assertions made sure that all approaches yielded – besides numeric inaccuracies – the same ω^*. We compared the CPU-time for different optimization criteria and dimensions and obtained the following results

	CI naïve	CI naïve opt.	closed-form
det(n=2)	1075.44 s	274.71 s	22.56 s
det(n=3)	933.95 s	257.19 s	29.91 s
trace(n=2)	1017.64 s	309.62 s	24.90 s

As can be seen, the speed-up of the closed-form solutions is remarkably for all dimensions and optimization criteria. The performance increase between the two naïve implementations is due to a more efficient call of 'fminbnd'.

It is worth to mention that the closed-form formulas are implemented with the standard eigenvalue decomposition method of MATLAB that does not calculate the eigenvalues in closed from. These calculations count for approximately 25% of the computation effort of the closed-form approach and can be reduced significantly.

Overall, the closed-form CI calculations are between 31 and 48 times faster than the naïve implementation from [22] and still 8.6 to 12.4 times faster than the optimized naïve implementation.

While the performance improvements are an obvious advantage of the proposed closed-form approach, it is worth to point out that the precision of ω^* depends only on the value representation in the computer and no longer on the number of iterations in the numeric procedures. Although this is probably not relevant in practical applications, it might be interesting for theoretic considerations.

V. Conclusion

In this paper we proposed a skeleton algorithm to speed up the CI trace and determinant optimization. It has been shown that the algorithm does not require numeric calculations when the determinant minimization is performed with covariance matrices of dimension lower than 5 or when the trace optimization is performed with covariance matrices of dimension lower than 4. In an evaluation with random covariance matrices we have demonstrated that the new algorithm outperforms the naïve CI optimization implementation by a factor higher than 8.

In particular, when only low-dimensional covariance matrices are combined, the proposed algorithm can guarantee exact results under tight runtime constraints. As there is no need for optimization methods it is also preferable from an implementation view. In summary, the proposed algorithm is beneficial when only low computation power is available or when large scenarios are simulated and the simulation time depends for the most part on CI optimizations.

Although there seems to be no chance to solve the optimization for arbitrary dimensions in closed form when the proposed approach is utilized, it remains to improve the root-finding algorithm application. Additionally, the fusion of multiple covariance matrices with CI is an interesting area. While it is obvious that the block-fusion supplies better results than the recursive one, the optimization becomes complex as the number of optimization parameters linearly increases with the number of covariance matrices to fuse. Especially for low-dimensional covariance matrices, this problem might be solvable when an algorithm to simultaneously diagonalize multiple matrices is applied. Although such an algorithm will be suboptimal as the simultaneous diagonalization of more than two matrices can be calculated approximately only, the overall result will likely be better than a recursive application of CI.

Acknowledgements

This work was partially supported by the German Research Foundation (DFG) within the Research Training Group GRK 1194 'Self-organizing Sensor-Actuator-Networks' and by the Karlsruhe Institute of Technology (KIT).

Appendix

A. Determinant Minimization of 4-dimensional Matrices

We assume $\tilde{d}_i$ to be defined as in (9) $\forall i \in 1, \ldots, 4$. From (10), we obtain the condition

$$0 = 4\omega^3 + 3\left(\sum_{i=1}^{4} \tilde{d}_i\right)\omega^2 + 2\left(\sum_{i=1}^{n}\sum_{j=i+1}^{n} \tilde{d}_i\tilde{d}_j\right)\omega^1 + \sum_{i=1}^{n}\sum_{j=i+1}^{n}\sum_{k=j+1}^{n} \tilde{d}_i\tilde{d}_j\tilde{d}_k \ .$$

Let $b = \frac{3}{4}\left(\sum_{i=1}^{4} \tilde{d}_i\right)$, $c = \frac{1}{2}\left(\sum_{i=1}^{n}\sum_{j=i+1}^{n} \tilde{d}_i\tilde{d}_j\right)$ and $d = \frac{1}{4}\sum_{i=1}^{n}\sum_{j=i+1}^{n}\sum_{k=j+1}^{n} \tilde{d}_i\tilde{d}_j\tilde{d}_k$. Then, possible candidates for the determinant minimizing ω^* are

$$\begin{aligned} \omega_1 &= -\frac{b}{3} - \frac{C}{3} - \frac{b^2 - 3c}{3C} \\ \omega_2 &= -\frac{b}{3} + \frac{C(1 + i\sqrt{3})}{6} + \frac{(1 - i\sqrt{3})(b^2 - 3c)}{6C} \\ \omega_3 &= -\frac{b}{3} + \frac{C(1 - i\sqrt{3})}{6} + \frac{(1 + i\sqrt{3})(b^2 - 3c)}{6C} \end{aligned} \qquad (20)$$

with

$$\begin{aligned} Q &= \sqrt{(2b^3 - 9bc + 27d)^2 - 4(b^2 - 3c)^3} \\ C &= \sqrt[3]{\tfrac{1}{2}(Q + 2b^3 - 9bc + 27d)} \ . \end{aligned}$$

References

[1] M. Reinhardt, B. Noack, and U. D. Hanebeck, "On Optimal Distributed Kalman Filtering in Non-ideal Situations," in *Proceedings of the 15th International Conference on Information Fusion (Fusion 2012)*, 2012.

[2] Y. Bar-Shalom and L. Campo, "The Effect of the Common Process Noise on the Two-Sensor Fused-Track Covariance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-22, no. 6, pp. 803 –805, Nov. 1986.

[3] C. Chong, E. Tse, and S. Mori, "Distributed Estimation in Networks," in *American Control Conference, 1983*. IEEE, 1983, pp. 294–300.

[4] A. Makarenko and H. Durrant-Whyte, "Decentralized Data Fusion and Control in Active Sensor Networks," in *Proceedings of the 7th International Conference on Information Fusion(FUSION 2004)*, 2004.

[5] W. Koch, "On Accumulated State Densities with Applications to Out-of-Sequence Measurement Processing," in *Proceedings of the 12th International Conference on Information Fusion(FUSION 2009)*, 2009, pp. 2201–2208.

[6] J. Sijs, M. Lazar, and P. P. J. v. d. Bosch, "State Fusion with Unknown Correlation: Ellipsoidal Intersection," in *Proceedings of the American Control Conference (ACC 2010)*, 2010, pp. 3992–3997.

[7] J. Sijs and M. Lazar, "Empirical Case-studies of State Fusion Via Ellipsoidal Intersection," in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, Jul. 2011, pp. 1 –8.

[8] M. Reinhardt, B. Noack, M. Baum, and U. D. Hanebeck, "Analysis of Set-theoretic and Stochastic Models for Fusion Under Unknown Correlations," in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, Chicago, Illinois, USA, Jul. 2011.

[9] S. J. Julier and J. K. Uhlmann, "A Non-divergent Estimation Algorithm in the Presence of Unknown Correlations," in *Proceedings of the American Control Conference 1997*, vol. 4, Jun. 1997, pp. 2369–2373.

[10] S. Julier and J. Uhlmann, "Simultaneous Localisation and Map Building Using Split Covariance Intersection," in *Proceedings of the International Conference on Intelligent Robots and Systems(IROS 2001)*, vol. 3, 2001, pp. 1257 –1262 vol.3.

[11] L. Chen, P. O. Arambel, and R. K. Mehra, "Fusion under Unknown Correlation - Covariance Intersection as a Special Case," in *Proceedings of the 5th International Conference on Information Fusion (Fusion 2002)*, Annapolis, Maryland, USA, Jul. 2002.

[12] W. Niehsen, "Information fusion based on fast covariance intersection filtering," in *Information Fusion, 2002. Proceedings of the Fifth International Conference on*, vol. 2. IEEE, 2002, pp. 901–904.

[13] D. Franken and A. Hupper, "Improved fast covariance intersection for distributed data fusion," in *Information Fusion, 2005 8th International Conference on*, vol. 1. Ieee, 2005, pp. 7–pp.

[14] Y. Wang and X. Li, "A Fast and Fault-tolerant Convex Combination Fusion Algorithm Under Unknown Cross-correlation," in *Information Fusion, 2009. FUSION'09. 12th International Conference on*. IEEE, 2009, pp. 571–578.

[15] Y. Wang and X. R. Li, "Distributed Estimation Fusion Under Unknown Cross-correlation: An Analytic Center Approach," in *Proceedings of the 13th International Conference on Information Fusion (FUSION 2010)*, 2010, pp. 1–8.

[16] W. Kahan, "Circumscribing an Ellipsoid About the Intersection of Two Ellipsoids," no. 3, 2006.

[17] ——, "Circumscribing an Ellipsoid About the Intersection of Two Ellipsoids," *Can. Math. Bull*, vol. 11, no. 3, pp. 437–441, 1968.

[18] A. R. Benaskeur, "Consistent Fusion of Correlated Data Sources," in *Proceedings of the 28th Annual Conference of the IEEE Industrial Electronics Society*, vol. 4, 2002, pp. 2652–2656.

[19] R. Horn and C. Johnson, *Matrix Analysis*. Cambridge University Press, 2005.

[20] U. Hanebeck and K. Briechle, "New Results for Stochastic Prediction and Filtering with Unknown Correlations," in *Proceedings of International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2001)*. IEEE, 2001, pp. 147–152.

[21] N. Higham, "Analysis of the cholesky decomposition of a semi-definite matrix," 1990.

[22] M. E. Liggins, D. L. Hall, and J. Llinas, *Handbook of Multisensor Data Fusion: Theory and Practice*, 2nd ed., C. Press, Ed. Martin E. Liggins and David Lee Hall and James Llinas, 2008.