Convex Combination and Covariance Intersection
Algorithms in Distributed Fusion

Chee-Yee Chong
Booz Allen & Hamilton, Inc.
101 California Street, Suite 3300
San Francisco, CA 94111, USA
chong_chee@bah.com  cychong@alum.mit.edu

Shozo Mori
Information Extraction and Transport, Inc.
1911 N. Fort Myer Dr., Suite 600
Arlington, VA 22209, USA
smori@iet.com

Abstract - In a distributed estimation or tracking system, local estimates are first generated from individual sensors. The state estimates of associated objects are then fused to generate the global estimates. The fusion algorithm has to deal with correlated estimation errors due to common past information or common process noise. Most approaches to estimation fusion use a convex combination of the local estimates to minimize the mean square error. Recently an alternative approach to fusion, covariance intersection, has been proposed and is claimed to be more robust. This paper provides a set-theoretic interpretation of the covariance intersection approach and develops a tighter bound for the estimation error. Numerical results are used to compare the performance of the different fusion approaches, and the probability for the bound to be true is computed for some examples. Critical parameters that affect the performance of a fusion algorithm are identified.

Keywords: Track fusion, estimation fusion, distributed fusion, distributed estimation, convex combination, covariance intersection, set theoretic estimation

1 Introduction

Modern estimation or tracking systems usually utilize multiple sensors to take advantage of multiple viewing angles, complementary phenomenology, etc. Although centralized or measurement fusion can theoretically provide the best estimation performance, processing all sensor measurements at a single location is sometimes not feasible due to communication or reliability constraints. In a distributed fusion system, each sensor processes its sensor measurements and communicates the results with other sensors or processors. When tracking is the problem, the sensors will form sensor tracks from the measurements and communicate the tracks with other sensors or processors. The tracks are then associated and the estimates of associated tracks are fused. Several distributed fusion architectures were discussed in [1]-[4], ranging from hierarchical architecture with or without feedback to a general distributed architecture.

Distributed fusion has the advantages of lower communication requirements, improved robustness, etc. as compared to centralized fusion. However, the fusion algorithms have to deal with issues that are not present in centralized or measurement fusion. In particular, the state estimates to be fused cannot be treated like sensor measurements and fused with a standard centralized algorithm such as Kalman filter. This is due to the fact that while sensor measurement errors are usually independent across sensors and time, the errors in state estimates are generally correlated. The sources of correlation may be common process noise in the target when the state estimates are not fused at each sampling instant or common prior information in the estimates from previous communication.

Much research has been performed over the last two decades on distributed fusion [1]-[21]. There are two basic approaches to developing optimal distributed fusion algorithms. The first is finding the “best” linear combination of estimates to optimize some criteria, e.g., weighted least squares or minimum variance [7], [8]. If the estimates to be fused are not the sufficient statistics, the optimal linear combination may not be as optimal as the centralized estimate. References [22]-[25] develop a unified model for estimation fusion based upon the best linear unbiased estimation (BLUE) or linear minimum variance approach.

The second approach attempts to reconstruct the optimal centralized estimate (by fusing the measurements directly) from the estimates [1]-[6], [9]-[21]. In particular, the information graph approach [1]-[4] has been used to identify additional information needed for complicated fusion architectures. Algorithms for fusing both means and covariances for linear systems and general probability distributions have been developed. This fusion algorithm focuses on de-correlation of common prior information, and is not particularly effective in handling correlation in measurement noise (which may result from common process noise accumulated between fusion instants). It is also a linear combination of estimates when means and covariance’s are given.

Recently, the covariance intersection filter [26]-[29] has been proposed to fuse estimates without assuming any knowledge on the correlation between the estimates to be fused. It is supposed to be more robust than the linear com-
2 Distributed Fusion Problem

Suppose the state to be estimated is a finite dimensional random vector \( \mathbf{x} \) with mean \( \mathbf{\mu} \) and covariance \( \mathbf{P} \). Suppose the two estimates to be fused are \( \mathbf{\hat{x}}_i, i=1, 2 \), with error covariance’s given by \( \mathbf{P}_y = \mathbb{E}[(\mathbf{x} - \mathbf{\hat{x}}_i)(\mathbf{x} - \mathbf{\hat{x}}_j)'] \) for \( i=1, 2 \), and \( j=1, 2 \). The distributed fusion problem is to generate an “optimal” estimate \( \mathbf{\hat{x}} \) from the estimates \( \mathbf{\hat{x}}_1 \) and \( \mathbf{\hat{x}}_2 \).

There are three possible architectures depending on the sources of \( \mathbf{\hat{x}}_1 \) and \( \mathbf{\hat{x}}_2 \).

- **Hierarchical fusion with no memory at fusion site**: \( \mathbf{\hat{x}}_1 \) and \( \mathbf{\hat{x}}_2 \) are estimates from the sensors, \( \mathbf{\mu} \) is the common prior mean used by the sensors or the information provided in the most recent feedback from fusion.

- **Hierarchical fusion with memory at fusion site**: \( \mathbf{\hat{x}}_1 \) is the most current estimate at the fusion site, \( \mathbf{\hat{x}}_2 \) is the sensor estimate to be fused, and \( \mathbf{\mu} \) is latest feedback from the fusion site to the sensor or the previous communication from the sensor to the fusion (extrapolated if necessary).

- **Arbitrary distributed fusion**: \( \mathbf{\hat{x}}_1 \) and \( \mathbf{\hat{x}}_2 \) are any two arbitrary estimates to be fused, and \( \mathbf{\mu} \) is the common prior information shared by the estimates.

The main problem in distributed fusion is caused by correlated estimation errors. In general \( \mathbf{P}_{12} = \mathbf{P}_{21} \neq 0 \), and their values may not even be known. Suppose \( \mathbf{\hat{x}}_i \) is the linear minimum variance estimate form the observation equation

\[
\mathbf{z}_i = \mathbf{H}_i \mathbf{x} + \mathbf{v}_i
\]

where \( \mathbf{v}_i \) is zero mean with \( \mathbb{E}[\mathbf{v}_i\mathbf{v}_j'] = \mathbf{0}_y \) and \( \mathbf{H}_i \) is the observation matrix. Then it can be shown that

\[
\mathbf{P}_y = \mathbf{P}_n \mathbf{P}^{-1} \mathbf{P}_y + \mathbf{K}_i \mathbf{R}_i \mathbf{K}_i'
\]

where \( \mathbf{K}_i = \mathbf{H}_i' \mathbf{P}^{-1} \mathbf{H}_i' + \mathbf{R}_i \). The correlated measurement errors may be actual correlated noises or common process noise accumulated over multiple times. When \( \mathbf{R}_y = \mathbf{0}, \mathbf{P}_y = \mathbf{P}_n \mathbf{P}^{-1} \mathbf{P}_y \). If furthermore, there is no prior information, i.e., \( \mathbf{P}^{-1} = \mathbf{0} \), then \( \mathbf{P}_y = \mathbf{0} \).

3. Distributed Fusion Algorithms

The following are the main approaches to fusion and their underlying assumptions. To simplify notation, we use \( \mathbf{P}_i \) to represent \( \mathbf{P}_{yi} \).

3.1 Simple Convex Combination

When the cross covariance between the two estimates can be ignored, the fusion algorithm is given by:

- **State estimate**:
  \[
  \mathbf{\hat{x}}_{cc} = \mathbf{P}_2 (\mathbf{P}_1 + \mathbf{P}_2)^{-1} \mathbf{\hat{x}}_1 + \mathbf{P}_1 (\mathbf{P}_1 + \mathbf{P}_2)^{-1} \mathbf{\hat{x}}_2
  \]

- **Error covariance**:
  \[
  \mathbf{P}_{cc} = \mathbf{P}_1 (\mathbf{P}_1 + \mathbf{P}_2)^{-1} \mathbf{P}_1 = \mathbf{P}_1 (\mathbf{P}_1 + \mathbf{P}_2)^{-1} \mathbf{P}_2
  \]

This simple convex combination algorithm has been used extensively because of its simple implementation. It is optimal if \( \mathbf{P}_y = \mathbf{0} \), i.e., \( \mathbf{P}^{-1} = \mathbf{0} \) and \( \mathbf{R}_y = \mathbf{0} \). Otherwise it is only sub-optimal and \( \mathbf{P}_{cc} \) is not the actual error covariance.

3.2 Bar-Shalom/ Campo Combination

When the prior information can be ignored, i.e., \( \mathbf{P}^{-1} = \mathbf{0} \), but the measurement noises are correlated, i.e., \( \mathbf{R}_y \neq \mathbf{0} \), then \( \mathbf{P}_{12} = \mathbf{P}_{21} \neq \mathbf{0} \). The following algorithm can be used to combine the estimates [8]:

- **State estimate**:
  \[
  \mathbf{\hat{x}}_{bc} = \mathbf{\hat{x}}_1 + (\mathbf{P}_1 - \mathbf{P}_1^2) (\mathbf{P}_1 + \mathbf{P}_2 - \mathbf{P}_2^2)^{-1} (\mathbf{\hat{x}}_2 - \mathbf{\hat{x}}_1)
  \]

- **Error covariance**:
  \[
  \mathbf{P}_{bc} = \mathbf{P}_1 - (\mathbf{P}_1 - \mathbf{P}_1^2) (\mathbf{P}_1 + \mathbf{P}_2 - \mathbf{P}_2^2)^{-1} (\mathbf{P}_1 - \mathbf{P}_2)
  \]

The cross-covariance’s \( \mathbf{P}_{12} \) and \( \mathbf{P}_{21} \) are computed from the observation matrices and the Kalman filter gains according to (2).

This algorithm can be viewed as a special case of the best least unbiased estimate (BLUE) or linear minimum variance estimate when the prior is ignored.

3.3 Information De-correlation

When \( \mathbf{R}_y = \mathbf{0} \) but \( \mathbf{P}^{-1} \neq \mathbf{0} \), the state estimate fusion algorithm is given by:

- **State estimate**:
  \[
  \mathbf{\hat{x}}_{id} = \mathbf{P}_n (\mathbf{P}_1^{-1} \mathbf{\hat{x}}_1 + \mathbf{P}_2^{-1} \mathbf{\hat{x}}_2 - \mathbf{P}^{-1} \mathbf{\hat{x}})
  \]

- **Error covariance**:
\[ P_{id} = (P_1^{-1} + P_2^{-1} - \overline{P})^{-1} \] (8)

This fusion algorithm is based on the concept of decorrelation and was first presented in [5]. It can be extended to handle arbitrary fusion architectures as long as the correlation in the estimation errors is due to prior information, and the communication history is known. There is also a nonlinear version of this algorithm that can be used to fuse probability distributions [4]. Note that even though the measurement errors are un-correlated, the prior information causes the estimation errors in \( \hat{x}_1 \) and \( \hat{x}_2 \) to be correlated, i.e., \( P_y = P_1 \overline{P}^{-1} P_y \neq 0 \). The actual estimation error covariance is generally different from \( P_{id} \).

### 3.4 Linear Minimum Variance Estimate

Let \( \xi = [\hat{x}_1, \hat{x}_2]' \). Then the linear minimum variance fusion algorithm, also called the best linear unbiased estimate (BLUE), [22]-[25], is given by:

- **State estimate:**
  \[ \hat{x}_{LMV} = \overline{x} + W_{LMV1}(\hat{x}_1 - \overline{x}) + W_{LMV2}(\hat{x}_2 - \overline{x}) \] (9)
  where the weights \( W_{LMV1} \) and \( W_{LMV2} \) are given by
  \[ W_{LMV} = [W_{LMV1} W_{LMV2}] = V_{\xi\xi}^{-1} \] (10)
- **Error covariance:**
  \[ P_{LMV} = \overline{P} - V_{\xi\xi}^{-1} V_{\xi\xi}' \] (11)

The matrix \( V_{\xi\xi} \) is the covariance between \( x \) and \( \xi \), and \( V_{\xi\xi}' \) is the covariance of \( \xi \). It can be shown that

\[ V_{\xi\xi} = \begin{bmatrix} \overline{P} - P_{11} & \overline{P} - P_{12} & \overline{P} - P_{21} & \overline{P} - P_{22} \end{bmatrix} \] (12)

\[ V_{\xi\xi}' = \begin{bmatrix} \overline{P} - P_{11} & \overline{P} - P_{12} & \overline{P} - P_{21} & \overline{P} - P_{22} \end{bmatrix} \] (13)

The three estimates discussed earlier are special cases of this estimate with the appropriate assumptions. When all the random variables are Gaussian, this estimate is also the maximum a posteriori (MAP) estimate. The error covariance \( P_{LMV} \) gives the actual error covariance.

### 3.5 Covariance Intersection

The covariance intersection algorithm is given by:
- **State estimate:**
  \[ \hat{x}_{CI} = \omega P_1^{-1} \hat{x}_1 + (1 - \omega) P_2^{-1} \hat{x}_2 \] (14)
  or
  \[ \hat{x}_{CI} = P_{CI} [\omega P_1^{-1} \hat{x}_1 + (1 - \omega) P_2^{-1} \hat{x}_2] \] (15)
- **Error covariance:**
  \[ P_{CI} = (\omega P_1^{-1} + (1 - \omega) P_2^{-1})^{-1} \] (16)

where \( \omega \) is a number between 0 and 1. The covariance intersection algorithm is motivated by the need to provide a conservative and robust estimate when the correlation between \( \hat{x}_1 \) and \( \hat{x}_2 \) is unknown, e.g., in a general distributed fusion system. In that case, the optimal fusion algorithm may be computational infeasible [32]. The actual error covariance using this filter is bounded by \( P_{CI} \).

### 4. Set-Theoretic Interpretation of Covariance Intersection

The covariance intersection fusion algorithm has the same form as the simple convex combination algorithm except that the error covariance \( P_i \)'s are inflated. More specifically,

\[ P_1 \rightarrow \omega P_1 \] (17)
\[ P_2 \rightarrow (1 - \omega) P_2 \] (18)

Thus the estimated covariance is conservative compared to the convex combination algorithm. It was shown in [] that \( P_{CI} \) provides a bound for the actual estimation error covariance. The fused estimate itself depends on the specific weight \( \omega \) used. When \( \omega = 0.5 \), the estimate is identical to the simple convex combination estimate, i.e.,

\[ \hat{x}_{CI} = (P_1^{-1} + P_2^{-1} - \overline{P})^{-1}$ [P_1^{-1} \hat{x}_1 + P_2^{-1} \hat{x}_2] = \hat{x}_{CC} \] (19)

but the error covariance is twice that of the error covariance of the convex combination algorithm, i.e.,

\[ P_{CI} = 2(P_1^{-1} + P_2^{-1} - \overline{P})^{-1} = 2P_{CC} \] (20)

In general, the weight \( \omega \) is chosen to minimize some function, e.g., the determinant or trace of \( P_{CI} \). When \( P_1 = P_2 \), then independent of the value of \( \omega \),

\[ P_{CI} = P_1 = P_2 \] (21)
\[ \hat{x}_{CI} = \omega \hat{x}_1 + (1 - \omega) \hat{x}_2 \] (22)

Thus the error bound does not represent any possible reduction in uncertainty in the fused estimate. This is true no matter how many estimates are being fused, and is somewhat counterintuitive since one generally expects a reduction in uncertainty with more measurements.

### 4.1 Set Theoretic Estimation

We now provide an alternative derivation of the covariance intersection filter based on estimation using sets. This theory of estimation using the unknown but bounded approach was developed by Schweppe [30], [31] in the late sixties. This set-theoretic derivation provides a natural interpretation of the properties of the covariance intersection filter.

Suppose the state \( x \) is known to be in the ellipsoids \( \Omega(\hat{x}_1; P_1) \) and \( \Omega(\hat{x}_2; P_2) \) where

\[ \Omega(\hat{x}_i; P_i) = \{x : (x - \hat{x}_i)' P_i^{-1} (x - \hat{x}_i) \leq 1\} \] (23)
The centers of the ellipsoids, \( \hat{x}_1 \) and \( \hat{x}_2 \), can be viewed as the estimates for \( x \), and the matrices \( P_1 \) and \( P_2 \), represent the uncertainty in the estimates. When both estimates are given, the feasible state lies in the intersection of the two sets, \( \Omega(\hat{x}_1;P_1) \cap \Omega(\hat{x}_2;P_2) \), which is not an ellipsoid. However, we can find a bounding ellipsoid for this intersection as

\[
\Omega(\omega) = \{ x : \omega(x - \hat{x}_1) P_1^{-1}(x - \hat{x}_1) + (1 - \omega)(x - \hat{x}_2) P_2^{-1}(x - \hat{x}_2) \leq 1 \}
\]

where \( 0 \leq \omega \leq 1 \).

We can show that \( \Omega(\hat{x}_1;P_1) \cap \Omega(\hat{x}_2;P_2) \subset \Omega(\omega) \) since

\[
(x - \hat{x}_1) P_1^{-1}(x - \hat{x}_1) \leq 1
\]

and

\[
(x - \hat{x}_2) P_2^{-1}(x - \hat{x}_2) \leq 1
\]

implies that

\[
\omega(x - \hat{x}_1) P_1^{-1}(x - \hat{x}_1) + (1 - \omega)(x - \hat{x}_2) P_2^{-1}(x - \hat{x}_2) \leq 1
\]

But

\[
\omega(x - \hat{x}_1) P_1^{-1}(x - \hat{x}_1) + (1 - \omega)(x - \hat{x}_2) P_2^{-1}(x - \hat{x}_2)
= \omega \hat{x}_1 P_1^{-1} x - 2 \omega \hat{x}_1 P_1^{-1} x + \omega \hat{x}_1 P_1^{-1} \hat{x}_1
+ (1 - \omega)[x P_2^{-1} x - 2 x P_2^{-1} \hat{x}_2 + \hat{x}_2 P_2^{-1} \hat{x}_2]
= x[x P_1^{-1} + (1 - \omega) P_2^{-1} x - 2 x[x P_1^{-1} \hat{x}_1 + (1 - \omega) P_2^{-1} \hat{x}_2]
+ \omega \hat{x}_1 P_1^{-1} \hat{x}_1 + (1 - \omega) \hat{x}_2 P_2^{-1} \hat{x}_2
= x P_1^{-1} x - 2 x P_2^{-1} \hat{x}_2 + \omega \hat{x}_1 P_1^{-1} \hat{x}_1 + (1 - \omega) \hat{x}_2 P_2^{-1} \hat{x}_2
\]

The bounding ellipsoid is then given by:

\[
x P_1^{-1} x - 2 x P_2^{-1} \hat{x}_2 + \omega \hat{x}_1 P_1^{-1} \hat{x}_1 + (1 - \omega) \hat{x}_2 P_2^{-1} \hat{x}_2 \leq 1
\]

or, by completing squares

\[
(x - \hat{x}_1) P_1^{-1}(x - \hat{x}_1) - \hat{x}_1 P_1^{-1} \hat{x}_1 + (1 - \omega) \hat{x}_2 P_2^{-1} \hat{x}_2 \leq 1
\]

Using the lemma in the appendix, we can show that

\[
\omega \hat{x}_1 P_1^{-1} \hat{x}_1 + (1 - \omega) \hat{x}_2 P_2^{-1} \hat{x}_2 - \hat{x}_1 P_1^{-1} \hat{x}_1
= (\hat{x}_1 - \hat{x}_2)(\omega P_1^{-1} + (1 - \omega) P_2^{-1})^{-1}(\hat{x}_1 - \hat{x}_2)
\]

Then

\[
(x - \hat{x}_1) P_1^{-1}(x - \hat{x}_1) \leq 1 - (\omega \hat{x}_1 P_1^{-1} \hat{x}_1
+ (1 - \omega) \hat{x}_2 P_2^{-1} \hat{x}_2 - \hat{x}_1 P_1^{-1} \hat{x}_1)
= 1 - \omega^2
\]

where \( \omega^2 = (\hat{x}_1 - \hat{x}_2)(\omega P_1^{-1} + (1 - \omega) P_2^{-1})^{-1}(\hat{x}_1 - \hat{x}_2) \).

If \( \omega^2 < 1 \), which is true if the ellipsoids \( \Omega(\hat{x}_1;P_1) \) and \( \Omega(\hat{x}_2;P_2) \) touch, i.e., the ellipsoids are accurate representations of the uncertainty, then

\[
(x - \hat{x}_2) [(1 - \omega^2) P_2^2]^{-1}(x - \hat{x}_2) \leq 1
\]

defines the bounding ellipsoid. The center of the bounding ellipsoid \( \hat{x}_2 = P_2^2(\omega P_1^{-1} \hat{x}_1 + (1 - \omega) P_2^{-1} \hat{x}_2) \) can be viewed as the estimate of \( x \) and the matrix \( (1 - \omega^2) P_2^2 \) represents the uncertainty in the estimate. This is a tighter uncertainty characterization than \( P_2 \). However, \( \omega^2 \) depends on the estimates to be fused and cannot be computed \emph{a priori}. The matrix \( (1 - \omega^2) P_2^2 \) decreases when the estimates to be fused are far from each other. Figure 1 shows how the bounding ellipsoid depends on the distance between estimates to be fused.

![Figure 1: Intersecting Ellipsoids](image)

With the set theoretic interpretation, the state is estimated to lie inside the set, but there is no probability distribution over the set. Thus, one will not expect the uncertainty set to decrease with additional estimates are fused since the estimate can be viewed as uniformly distributed over the intersection set.

### 5 Numerical Results

Except for the linear minimum variance or BLUE algorithm, all the other algorithms are sub-optimal when \( P_2 = P_2^i \neq 0 \). Suppose the fusion algorithm is of the form

\[
\hat{x} = W_1 \tilde{x}_1 + W_2 \tilde{x}_2
\]

and the estimate is unbiased, i.e., \( W_1 + W_2 = I \), then the actual error covariance is given by

\[
P = \begin{bmatrix} W_1 & W_2 \end{bmatrix} P_{21} P_{22} W_1^T + W_2^T P_{22} W_1
\]

Similarly, for the estimator of the form

\[
\hat{x} = W_o \tilde{x}_1 + W_1 \tilde{x}_1 + W_2 \tilde{x}_2
\]

and \( W_o + W_1 + W_2 = I \), the actual error covariance matrix is
\[ P = \begin{bmatrix} W_0 & W_1 & W_2 \end{bmatrix} \begin{bmatrix} \tilde{P} & P_{11} & P_{12} \\ P_{11} & P_{11} & P_{12} \\ P_{12} & P_{12} & P_{22} \end{bmatrix} \begin{bmatrix} W_0' \\ P_{11}' \\ P_{12}' \end{bmatrix} \]

In general the actual error covariance will be different from calculated error covariance. For example, for the simple convex combination algorithm, the actual error covariance is given by

\[ P = P_{cc} + P_{cc}(P_{11}^{-1}P_{12} + P_{21}^{-1}P_{22}^{-1}P_{11})P_{cc} \]

It is possible for the actual error to be less than the estimated error. This is contrary to the common belief that \( P_{cc} \) is optimistic.

### 5.1 Sensitivity to Prior and Correlation

The main parameters affecting the performance of the fusion algorithms are the prior covariance \( \bar{P} \) of the state \( P_0 \) and the cross-covariance between the estimation errors. However, these parameters cannot be specified independently since they depend on the underlying mechanism generating the estimates. To simplify the performance comparison, we assume a scalar static state \( x \) with two scalar measurements given by \( z_i = x + v_i \). The fusion problem depends on the following parameters:

- \( P_1 \)
- \( R_1 > P_1 \)
- \( P_2 < (P_1^{-1} - R_1^{-1})^{-1} \)
- \( \gamma \) such that \(-1 < \gamma < 1\)

Then the measurement noise cross-covariance is given by \( R_{12} = R_{21} = \sqrt{\gamma R_1 R_2} \), the prior state covariance \( \bar{P} \) is \( \bar{P}^{-1} = P_1^{-1} - R_1^{-1} \), and the cross covariance \( P_{12} \) between the estimation errors is given by (2).

Given fixed values for \( P_1, P_2 \) and \( R_1 \), we vary \( \gamma \) and compute the following for each algorithm:

- Actual error covariance as given by (35) and (37)
- Error covariance as estimated by algorithm

The estimated error covariance of the linear minimum variance estimate and the Bar-Shalom/Campo are the same as the actual error covariance. For this example, the linear minimum variance estimate also has the same performance as the centralized fusion estimate. The other algorithms, simple convex combination, information de-correlation, and covariance intersection, all have estimates error covariance’s that are constant over \( \gamma \).

The following figures show the performance of the fusion algorithms as a function of \( \gamma \). Each figure also plots \( P_{12} \) to show how performance depends on the correlation between the estimation errors. We will use the following abbreviation for the algorithms:

- Simple convex combination – CC
- Bar-Shalom/Campo – BC
- Information de-correlation – ID
- Linear minimum variance – LMV
- Covariance intersection - CI

In general, the performance of CC, ID and CI deteriorates linearly with increasing correlation. ID and LMV have the same performance when \( \gamma = 0 \).

**Case 1 (Figure 2):** \( P_1 = 1, P_2 = 1, R_1 = 8, R_2 = 8, \bar{P} = 1.14, P_{cc} = 1, P_{CC} = 0.5, P_{ID} = 0.89 \)

Since \( P_1 = P_2 \), the weight \( \sigma \) does not affect \( P_{cc} \). With a value of 0.5, CI is identical to CC. Basically BC, CI and CC all have the same performance. Since \( \bar{P} \) is quite small, ID performs better than the other algorithms until \( \gamma \) becomes quite large. The linear minimum variance algorithm performs well for large negative correlation but no algorithm performs well for positive correlation.

**Case 2 (Figure 3):** \( P_1 = 1, P_2 = 1, R_1 = 2, R_2 = 2, \bar{P} = 2, P_{cc} = 1, P_{CC} = 0.5, P_{ID} = 0.667 \)

ID is even closer to LMV in performance while the other algorithms have the same performance.
Correlation Coefficient

Estimation Error Variance

Case 3 (Figure 4): \( P_1 = 1, \ P_2 = 1, \ R_1 = 1.113, \ R_2 = 1.113, \ \overline{P} = 9.85, \ P_{CI} = 1, \ P_{CC} = 0.5, \ P_{ID} = 0.53 \)

The large \( \overline{P} \) means that ID is similar to CI, CC and BC since there is no need to de-correlate the prior information. We also note that LMV does not perform much better than the other algorithms. This is probably due to identical error covariance’s in the estimation errors.

Case 4 (Figure 5): \( P_1 = 1, \ P_2 = 2, \ R_1 = 1.9, \ R_2 = 38, \ \overline{P} = 2.11, \ P_{CI} = 1, \ P_{CC} = 0.667, \ P_{ID} = 0.974 \)

Since \( P_1 < P_2 \), the weight is 1, i.e., CI only uses the estimate \( \hat{x}_1 \) and ignores \( \hat{x}_2 \) completely. However, its performance is better than CC all the time and better than ID for high correlation. This case also demonstrates that BC performs better than ID for high correlation, but worse than ID for low correlation.

Case 5 (Figure 6): \( P_1 = 1, \ P_2 = 2, \ R_1 = 1.113, \ R_2 = 2.51, \ \overline{P} = 9.85, \ P_{CI} = 1, \ P_{CC} = 0.667, \ P_{ID} = 0.715 \)

Again, a large \( \overline{P} \) means that ID does not have any significant advantage over CC. For negative values of \( \gamma \), all the algorithms except CI have similar performance. For high values of \( \gamma \), BC approaches the same performance as LMV, as expected.

The results for this simple example are consistent with the assumptions of the algorithms. When prior information is significant, ID performs well since it uses that information. When correlation is high but prior information is not significant, then BC has good performance. CI does not really offer any performance advantage over CC since the algorithm is basically the same as CC. However, it does provide a conservative bound for the actual error covariance.

5.2 Bound on Covariance Intersection Performance

In Section 4, we developed a tighter bound \((1 - \alpha^2)P_{CI}\) for the error performance of the covariance intersection filter. This bound is different from the standard bound by including the scaling factor \((1 - \alpha^2)\) which depends on the estimates to be fused and thus cannot be computed a priori. However, it is only valid when the uncertainty is modeled as unknown but bounded within a set. When the uncertainty is random, there is a finite probability that the error is bigger than \((1 - \alpha^2)P_{CI}\).

Suppose all the underlying random variables are Gaussian. The actual error covariance of the covariance intersection filter given \( \hat{x}_1 \) and \( \hat{x}_2 \) is

\[
\begin{align*}
V_{CI}(\hat{x}_1, \hat{x}_2) &= E[(x - \hat{x}_1)(x - \hat{x}_1)', (x - \hat{x}_2)(x - \hat{x}_2)'] \\
&= E[((x - \hat{x}_1) + (\hat{x} - \hat{x}_1))(x - \hat{x}_1) + (\hat{x} - \hat{x}_1)(x - \hat{x}_1)'] \hat{x}_1, \hat{x}_2 \\
&= E[((x - \hat{x})(x - \hat{x})') + (\hat{x} - \hat{x}_1)(\hat{x} - \hat{x}_1)'] \\
&= \overline{P} + (\hat{x} - \hat{x}_1)(\hat{x} - \hat{x}_1)' \\
&= \overline{P}
\end{align*}
\]

where \( \hat{x} \) and \( P \) are the linear minimum variance estimate or BLUE and the error covariance is defined in (11) to (13). Since \( \hat{x}_1 \) and \( \hat{x}_2 \) are random, \( V_{CI}(\hat{x}_1, \hat{x}_2) \) is also random.
Similarly, \((1 - \alpha^2)P_{ci}\) is also a random variable. If \((1 - \alpha^2)P_{ci}\) is a bound on the error, then
\[
(1 - \alpha^2)P_{ci} \geq P_0 + (\hat{x} - \hat{x}_{ci})(\hat{x} - \hat{x}_{ci})'
\] (41)

Since we are dealing with random variables, the inequality (41) is not always true. Thus, we will compute the probability
\[
P((1 - \alpha^2)P_{ci} \geq V_{ci}(\hat{x}_1, \hat{x}_2))
\] (42)

The probability of (42) can be manipulated into a probability of the form
\[
P\{w'Aw \leq 1\}
\] (43)
where \(w\) is a Gaussian random vector with known mean and covariance and \(A\) is a positive semi-definite matrix. In general, (43) has to evaluated numerically.

Figure 7 shows the probability (42) for the example analyzed in Section 5.1. Note that the inequality (41) or bound is satisfied with probability greater than 0.66 most of the time. Although the conclusion from this simple example may not be true in general cases, \((1 - \alpha^2)P_{ci}\) may indeed provide a tighter bound for the covariance intersection filter. More investigation in this area is needed.

### References


