A Game-Theoretic Approach to Robust Fusion and Kalman Filtering Under Unknown Correlations

Spyridon Leonidas\textsuperscript{1} and Kostas Daniilidis\textsuperscript{2}

Abstract—This work addresses the problem of fusing two random vectors with unknown cross-correlations. We present a formulation and a numerical method for computing the optimal estimate in the minimax sense. We extend our formulation to linear measurement models that depend on two random vectors with unknown cross-correlations. As an application we consider the problem of decentralized state estimation. The proposed estimator takes cross-correlations into account while being less conservative than the widely used Covariance Intersection. We demonstrate the superiority of the proposed method compared to Covariance Intersection with numerical examples and simulations within the specific application of decentralized state estimation.

I. INTRODUCTION

State estimation is one of the fundamentals problem in control theory and robotics. The most common state estimators are undoubtedly the Kalman filter [1], which is optimal for the case of linear systems, and its generalizations for nonlinear systems: the Extended Kalman Filter (EKF) [2] and the Unscented Kalman Filter (UKF) [3]. In multi-agent systems, the task of state estimation takes a collaborative form in the sense that it involves inter-agent measurements and constraints. Examples are cooperative localization in robotics [4] using relative pose measurements, camera network localization using epipolar constraints [5] and many more. On the one hand, a decentralized solution that scales with the number of agents is necessary. On the other hand, the state estimates become highly correlated as information flows through the network. Ignoring these correlations has grave consequences: estimates become optimistic and result in divergence of the estimator.

The most popular algorithm for fusion under the presence of unknown cross-correlations is the Covariance Intersection (CI) method which was introduced by Julier and Uhlmann [6]. In its simplest form, the Covariance Intersection algorithm is designed to fuse two random vectors whose correlation is not known by forming a convex combination of the two estimates in the information space. Covariance Intersection produces estimates that are provably consistent, in the sense that estimated error covariance is an upper bound of the true error covariance. However, it has been observed [7], [8], [9] that Covariance Intersection produces estimates that are too conservative which may decrease the accuracy and convergence speed of the overall estimator when used as a component of an online estimator.

One of the most prominent applications of the proposed fusion algorithm is distributed state estimation in an EKF-based framework. However, the problem of distributed state estimation is far from new. There have been numerous approaches for EKF-based distributed state estimation and EKF-based cooperative localization. Yet, some of them require that each agent maintains the state of the entire network [4], [7], which is impractical and does not scale with the number of agents, while others ignore correlations [10], [11] in order to simplify the estimation process or use Covariance Intersection and variations of it [12], [13] despite its slow convergence.

The contributions of this work are summarized as follows. First of all, we propose a method for fusion of two random vectors with unknown cross-correlations which is less conservative than the widely used Covariance Intersection (CI) while taking cross-correlations into account. Second of all, we extend our formulation for the case of a linear measurement model. Finally, we present numerical examples and simulations in a distributed state estimation scenario which demonstrate the validity of the proposed approach.

The paper is structured as follows: in Section II we include definitions of consistency and related notions and we introduce the problem at hand. Our game-theoretic approach to fusing two random variables with unknown cross-correlations is the topic of Section III and it is generalized for arbitrary linear measurement models in Section III. In Section V we include details on the implemented numerical algorithm. Numerical examples and simulation results are presented in Sections VI and VII respectively.

II. PROBLEM FORMALIZATION

In this section, we formalize the problem at hand. First, we need a precise definition of consistency.

Definition 2.1 (Consistency [6]): Let \( z \) be a random vector with expectation \( E[z] = \bar{z} \). An estimate \( \tilde{z} \) of \( \bar{z} \) is another random vector. The associated error covariance is denoted \( \Sigma_{zz} = \text{Cov}(\tilde{z} - \bar{z}) \). The pair \((\tilde{z}, \Sigma_{zz})\) is consistent if \( E[\tilde{z}] = \bar{z} \) and \( \Sigma_{zz} \preceq \Sigma_{zz} \).

Problem Statement 1 (Consistent fusion): Given two consistent estimates \((\tilde{x}, \Sigma_{xx}), (\tilde{y}, \Sigma_{yy})\) of \( \bar{z} \), where \( \Sigma_{xx}, \Sigma_{yy} \) are known upper bounds on the true error covariances. The problem at hand consists of fusing the two consistent estimates \((\tilde{x}, \Sigma_{xx}), (\tilde{y}, \Sigma_{yy})\) in a single consistent estimate \((\tilde{z}, \Sigma_{zz})\) where \( \tilde{z} \) is of the form

\[
\tilde{z} = W_x \tilde{x} + W_y \tilde{y}
\]
with $W_x + W_y = I$ in order for the mean to be preserved. The most widely used solution of the above problem is given by the Covariance Intersection algorithm [6]. Given upper bounds $\Sigma_{xx} \succeq \bar{\Sigma}_{xx}$, $\Sigma_{yy} \succeq \bar{\Sigma}_{yy}$ the Covariance Intersection equations read

$$
\begin{align*}
\bar{\Sigma} &= \bar{\Sigma}_{xx}\left\{\omega \Sigma_{xx}^{-1}\bar{x} + \left(1 - \omega\right) \Sigma_{yy}^{-1}\bar{y}\right\} \\
\Sigma_{zz}^{-1} &= \omega \Sigma_{xx}^{-1} + \left(1 - \omega\right) \Sigma_{yy}^{-1}
\end{align*}
$$

(2)

where $\omega \in [0,1]$. It can be immediately seen that $\Sigma_{zz} \left\{\omega \Sigma_{xx}^{-1} + \left(1 - \omega\right) \Sigma_{yy}^{-1}\right\} = I$ which implies $E[\bar{z}] = \bar{\pi}$. Moreover, it is easy to check that $(\bar{\pi}, \Sigma_{zz})$ is consistent. The above can be easily generalized for the case of more than 2 random variables, for partial measurements and for the linear measurement model we consider in Section IV. Usually, $\omega$ is chosen such that either $\mathrm{tr}(\Sigma_{zz}^{-1})$ or $\log \det(\Sigma_{zz}^{-1})$ is minimized.

Next, we introduce a notion related to consistency but with relaxed requirements. Let $S_+^n$ denote the positive semidefinite cone, that is the set of $n \times n$ positive semidefinite matrices. First, recall that a function $f : S_+^n \to \mathbb{R}$ is called $S_+^n$-nondecreasing [14] if $X \succeq Y$ implies $f(X) \geq f(Y)$, for any $X, Y \in S_+^n$. An example of such a function is $f(X) = \mathrm{tr}(X)$. Now, we are ready to introduce the notion of consistency with respect to a $S_+^n$-nondecreasing function.

**Definition 2.2 (f-Consistency):** Let $f : S_+^n \to \mathbb{R}$ be a nondecreasing function (with respect to $S_+^n$) satisfying $f(0) = 0$. Let $z$ be a random vector with expectation $E[z] = \pi$ and $\bar{z}$ be an estimate of $\pi$ with associated error covariance $\Sigma_{zz}$. The pair $(\bar{z}, \Sigma_{zz})$ is $f$-consistent if $E[\bar{z}] = \pi$ and $f(\Sigma_{zz}) \geq f(\Sigma_{zz})$.

**Remark 1:** Observe that consistency implies $f$-consistency. However, the converse in not necessarily true.

**Problem Statement 2 (Trace-consistent fusion):** Given two consistent estimates $(\bar{x}, \Sigma_{xx})$, $(\bar{y}, \Sigma_{yy})$ of $\pi$, where $\Sigma_{xx}$, $\Sigma_{yy}$ are known upper bounds on the true error variances. The problem at hand consists of fusing the two consistent estimates $(\bar{x}, \Sigma_{xx})$, $(\bar{y}, \Sigma_{yy})$ in a single trace-consistent estimate $(\bar{z}, \Sigma_{zz})$, where $\bar{z}$ is a linear combination of $x$ and $y$ and $\mathrm{tr}(\Sigma_{zz}) \geq \mathrm{tr}(\Sigma_{zz})$.

### III. ROBUST FUSION

The goal of this section is the derivation of our minimax approach. First, we need some basic notions from game theory. A two-player game on $\mathbb{R}^m \times \mathbb{R}^n$ is defined by a pay-off function $f : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$. Intuitively, the first player makes a move $u \in \mathbb{R}^m$ then, the second player makes a move $v \in \mathbb{R}^n$ and receives payment from the first player equal to $f(u,v)$. The goal of the first player is to minimize its payment and the goal of the second player is to maximize the received payment. The game is convex-concave if the pay-off function $f(u,v)$ is convex in $u$ for fixed $v$ and concave in $v$ for fixed $u$. For a review minimax and convex-concave games in the context of convex optimization, we refer the reader to [15].

Let $z$ be a random vector with expectation $E[z] = \pi$. Assume we have two estimates $(\bar{x}, \Sigma_{xx})$, $(\bar{y}, \Sigma_{yy})$ of $\pi$ where $\Sigma_{xx}$, $\Sigma_{yy}$ are approximations to the true error covariances $\Sigma_{xx}$, $\Sigma_{yy}$. Based on the discussion of Section II, the fused estimate is of the form

$$
\bar{z} = (I - K)\bar{x} + K\bar{y}
$$

(3)

and the associated error covariance $\Sigma_{zz}$ is given by

$$
\Sigma_{zz} = [I - K \ K] \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} I - K^T \\ K^T \end{bmatrix}
$$

(4)

However, $\Sigma_{xx}$, $\Sigma_{yy}$ are not known. Therefore, we define

$$
\Sigma_{zz} = [I - K \ K] \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} I - K^T \\ K^T \end{bmatrix}
$$

(5)

where we have the following Linear Matrix Inequality (LMI) constraint on $\Sigma_{yy}$

$$
\begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \succeq 0
$$

(6)

**Remark 2:** It can be seen that $\mathrm{tr}(\Sigma_{zz})$ is convex in $K$ for a fixed $\Sigma_{xy}$ satisfying (6). Therefore, the supremum of $\mathrm{tr}(\Sigma_{zz})$ over all $\Sigma_{xy}$ satisfying (6) is a convex function of $K$. Moreover, for a fixed $K$, $\mathrm{tr}(\Sigma_{zz})$ is linear, and thus concave as well, in $\Sigma_{xy}$ with a convex domain defined by (6). It follows that $\mathrm{tr}(\Sigma_{zz})$ is a convex-concave function in $(K, \Sigma_{xy})$.

As anticipated, we formulate the problem of finding the weighting matrix $K$ as a convex-concave game: the first player chooses $K$ to minimize $\mathrm{tr}(\Sigma_{zz})$ whereas the second player chooses $\Sigma_{xy}$ to maximize $\mathrm{tr}(\Sigma_{zz})$. More specifically, let $(K^*, \Sigma_{xy}^*)$ be the solution to the following minimax optimization problem

$$
\begin{align*}
\text{minimize} & \quad \sup_{\Sigma_{xy}} \ \mathrm{tr}(\Sigma_{zz}) \\
\text{subject to} & \quad \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \succeq 0
\end{align*}
$$

(7)

Then, the fused estimated and the associated error covariance are given by

$$
\begin{align*}
\bar{z} & = (I - K^*)\bar{x} + K^*\bar{y} \\
\Sigma_{zz}^* & = [I - K^* \ K^*] \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \begin{bmatrix} I - K^{*T} \\ K^{*T} \end{bmatrix}
\end{align*}
$$

(8)

Naturally, we have the following lemma.

**Lemma 3.1:** If $(\bar{x}, \Sigma_{xx})$ and $(\bar{y}, \Sigma_{yy})$ are consistent, then the pair $(\bar{z}, \Sigma_{zz})$ given by (8) is trace-consistent.

A proof of lemma 3.1 can be found in [16].

### IV. ROBUST LINEAR UPDATE

In this section, we explore a more general setting. We assume we have two random vectors $x$, $y$ with expectations $E[x] = \pi$ and $E[y] = \bar{\pi}$. We have some estimates $\bar{x}$ and $\bar{y}$ of $\pi$ and $\bar{\pi}$ respectively with associated error covariances $\Sigma_{xx}$ and $\Sigma_{yy}$. As before, we assume that the true error
covariances are only approximately known. Let \( \Sigma_{xx} \) and \( \Sigma_{yy} \) denote these approximate values. We assume we have a linear measurement model of the form
\[
z = Cx + D\bar{y} + \eta \quad (9)
\]
where \( \eta \) is a zero-mean noise process with covariance \( \Sigma_\eta \). We assume that the measurement noise process \( \eta \) is independent of the estimates \( \bar{x} \) and \( \bar{y} \). As in the classic Kalman filter derivation, the update step is of the form
\[
\bar{x}^+ = \bar{x} + K(z - \bar{z}) \quad (10)
\]
where \( \bar{z} \triangleq C\bar{x} + D\bar{y} \). The error of the update is given by
\[
\bar{x}^+ - \bar{x} = (I - KC)(\bar{x} - \bar{x}) - KD(\bar{y} - \bar{y}) + K\eta \quad (11)
\]
and the associated error covariance is defined as \( \tilde{\Sigma}_{xx} \triangleq \text{Cov}((\bar{x}^+ - \bar{x})) \) and is given by
\[
\tilde{\Sigma}_{xx} = \left[ I - KC - KD \right] \left[ \begin{array}{cc} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{array} \right] \left[ I - CT K^T \right] + K\Sigma_\eta K^T \quad (12)
\]
However, the true error covariances \( \tilde{\Sigma}_{xx} \) and \( \tilde{\Sigma}_{yy} \) are not known. Therefore, we define
\[
\Sigma_{xx}^+ = \left[ I - KC - KD \right] \left[ \begin{array}{cc} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{array} \right] \left[ I - CT K^T \right] + K\Sigma_\eta K^T \quad (13)
\]
where \( \Sigma_{xy} \) should satisfy (6) in order to be a valid cross-correlation. To alleviate notation, let \( X = K^T \) and define
\[
f(X, \Sigma_{xy}) \triangleq \text{tr}(\Sigma_{xx}^+). \quad (14)
\]
By rewriting (6) using Schur complement, the minimax formulation is written as follows
\[
\begin{align*}
\text{minimize} \quad & \sup_{X, Q} f(X, Q) \\
\text{subject to} \quad & \Sigma_{yy}^{-1/2} Q^T \Sigma_{xx}^{-1} Q \Sigma_{yy}^{-1/2} - I \preceq 0
\end{align*} \quad (15)
\]
Let \((X^*, Q^*)\) be the optimal solution of problem (15) and let \((K^*, \Sigma_{xy}^*) = (X^{*T}, Q^*)\). Then,
\[
\begin{align*}
\bar{x}^+ &= (I - K^* C)\bar{x} - K^* D\bar{y} \\
\Sigma_{xx}^+ &= \left[ I - K^* C - K^* D \right] \left[ \begin{array}{cc} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{array} \right] \left[ I - CT K^{*T} \right] + K\Sigma_\eta K^T
\end{align*} \quad (16)
\]
Naturally, we have the following lemma.

**Lemma 4.1**: If \((\bar{x}, \Sigma_{xx})\) and \((\bar{y}, \Sigma_{yy})\) are consistent, then the pair \((\bar{x}^+, \Sigma_{xx}^+)\) given by (16) is trace-consistent.

**V. INTERIOR POINT METHODS FOR CONVEX-CONCAVE GAMES**

In this section, we describe the numerical method we use to solve problem (15). First, we will look at the simpler case of an unconstrained convex-concave game with pay-off function \(f(u, v)\). A point \((u^*, v^*)\) is a saddle point for an unconstrained convex-concave game with pay-off function \(f(u, v)\) if
\[
f(u^*, v) \leq f(u^*, v^*) \leq f(u, v^*) \quad (17)
\]
and the optimality conditions for differentiable convex-concave pay-off function are
\[
\nabla_u f(u^*, v^*) = 0, \quad \nabla_v f(u^*, v^*) = 0 \quad (18)
\]
We use the infeasible start Newton method [14], outlined in Algorithm 1, to find the optimal solution of the unconstrained problem:
\[
\begin{align*}
\text{minimize} \quad & f(u, v) \\
\text{maximize} \quad & f(u, v)
\end{align*} \quad (19)
\]
Intuitively, at each step the directions \(\Delta u_{nt}, \Delta v_{nt}\) are the solutions of the first order approximation
\[
0 = r(u + \Delta u_{nt}, v + \Delta v_{nt}) \approx r(u, v) + Dr(u, v)[\Delta u_{nt}, \Delta v_{nt}] \quad (20)
\]
where \(r(u, v) = [\nabla_u f(u, v)^T, \nabla_v f(u, v)^T]^T\). Then, a backtracking line search is performed on the norm of the residual along the previously computed directions.

**Algorithm 1** Infeasible start Newton method.

given: starting points \(u, v \in \text{dom} f\),
tolerance \(\epsilon > 0, \alpha \in (0, 1/2), \beta \in (0, 1)\).

Repeat
1. \(r(u, v) = [\nabla_u f(u, v)^T, \nabla_v f(u, v)^T]^T\)
2. Compute Newton steps by solving
   \[
   Dr(u, v)[\Delta u_{nt}, \Delta v_{nt}] = -r(u, v)
   \]
3. Backtracking line search on \(|r|/2\).
   \[
   t = 1, \quad u_t = u + t\Delta u_{nt}, \quad v_t = v + t\Delta v_{nt}.
   \]
   While \(|r(u_t, v_t)|/2 > (1 - \alpha t)|r(u, v)|/2\) \(
   t = \beta t, \quad u_t = u + t\Delta u_{nt}, \quad v_t = v + t\Delta v_{nt}.
   \]
EndWhile
4. Update: \(u = u + t\Delta u_{nt}, \quad v = v + t\Delta v_{nt}\).
   until \(|r(u, v)|/2 \leq \epsilon\)

However, the problem at hand is slightly more complicated since it involves a linear matrix inequality. Therefore, we use the barrier method [14]. Intuitively, a sequence of unconstrained minimization problems is solved, using the last point iteration as the starting point for the next iteration. Define for \(t > 0\), the cost function \(f_t(X, Q)\) by
\[
f_t(X, Q) = f(X, Q) + \log \det(-f_1(Q)) \quad (21)
\]
where \(f(X, Q)\) as defined in (14) and
\[
f_1(Q) = \Sigma_{yy}^{-1/2} Q^T \Sigma_{xx}^{-1} Q \Sigma_{yy}^{-1/2} - I \quad (22)
\]
Intuitively, $\frac{1}{t}f_t$ approaches $f$ as $t \to \infty$. Note that $f_t(X, Q)$ is still convex-concave for $t > 0$. The optimality conditions for a fixed $t > 0$ are given by

$$\nabla_X f_t(X^*, Q^*) = 0, \quad \nabla_Q f_t(X^*, Q^*) = 0 \quad (23)$$

where explicit expressions for $\nabla_X f_t$ and $\nabla_Q f_t$ are presented in Appendix IX-A along with the linear equations for computing $A_{X,t}, \Delta Q_{at}$. Finally, the structure of the problem allows us to easily identify a strictly feasible initial point $(X_0, Q_0)$ where $Q_0 = 0$ and $X_0$ satisfies

$$(C\Sigma_{xx}C^T + D\Sigma_{yy}D^T + \Sigma_\eta)X_0 = C\Sigma_{xx} \quad (24)$$

For details on the convergence of the infeasible start Newton method and the barrier method for convex-concave games, we refer the reader to [15], [14].

**Remark 3:** Notation: $Df(x)[h]$ denotes the (Fréchet) derivative or differential of $f$ at $x$ along $h$. Similarly, $Df(x, y)[h_x, h_y]$ denotes the differential of $f$ at $(x, y)$ along $(h_x, h_y)$.

**VI. NUMERICAL EXAMPLES**

In this section, we present two numerical examples which shed light on the differences between the Covariance Intersection (CI) and the proposed Robust Fusion (RF). First, consider the example of fusing two random variables with means $\bar{x} = \bar{y} = [0 \ 0]^T$ and covariances

$$\Sigma_{xx} = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}, \quad \Sigma_{yy} = \begin{bmatrix} 3 & 0 \\ 0 & 7 \end{bmatrix} \quad (25)$$

Let $(\bar{z}_{CI}, \Sigma_{CI})$ and $(\bar{z}_{RF}, \Sigma_{RF})$ be the fused estimates and the corresponding error covariances obtained from Covariance Intersection and Robust Fusion. We have that $\bar{z}_{CI} = \bar{z}_{RF} = [0 \ 0]^T$ and

$$\Sigma_{CI} = \begin{bmatrix} 3.79 & 0 \\ 0 & 5.79 \end{bmatrix}, \quad \Sigma_{RF} = \begin{bmatrix} 3 & 0 \\ 0 & 5 \end{bmatrix} \quad (26)$$

In the second example, we consider the case of partial measurements. More specifically, using notation of Section IV, let

$$\bar{x} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \Sigma_{xx} = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix} \quad (27)$$

and $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$, $z = \bar{z} = 0$, $\Sigma_{yy} = 1$, $D = 1$ and $\Sigma_\eta = 0$. Both Covariance Intersection and Robust Fusion yield $\bar{z}^T = 0$ but

$$\Sigma_{CI} = \begin{bmatrix} 3 & 0 \\ 0 & 6 \end{bmatrix}, \quad \Sigma_{RF} = \begin{bmatrix} 1 & 0 \\ 0 & 5 \end{bmatrix} \quad (28)$$

Observe that despite we have a measurement of only the first coordinate, the error variance of the second coordinate increased. The reason for this phenomenon is that the CI updates the current estimate and the associated error covariance along a predefined direction only. Although $\text{tr}(\Sigma_{CI}) < \text{tr}(\Sigma_{xx})$, the bound on the true error covariance estimated by Covariance Intersection is very conservative.

**VII. SIMULATIONS**

Finally, we consider an application in distributed state estimation using relative position measurements. We experiment with a group of $n = 4$ agents on the plane with a communication network topology as depicted in Fig. 3. If there is an edge from $i$ to $j$, then agent $i$ transmits its current state estimate and the corresponding error covariance estimate to agent $j$ which upon receipt, takes a measurement of the relative position and updates its own state estimate and associated error covariance estimate. All agents have identical dynamics described by

$$\begin{bmatrix} x_i(t+1) \\ v_i(t+1) \end{bmatrix} = \begin{bmatrix} I & I \\ 0 & I \end{bmatrix} \begin{bmatrix} x_i(t) \\ v_i(t) \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} w_i(t) \quad (29)$$

where $x_i(t) \in \mathbb{R}^2$ and $v_i(t) \in \mathbb{R}^2$ denote respectively the position and velocity of agent $i$ at time instance $t$ and $w_i(t) \sim \mathcal{N}(0, Q_i(t))$ is the process noise. If $x_i(t) \in \mathbb{R}^2$ and $v_i(t) \in \mathbb{R}^2$ denote respectively the position and velocity of agent $i$ at time instance $t$ and $w_i(t) \sim \mathcal{N}(0, Q_i(t))$ is the process noise. If $x_i(t) \in \mathbb{R}^2$ and $v_i(t) \in \mathbb{R}^2$ denote respectively the position and velocity of agent $i$ at time instance $t$ and $w_i(t) \sim \mathcal{N}(0, Q_i(t))$ is the process noise. If $x_i(t) \in \mathbb{R}^2$ and $v_i(t) \in \mathbb{R}^2$ denote respectively the position and velocity of agent $i$ at time instance $t$ and $w_i(t) \sim \mathcal{N}(0, Q_i(t))$ is the process noise.
Fig. 3. Network topology.

\[
\begin{bmatrix}
 x_i(t)^T \\
 v_i(t)^T
\end{bmatrix}^T, \quad \text{then let } A, B \text{ such that }
\]
\[
x_i(t+1) = Ax_i(t) + Bv_i(t)
\quad (30)
\]

Only agent 1 is equipped with global position system (GPS), that is we have a measurement of the form
\[
y_1(t) = x_1(t) + \eta_1(t)
\quad (31)
\]
where \( \eta_1(t) \sim \mathcal{N}(0, R_1(t)) \). Agent 1 performs a standard Kalman Filter update step after a GPS measurement. For each edge \((i, j)\) we have a pairwise measurement of the form
\[
y_{ij}(t) = x_j(t) - x_i(t) + \eta_{ij}(t)
\quad (32)
\]
where \( \eta_{ij}(t) \sim \mathcal{N}(0, R_{ij}(t)) \). Updates of the state estimates can be performed by either ignoring cross-correlations (Naive Fusion) or by one of Covariance Intersection or the proposed Robust Fusion.

Each agent maintains only its one state and communicates its to each neighbors at each time instance. The individual prediction step is the same as the Kalman Filter (KF) prediction step, that is
\[
\bar{x}_i(t+1|t) = \hat{A} \bar{x}_i(t|t)
\quad (33)
\]
\[
\tilde{\Sigma}_i(t+1|t) = \hat{A} \tilde{\Sigma}_i(t|t) \hat{A}^T + \hat{B} \Sigma_q \hat{B}^T
\quad (34)
\]
where \( \bar{x}_i(t+1|t) \) denotes the estimate of agent \( i \) for its state at time \( t+1 \) having received measurements up to time \( t \) and \( \tilde{\Sigma}_i \) is the associated error covariance.

We evaluate four estimator, three decentralized and one centralized: Naive Fusion (NF) which ignores cross-correlations, Robust Fusion (RF), Covariance Intersection (CI) and Centralized Kalman Filter (CKF). The Centralized Kalman Filter (CKF) is simply a standard Kalman Filter containing all agent states. It serves as a measure of how close the decentralized estimators are to the optimal centralized estimator. Results can be seen in Fig. 4 and Table I. We used the following values for the noise parameters: \( Q_i = 10^{-6} I_2 \) for all agents, \( R_1 = I_2 \) and \( R_{ij} = 10^{-2} I_2 \) for all pairwise measurements. The Robust Fusion based estimator significantly outperforms the Covariance Intersection based estimator which produces particularly noisy velocity estimates.

VIII. CONCLUSIONS AND FUTURE WORK

This work addressed the problem of fusing two random vectors with unknown cross-correlations by proposing a minimax formulation. The formulation was extended to handle linear measurement models. The problem of decentralized state estimation from relative position measurements was presented as an application. In the future, we plan to explore potential applications in decentralized multi-robot SLAM settings [17].

IX. APPENDIX

A. Formulas for computing the Newton steps

First of all, the differential of \( f_1(Q) \) at the direction of \( \Delta Q \) is given by
\[
D f_1(Q)[\Delta Q] = \Sigma_{yy}^{-1/2} (\Delta Q T \Sigma_{xx}^{-1} Q + Q T \Sigma_{xx}^{-1} \Delta Q) \Sigma_{yy}^{-1/2}
\quad (35)
\]
For small \( \Delta X \), we have the first order approximation [14]:
\[
\log \det (X + \Delta X) \approx \log \det (X) + \text{tr}(X^{-1} \Delta X)
\quad (36)
\]
and thus, using the chain rule, we obtain
\[
\nabla_Q \log \det (-f_1(Q)) = 2 \Sigma_{xx}^{-1} Q \Sigma_{yy}^{-1/2} f_1(Q)^{-1} \Sigma_{yy}^{-1/2}
\quad (37)
\]
Moreover, we have
\[
\nabla_X f(X, Q) = 2 \left( \begin{bmatrix} C & D \\ Q T & \Sigma_{yy} \end{bmatrix} \right) \left( \begin{bmatrix} C T \\ \Sigma_{yy} \end{bmatrix} + \Sigma_q \right) X
\]
\[
-2 \left( C \Sigma_{xx} + D Q T \right)
\quad (38)
\]
\[
\nabla_Q f(X, Q) = 2 (C T X X T D - X T D)
\quad (39)
\]
Let \( g_1(Q) = \Sigma_{yy}^{-1/2} f_1(Q)^{-1} \Sigma_{yy}^{-1/2} \). Using
\[
(X + \Delta X)^{-1} \approx X^{-1} - X^{-1} \Delta X X^{-1}
\quad (40)
\]
for small \( \Delta X \) and the chain rule, we obtain the following system of linear equations for \( \Delta X_{nt}, \Delta Q_{nt} \):
\[
2t \left( \begin{bmatrix} C & D \\ Q T & \Sigma_{yy} \end{bmatrix} \right) \left( \begin{bmatrix} C T \\ \Sigma_{yy} \end{bmatrix} + \Sigma_q \right) \Delta X_{nt}
\]
\[
+ 2t (C \Delta Q_{nt} D T X - D \Delta Q_{nt} T (I - C T X)) = -\nabla_X f_1(X, Q)
\quad (41)
\]
and
\[
2t (C T \Delta X_{nt} X T D - (I - C T X) \Delta X_{nt} D) + 2 \Sigma_{xx}^{-1} Q \eta_1(Q) \left( \Delta Q_{nt} T \Sigma_{xx}^{-1} Q + Q T \Sigma_{xx}^{-1} \Delta Q_{nt} \right) \eta_1(Q)
\]
\[
+ 2 \Sigma_{xx}^{-1} \Delta Q_{nt} \eta_1(Q) = -\nabla_Q f_1(X, Q)
\quad (42)
\]

| Agent # | CKF     | RF     | CI     |
|--------|---------|--------|--------|
| 1      | 0.174 ± 0.109 m 0.222 ± 0.091 m 0.250 ± 0.090 m 0.280 ± 0.094 m 0.286 ± 0.145 m 0.343 ± 0.187 m 0.285 ± 0.128 m 0.300 ± 0.152 m | 23.8 ± 2.4 m 248 ± 22 m 238 ± 18 m | 238 ± 18 m 238 ± 18 m 238 ± 18 m |
| 2      | 0.166 ± 0.098 m 0.248 ± 0.108 m 0.286 ± 0.145 m 0.343 ± 0.187 m 0.285 ± 0.128 m 0.300 ± 0.152 m | 23.8 ± 2.4 m 248 ± 22 m 238 ± 18 m | 238 ± 18 m 238 ± 18 m 238 ± 18 m |
| 3      | 0.170 ± 0.099 m 0.248 ± 0.114 m 0.343 ± 0.187 m 0.285 ± 0.128 m 0.300 ± 0.152 m | 23.8 ± 2.4 m 248 ± 22 m 238 ± 18 m | 238 ± 18 m 238 ± 18 m 238 ± 18 m |
| 4      | 0.161 ± 0.085 m 0.258 ± 0.090 m 0.285 ± 0.128 m 0.300 ± 0.152 m | 23.8 ± 2.4 m 248 ± 22 m 238 ± 18 m | 238 ± 18 m 238 ± 18 m 238 ± 18 m |
Fig. 4. Comparison of the four methods. The Naive Fusion estimator quickly diverges, whereas the Robust Fusion and Covariance Intersection do not. Clearly the Robust Fusion estimator is more accurate than the Covariance Intersection and produces less noisy estimates due to its less conservative nature. The steady state error covariance estimate of the Covariance Intersection is much larger than the actual error covariance making the estimator more susceptible to measurement noise.

REFERENCES

[1] R. E. Kalman, “A new approach to linear filtering and prediction problems,” Journal of basic Engineering, vol. 82, no. 1, pp. 35–45, 1960.
[2] H. W. Sorenson, Kalman filtering: theory and application. IEEE, 1985.
[3] S. J. Julier and J. K. Uhlmann, “New extension of the kalman filter to nonlinear systems,” in AeroSense’97. International Society for Optics and Photonics, 1997, pp. 182–193.
[4] S. I. Roumeliotis and G. A. Bekey, “Distributed multirobot localization,” IEEE Transactions on Robotics and Automation, vol. 18, no. 5, pp. 781–795, 2002.
[5] R. Tron and R. Vidal, “Distributed 3-d localization of camera sensor networks from 2-d image measurements,” IEEE Transactions on Automatic Control, vol. 59, no. 12, pp. 3325–3340, 2014.
[6] S. J. Julier and J. K. Uhlmann, “A non-divergent estimation algorithm in the presence of unknown correlations,” in Proceedings of the IEEE American Control Conference, vol. 4, 1997, pp. 2369–2373.
[7] P. O. Arambel, C. Rago, and R. K. Mehra, “Covariance intersection algorithm for distributed spacecraft state estimation,” in Proceedings of the IEEE American Control Conference, vol. 6, 2001, pp. 4398–4403.
[8] X. Xu and S. Negahdaripour, “Application of extended covariance intersection principle for mosaic-based optical positioning and navigation of underwater vehicles,” in Proceedings of the IEEE International Conference on Robotics and Automation, vol. 3, 2001, pp. 2759–2766.
[9] E. D. Nerurkar and S. I. Roumeliotis, “Power-slam: A linear-complexity, consistent algorithm for slam,” in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007, pp. 636–643.
[10] S. Panzieri, F. Pascucci, and R. Setola, “Multirobot localization using interfaced extended kalman filter,” in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 2816–2821.
[11] A. Martinelli, “Improving the precision on multi robot localization by using a series of filters hierarchically distributed,” in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007, pp. 1053–1058.
[12] H. Li and F. Nishashibi, “Cooperative multi-vehicle localization using split covariance intersection filter,” IEEE Intelligent Transportation Systems Magazine, vol. 5, no. 2, pp. 33–44, 2013.
[13] L. C. Carrillo-Arce, E. D. Nerurkar, J. L. Gordillo, and S. I. Roumeliotis, “Decentralized multi-robot cooperative localization using covariance intersection,” in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2013, pp. 1412–1417.
[14] S. Boyd and L. Vandenberghe, Convex optimization. Cambridge university press, 2004.
[15] A. Ghosh and S. Boyd, “Minimax and convex-concave games,” lecture notes for course EE392: ‘Optimization Projects” Stanford Univ., Stanford, CA, 2003.
[16] S. Leonardos and K. Daniilidis, “A game-theoretic approach to robust fusion and kalman filtering under unknown correlations,” arXiv preprint arXiv:1610.01045, 2016.
[17] X. S. Zhou and S. I. Roumeliotis, “Multi-robot slam with unknown initial correspondence: The robot rendezvous case,” in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 1785–1792.