Image Cover Sheet

CLASSIFICATION	SYSTEM NUMBER
UNCLASSIFIED	518345

TITLE

A Consistent Filter for Robust Decentralized Data Fusion

System Number:

Patron Number:

Requester:

Notes:

DSIS Use only:

Deliver to: CL

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
OCT 2001		

4. TITLE AND SUBTITLE
A Consistent Filter for Robust Decentralized Data Fusion

5a. CONTRACT NUMBER
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Defence R&D Canada - Valcartier,2459 Pie-XI Blvd North,Quebec (Quebec) G3J 1X5 Canada, ,

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)
11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The Situation Analysis Support Systems (SASS) Group in the Decision Support Systems (DSS) Section at Defence Research & Development Canada (DRDC) - Valcartier is currently investigating advanced concepts for adaptation and integration of the data fusion and sensor management processes. These concepts could apply to any current Canadian military platform's sensor suite, as well as its possible future upgrades, to improve its performance against the predicted future threat. The reported work addresses the problem of automatically aggregating information from multiple data sources. "Multiple Source Data Fusion" (MSDF) is used to indicate the general approach for combining the sensor data into global tracks. The selection of the appropriate MSDF techniques depends on the underlying architecture. For the centralized scheme, the sources are known to be independent and the Kalman filter provides an optimal solution. Unfortunately, when the decentralized architecture is used the sources become correlated and the Kalman filter cannot be applied. The covariance intersection method has been proposed as a solution to the problem of the decentralized data fusion, but results in a decrease in performance. A new fusion algorithm, that avoids both the inconsistency of the Kalman filter and the lack of performance of the covanance intersection, is proposed. The superiority of the proposed approach is illustrated through the target's tracking problem.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**71**	

THIS PAGE IS LEFT BLANK

THIS PAGE IS LEFT BLANK

A consistent filter for robust decentralized data fusion

Abder Rezak Benaskeur and Jean Roy
Decision Support Systems Section

Defence Research & Development Canada - Valcartier
Technical Report
TR 2001-223
2002-10-29

Author

A. Benaskeur and J. Roy

Approved by

Éloi Bossé
Head/DSS Section

REPRODUCTION QUALITY NOTICE

This document is the best quality available. The copy furnished to DRDCIM contained pages that may have the following quality problems:

: Pages smaller or Larger than normal

: Pages with background colour or light coloured printing

: Pages with small type or poor printing; and or

: Pages with continuous tone material or colour photographs

Due to various output media available these conditions may or may not cause poor legibility in the hardcopy output you receive.

☒ If this block is checked, the copy furnished to DRDCIM contained pages with colour printing, that when reproduced in Black and White, may change detail of the original copy.

Abstract (U)

The Situation Analysis Support Systems (SASS) Group in the Decision Support Systems (DSS) Section at Defence Research & Development Canada (DRDC) - Valcartier is currently investigating advanced concepts for adaptation and integration of the data fusion and sensor management processes. These concepts could apply to any current Canadian military platform's sensor suite, as well as its possible future upgrades, to improve its performance against the predicted future threat. The reported work addresses the problem of automatically aggregating information from multiple data sources. "Multiple Source Data Fusion" (MSDF) is used to indicate the general approach for combining the sensor data into global tracks. The selection of the appropriate MSDF techniques depends on the underlying architecture. For the centralized scheme, the sources are known to be independent and the Kalman filter provides an optimal solution. Unfortunately, when the decentralized architecture is used the sources become correlated and the Kalman filter cannot be applied. The covariance intersection method has been proposed as a solution to the problem of the decentralized data fusion, but results in a decrease in performance. A new fusion algorithm, that avoids both the inconsistency of the Kalman filter and the lack of performance of the covariance intersection, is proposed. The superiority of the proposed approach is illustrated through the target's tracking problem.

Résumé (U)

Des activités de recherche entreprises par le groupe systèmes d'aide à l'analyse de la situation (SAAS) de la section systèmes d'aide à la décision (SAD), à Recherche et Dèvelopment pour la Défence Canada (RDDC) - Valcartier, concernent l'étude des concepts avancés pour l'adaptation et l'intégration des processus de fusion des données et de gestion des capteurs. Ces concepts pourraient profiter au système de capteurs des plates-formes militaires canadiennes. Ce rapport aborde le problème de l'agrégation de l'information émanant de différentes sources. La "Fusion de Données Multi-Sources (FDMS)" désigne l'ensemble des techniques permettant la combinaison des données des capteurs en une seule piste globale. Le choix d'une technique appropriée pour cette combinaison dépend de l'architecture du système FDMS sous-jacent. Pour une architecture centralisée, les sources sont indépendantes et le filtre Kalman offre une solution optimale. Toutefois, dans une architecture décentralisée, une corrélation peut se créer entre les sources, et le filtre Kalman ne peut, par conséquent, être utilisé. La méthode de l'intersection des covariances a été proposée comme solution de rechange au filtre Kalman, pour la fusion des sources corrélées. Elle engendre toutefois une diminution de la performance. Un nouveau filtre, qui permet à la fois d'éviter l'inconsistance du filtre Kalman et la baisse de performance de l'intersection des covariances, est proposé dans ce document. La supériorité de l'approche proposée est illustrée pour le problème de pistage de cibles.

This page intentionally left blank.

Executive summary (U)

"Multiple Source Data Fusion" (MSDF) is used to indicate the general approach for combining the sensor data into global tracks. Since the benefits of a fusion process are very dependent on the way the sensor data are combined, the selection of the appropriate architecture represents a fundamental conceptual issue in developing surveillance and tracking systems. Hence, the architecture of an MSDF system can range from highly centralized to highly distributed. In the centralized approach, each individual sensor transmits its contacts to the fusion centre, where they serve to update and maintain a single master track file. On the contrary, the decentralized architecture allows each sensor to perform a maximum amount of pre-processing to generate sensor output decisions. Hence, each sensor individually maintains its own track file based exclusively upon its own measurement data processed by the local tracker. These sensor-level tracks are then transmitted to a central fusion node responsible for fusing them into composite tracks to form a master track file.

The selection of the appropriate MSDF techniques depends on the underlying architecture. For the centralized scheme, the sources (*i.e.*, contacts) are known to be independent. In this case, the Kalman filter provides an optimal solution. This independence assumption can however be relaxed in the case of correlated data, if the cross-covariance information is available. Unfortunately, if this correlation information is missing the Kalman filter cannot be applied. The case of correlated sources occurs particularly when the decentralized architecture is used. The pieces of data to be fused (*i.e.*, sensor-level tracks) are not statistically independent. In such situations, independence is often assumed and the correlation is simply ignored in the fusion process, to allow for the use of the Kalman filter. This makes the filter over optimistic in its estimation, which may lead to divergence. This property is known as the inconsistency phenomenon. The covariance intersection method has been proposed as a robust solution to the problem of the decentralized data fusion in the presence of an unknown correlation. To avoid the underestimation of the actual covariance matrix, the covariance intersection overestimates it, which results in a decrease in performance. To avoid both the inconsistency of the Kalman filter and the lack of performance of the covariance intersection, a new fusion algorithm, called the largest ellipsoid, is proposed. The proposed algorithm does not overestimate the intersection region, as does the covariance intersection, but slightly underestimates it. This will have no effect on the consistency of the fusion, since the intersection region represents an upper bound for the actual covariance. The three methods discussed are compared through the target's tracking problem, where the superiority of the proposed filter is clearly shown.

This document describes a part of the ongoing activity undertaken by the Situation Analysis Support Systems (SASS) Group in the Decision Support Systems (DSS) Section at Defence Research & Development Canada (DRDC) - Valcartier. This research activity concerns the investigation of advanced concepts for the adaptation and the integration of the data fusion and sensor management.

The results of this research could apply to any current Canadian military platform sensor suites and the possible future upgrades to improve their performance against predicted future threat.

Abder Rezak Benaskeur and Jean Roy. 2002. A consistent filter for robust decentralized data fusion. TR 2001-223. Defence Research & Development Canada - Valcartier.

Sommaire (U)

La fusion de données multi-sources sert à déterminer une démarche d'ensemble afin de combiner les données des capteurs en une seule piste globale. Comme les avantages du processus de fusion dépendent beaucoup de la façon dont sont combinées les données, le choix de l'architecture appropriée est fondamental dans la mise au point de système de surveillance et de poursuite. L'architecture du système de fusion de données multi-sources peut être très centralisée ou distribuée. Dans une architecture centralisée, chacun des capteurs renvoie les sources à un centre de fusion. Par contre, l'architecture décentralisée permet à chacun des capteurs d'effectuer un pré-traitement afin d'en arriver à une décision.

Le choix des techniques appropriées de fusion de données multi-sources dépend de l'architecture sous-jacente. Dans le modèle centralisé, les sources (contacts) sont indépendantes. Le filtre Kalman constitue la meilleure solution. Cette condition d'indépendance tombe dans le cas de données qui sont statistiquement dépendantes et dont on dispose de l'information d'inter-covariance. Malheureusement, on ne peut utiliser le filtre Kalman lorsqu'on ne possède pas cette information sur la corrélation. La question des sources corrélées se pose surtout lorsqu'on utilise une architecture décentralisée. Les résultas de la fusion (pistes) ne sont pas statistiquement indépendantes. Dans ce cas, on présuppose généralement qu'elles le sont et on ne tient pas compte de l'inter-covariance dans le processus de fusion, afin de pouvoir utiliser le filtre Kalman. Cette négligence a toutefois tendance à sous-estimer la matrice de covariance de l'erreur d'estimation. Cette caractéristique s'appelle le phénomène d'inconsistance.

La méthode "intersection des covariances " a été proposée pour résoudre le problème de la fusion décentralisée de données en présence de corrélation inconnue. Pour éviter de sous-estimer la matrice de covariance réelle, l'intersection des covariances la surestime, ce qui entraîne une diminution de la performance. Pour éviter à la fois l'incohérence du filtre Kalman et la faible performance de l'intersection des covariances, on a proposé un nouvel algorithme de fusion : la "plus grande ellipsoïde ". Contrairement à l'intersection des covariances, cet algorithme ne surestime pas la zone d'intersection, mais la sous-estime légèrement. Ceci n'a aucun effet sur la consistance du résultat de la fusion, puisque la zone d'intersection représente la limite supérieure de la covariance réelle. On a comparé les trois méthodes dans le cadre de la poursuite de cibles, où la supériorité du filtre proposé a été clairement démontrée.

Ce document décrit une partie d'une activité en cours entreprise par le Groupe des systèmes d'aide à l'analyse de la situation (SAAS) de la Section des systèmes d'aide à la décision (SAD) de Recherche et développement pour la défense Canada (RDDC) - Valcartier. Cette activité de recherche porte sur l'étude de concepts avancés pour l'adaptation et l'intégration de la fusion de données et la gestion des capteurs.

Les résultats de cette recherche pourraient profiter au système de capteurs des plates-formes militaires canadiennes, ainsi qu'à ses améliorations possibles à venir afin

d'en accroître la performance contre les menaces futures anticipées.

Abder Rezak Benaskeur and Jean Roy. 2002. Un filtre consistant pour la fusion décentralisée robuste de données. TR 2001-223. Recherche et Dèvelopment pour la Défence Canada - Valcartier.

Table of contents (U)

List of figures (U)

List of tables (U)

This page intentionally left blank.

1. Introduction

An ongoing activity undertaken by the Situation Analysis Support Systems (SASS) Group in the Decision Support Systems (DSS) Section at Defence Research & Development Canada (DRDC) - Valcartier is the investigation of advanced concepts for adaptation and integration of the data fusion and sensor management. These concepts could apply to any current Canadian military platform sensor suites, as well as the possible future upgrades, in order to improve their performance against the predicted future threat [1].

The reported work addresses the problem of automatically aggregating and extracting the desired information from multiple limited accuracy sensor/source data, in arbitrary fusion architecture. In most situations, the combination of such multiple source data allows for achieving more accuracy than could be achieved by the use of single source systems and yields more usable compromises. The term "Multiple Source Data Fusion (MSDF) architecture" is used to indicate the general method (or philosophy) used to combine the sensor data into global tracks. The following Joint Director Laboratory (JDL) definition of MSDF is being reported in [2].

> *"A continuous process dealing with association, correlation, and combination of data and information from multiple sources to achieve refined entity position and identity estimates, and complete and timely assessments of resulting situations and threats, and their significance".*

1.1 Fusion net architecture

For any given sensor suite configuration, there can be many different ways to combine data from the sensors. Therefore, a fundamental conceptual issue in developing an MSDF system for surveillance and tracking purposes is the selection of the appropriate architecture [3, 4, 5, 6]. This issue revolves about defining where to combine or fuse the data in the processing flow of multiple sensors, or equivalently the level of pre-processing of the information data that is fused.

The MSDF architecture is an important issue since the benefits of the fusion process are different depending on the way the sensor data are combined [7]. Within the surveyed literature, many different ways to combine data from multiple sensors have been found, offering as many architectural options to the MSDF system designer. The architecture of an MSDF system can range from highly centralized to highly distributed [8]. Examples of these two extreme architectures, for surveillance and tracking applications, are given below.

1.1.1 Centralized data fusion architecture

When the centralized fusion architecture is used, each individual sensor transmits its raw observations (or contact) to the fusion node where the

required processes are performed to generate and update global composite tracks, maintained in a single master track file [7]. The central fusion process performs the functions of data alignment, data association, target state estimation and kinematic behavior assessment, identification information fusion and track/cluster management to achieve joint estimates of the position, velocity, attributes or identity of entities in the environment.

1.1.2 Decentralized data fusion architecture

The decentralized sensor fusion architecture allows each sensor to perform a maximum amount of pre-processing to generate sensor output decisions (such as state vectors and declarations of identity) for the various entities in the environment [9]. Independent target detection, features extraction, state estimation and identification are thus potentially performed within the signal processor and tracker of each sensor [7]. The resulting target track data is normally stored in a track file. Hence, each sensor individually maintains its own track file based exclusively upon its own measurement data processed by the local tracker [10]. These sensor-level tracks are then transmitted to a central fusion process responsible for both finding the sensor tracks that likely represent the same target, and for combining or fusing these tracks into composite tracks to form a master track file. Within the central fusion process, data alignment, gating, assignment and fusion (positional and/or identity) are performed on state vectors rather than on raw data.

1.2 Fusion algorithm

The selection of the appropriate MSDF algorithms and techniques depends on the fusion architecture [7]. Hence, before an MSDF function can be implemented within a combat system, it must be analyzed in terms of the different types of architectures and implementations that are possible, the benefits and drawbacks of these architectures, and finally in terms of how all this relates to the performance and mission requirements of the platform.

1.2.1 Fusion algorithm for centralized architecture

For the centralized scheme, the sources of information to be fused (*i.e.*, contacts) are known to be statistically independent[1]. In this case, the Kalman filter yields, in the mean square error sense, the best performance [11].

[1]Two random variables are independent, if their joint probability density is the product of their marginal (individual) probability densities.

1.2.2 Fusion algorithm for decentralized architecture

The independence assumption can be relaxed in the case of correlated[2] data, if the cross-covariance information is available. In this case, the Kalman filter will still provide the best performance, by exploiting this additional information. Unfortunately, if this correlation information is missing or incomplete, which is often the case, the Kalman filter cannot be applied. This represents a major drawback, since, in practical applications, the complete independence of the sources cannot be guaranteed and the cross-covariance is rarely available.

The above-mentioned case of correlated sources occurs when the decentralized architecture is used. The pieces of data to be fused (*i.e.*, sensor-level tracks) are not guaranteed to be statistically independent, due mainly the common process noise and target's maneuvers that represent unavoidable sources of correlation. In such situations, to allow for the use of the Kalman filter, the independence is often assumed and the correlation is simply ignored in the fusion process. This makes the filter over optimistic in its estimation, which may lead to divergence [12]. The Kalman filter underestimates, in such a case, the actual error covariance. This property is known as the inconsistency phenomenon. One common solution, to avoid this problem, consists in increasing artificially the estimated covariance matrix, by introducing an empirically determined parameter. Since no rigorous method exists for choosing such a parameter [13], the reliability of the fusion process can be greatly compromised.

The covariance intersection method has been proposed recently [14] to bring a robust solution to the problem of decentralized data fusion in presence of an unknown and un-modeled correlation. This method allows for avoiding the inconsistency phenomenon, by replacing the Kalman filter fusion rule by a more conservative one. Since the covariance intersection uses only partial information about the sources, it leads to a sub-optimal solution. To avoid the underestimation of the actual covariance matrix, the covariance intersection overestimates it, which obviously results in a significant decrease in performance. To avoid both the inconsistency of the Kalman filter and the lack of performance of the covariance intersection, a new fusion algorithm, called the largest ellipsoid, is proposed. The latter provides a rigorous mean to increase the estimated error covariance matrix in the Kalman filter context. The amount of this increase is based upon the intersection concept used in the covariance intersection. Nevertheless, the proposed algorithm does not overestimate the intersection region, as does the covariance intersection, but slightly underestimates it. This will have no effect on the consistency of the fusion, since the intersection region represents an upper bound for the actual

[2]Correlation is the linear association between two random variables, usually measured by the cross-covariance matrix.

covariance. The Kalman filter covariance matrix is increased just enough to cover the intersection region, and it will therefore always remain smaller than the one resulting from the covariance intersection.

This report is organized as follows. The general problem of data fusion is stated in Section 2. The Kalman filter is introduced in Chapter 3, and in Chapter 4 is presented the covariance intersection. The limitation of both methods will be highlighted, and the new largest ellipsoid approach is proposed in Chapter 5. The two methods are compared, in Chapter 6, for the target's tracking problem. The conclusion appears in Chapter 7.

2. Data fusion

The data fusion provides the decision maker with an efficient tool to manage the information he might receive from a variety of sources and improve his current situation awareness, by producing an "as accurate as possible" explanatory picture of the battle environment. According to the JDL sub-panel [15], the data fusion process is subdivided into five levels (see Figure 1), where each succeeding level deals with a higher level of information abstraction. Level 1, to which we will be particularly interested in the sequel, deals with the object assessment. It uses the sensor data, from Level 0, to optimally estimate the current kinematical properties of the target, and predict their future positions. It also makes inferences about the target's identity and other key attributes. This identification issue will however not be considered in the sequel.

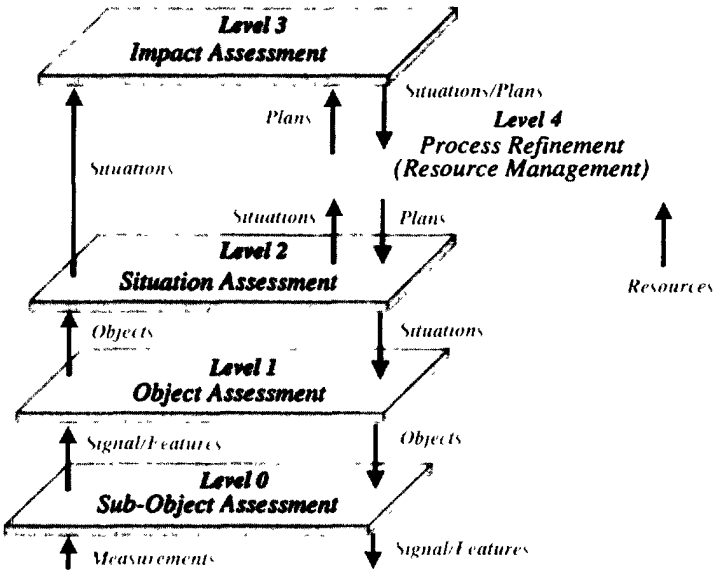


Figure 1: The revised JDL data fusion model

2.1 Source modeling

In the context of LIDF, the sources to be fused are often assumed corrupted by noise and therefore modeled as random variables, whose true statistics are unknown. Within the estimation theory framework, such stochastic variables are often represented in terms of means, or estimates, and covariance matrices, that is, the uncertainties associated with the estimation process. In this context, the main idea of data fusion is to obtain an estimate of some unknown variable from its two (or more) available noise-corrupted, direct or indirect, observations.

In the case of two sources, s_1 and s_2, the statistical representation of the available data

(observations) is given by the means $\hat{s}_1$ and $\hat{s}_2$, the assumed covariance of each source

$$\hat{S}_1 = E\left[\tilde{s}_1\tilde{s}_1^T\right] \tag{1}$$

$$\hat{S}_2 = E\left[\tilde{s}_2\tilde{s}_2^T\right] \tag{2}$$

and the assumed cross covariance of the two sources

$$\hat{S}_{12} = E\left[\tilde{s}_1\tilde{s}_2^T\right] \tag{3}$$

where the unknown estimation error terms, $\tilde{s}_1$ and $\tilde{s}_2$, are defined by

$$\tilde{s}_1 = \hat{s}_1 - s_1 \tag{4}$$

$$\tilde{s}_2 = \hat{s}_2 - s_2 \tag{5}$$

The fusion problem can then be characterized as finding a mathematical method to describe how the information about $\hat{s}_1$ and $\hat{s}_2$ can be used, in an efficient and provably optimal way, to construct a new estimate $\hat{s}$ (and an estimate of its associated measure of accuracy $\hat{S}$) which minimizes some cost function, while guaranteeing consistency. This consistency notion, upon which will be based the comparison of the fusion methods presented in the subsequent sections, is a very important issue in the estimation theory.

2.2 Consistency

An estimate $\hat{s}$ is said consistent, or conservative, if its stated level of performance is smaller than the actual one [16]. If the inverse of the covariance matrix (*i.e.*, the Fisher information matrix) is used as a measure of the performance, the consistency will be equivalent to

$$\hat{S}^{-1} \leq S^{-1} \qquad \equiv \qquad \hat{S} \geq S \tag{6}$$

where S is the actual covariance matrix and $\hat{S}$ its estimate provided by the fusion algorithm. The inequality in (6) is a matrix inequality, which means that the difference matrix is positive semi-definite, *i.e.*,

$$\hat{S} - S \geq 0 \tag{7}$$

In other words, this means that

$$x^T(\hat{S} - S)x \geq 0, \qquad \forall x \in R^n \tag{8}$$

2.3 Geometric interpretation

In the remainder, to better show the consistency notion, the covariance (or k_σ) contour concept will be used. For each covariance matrix P, a set of ellipsoids will be drawn and serves as an accuracy measure. These k_σ ellipsoids are defined by

$$C_k = \left\{ x \middle| (x - x_0)^T P^{-1} (x - x_0) = k \right\} \tag{9}$$

where k is a constant. The (covariance) matrix $\boldsymbol{P}$ determines then how far the ellipsoids extend in every direction from their center $\boldsymbol{x}_0$. This is a geometric visualization that provides a useful tool for comparing incompatible matrices. The lengths of the semi-axes of $\mathcal{C}_k$ are given by $\sqrt{\lambda_i}$, where λ_i are the eigenvalues of the matrix $\boldsymbol{P}$.

For a random variable, with a covariance matrix $\boldsymbol{P}$, the $\boldsymbol{k_\sigma}$ ellipsoids result in contours of constant probability. Figure 2 gives an example of contours for a normally distributed random variable in $\boldsymbol{R}^2$.

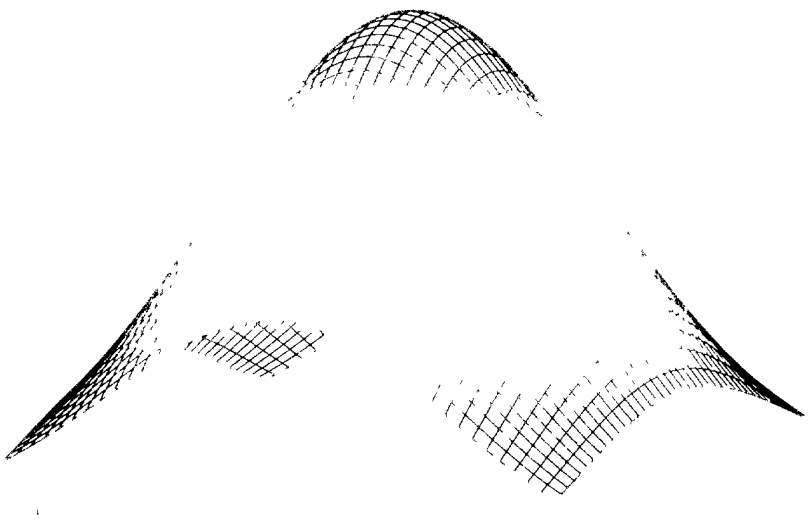


Figure 2: *Representation of the k_σ contours for a two-dimensional random variable, with a Gaussian distribution and a covariance matrix $\boldsymbol{P}$*

2.4 Fusion rule

The updated estimate is often obtained via a linear combination of the two available estimates $\hat{\boldsymbol{s}}_1$ and $\hat{\boldsymbol{s}}_2$.

$$\hat{\boldsymbol{s}} = \boldsymbol{W}_1\hat{\boldsymbol{s}}_1 + \boldsymbol{W}_2\hat{\boldsymbol{s}}_2 \tag{10}$$

where the weights $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ are computed to minimize some cost function of the resulting covariance matrix $\hat{\boldsymbol{S}}$, whose expression is given by

$$\hat{\boldsymbol{S}} = \boldsymbol{W}_1\hat{\boldsymbol{S}}_1\boldsymbol{W}_1^T + \boldsymbol{W}_1\hat{\boldsymbol{S}}_{12}\boldsymbol{W}_2^T + \boldsymbol{W}_2\hat{\boldsymbol{S}}_{21}\boldsymbol{W}_1^T + \boldsymbol{W}_2\hat{\boldsymbol{S}}_2\boldsymbol{W}_2^T \tag{11}$$

In the subsequent sections, solutions to the fusion problem are discussed. The Kalman filter will first be introduced in Chapter 3, where the emphasis will be put on its inconsistency problem. The superiority, from the consistency viewpoint, of the covariance intersection is shown in Chapter 4. To compensate for the decrease in

performance caused by the overestimation of the intersection region by the covariance intersection, a new method (*i.e.*, the largest ellipsoid) is proposed in Chapter 5. This solution represents a good compromise between the performance of the Kalman filter and the robustness of the covariance intersection.

3. Kalman filter

Incontestably, the Kalman filter is one of the most important tools in the estimation theory. It has rapidly found a natural place in the L1DF applications, to become one of the most used algorithms. Within the Kalman filtering context, the cost function to be minimized is the trace of the estimation error covariance matrix $\hat{S}$ given by equation (11). The Kalman filter yields the "best" performance, regardless of the underlying error distribution. It provides a mathematically rigorous and provably stable method for fusing information in real-time, and performs, in the Minimum Mean Square Error (MMSE) sense, better than any other linear filter. This MMSE (or $\mathcal{L}_2$) optimality means that, if a number of candidate filters were run many times on the same problem, the average result of the Kalman filter would be better than the average result of any other. Note that many ways are possible to define the optimality, depending on which criterion the evaluation of the filter performance is based on.

3.1 Fusion of independent sources

Under the source independence assumption

$$S_{12} = S_{21} = 0 \tag{12}$$

the combined covariance matrix (11) reduces to

$$\hat{S} = W_1 \hat{S}_1 W_1^T + W_2 \hat{S}_2 W_2^T \tag{13}$$

The solution to the underlying optimization problem is then given by

$$W_1 = \hat{S}_2 \left[\hat{S}_1 + \hat{S}_2 \right]^{-1} = \left[\hat{S}_1^{-1} + \hat{S}_2^{-1} \right]^{-1} \hat{S}_1^{-1} \tag{14}$$

$$W_2 = \hat{S}_1 \left[S_1 + S_2 \right]^{-1} = \left[\hat{S}_1^{-1} + \hat{S}_2^{-1} \right]^{-1} \hat{S}_2^{-1} \tag{15}$$

which results in the following fusion algorithm

$$\hat{S}^{-1} = \hat{S}_1^{-1} + \hat{S}_2^{-1} \tag{16}$$

$$\hat{S}^{-1} \hat{s} = \hat{S}_1^{-1} \hat{s}_1 + \hat{S}_2^{-1} \hat{s}_2 \tag{17}$$

3.2 Consistency

It is clear that, under the independence assumption, the consistency of both sources is sufficient to guarantee the consistency of the resulting estimate [16]. Indeed, if

$$\hat{S}_1 \geq S_1 \quad \text{and} \quad \hat{S}_2 \geq S_2 \tag{18}$$

then

$$\hat{S} \geq S \tag{19}$$

In the case of Gaussian sources, the inequality (19) becomes an equality. This means that in the case of the independent sources, the actual error covariance is equal to the one estimated by the Kalman filter, if all the errors follow Gaussian distributions.

3.3 Mutlisource fusion

The fusion algorithm given by equations (16) and (17) can be easily generalized to the case of n sources. This is known as the "batch fusion" and expressed as

$$\hat{S}^{-1} = \hat{S}_1^{-1} + \hat{S}_2^{-1} + \cdots + \hat{S}_n^{-1} \tag{20}$$

$$\hat{S}^{-1}\hat{s} = \hat{S}_1^{-1}\hat{s}_1 + \hat{S}_2^{-1}\hat{s}_2 + \cdots + \hat{S}_n^{-1}\hat{s}_n \tag{21}$$

3.4 Fusion of correlated sources

To analyze the effect of the source correlation on the performance of the Kalman filter, the cross covariance matrix of the two sources is redefined as follows

$$Cov\left(\begin{bmatrix}\tilde{s}_1\\ \tilde{s}_2\end{bmatrix}\right) = \begin{bmatrix}\hat{S}_1 & \hat{S}_{12}\\ \hat{S}_{21} & \hat{S}_2\end{bmatrix} = \begin{bmatrix}\hat{S}_1 & S_c\\ S_c^T & \hat{S}_2\end{bmatrix} \tag{22}$$

where

$$S_c = \varrho\sqrt{\hat{S}_1\hat{S}_2} \tag{23}$$

This notation allows for explicitly showing the effect of the correlation coefficient ϱ.

3.4.1 Known correlation

If the cross covariance information, *i.e.* S_c, is available, the independence assumption, imposed by the Kalman filter, can be relaxed. The filter will then still provide the "best" performance, by exploiting this additional information. The resulting optimal weights are given by

$$\begin{aligned} W_1 &= (\hat{S}_2 - S_c^T)\left[\hat{S}_1 + \hat{S}_2 - S_c - S_c^T\right]^{-1} \\ &= \left[(\hat{S}_1 - S_c)^{-1} + (\hat{S}_2 - S_c^T)^{-1}\right]^{-1}(\hat{S}_1 - S_c)^{-1} \end{aligned} \tag{24}$$

and

$$\begin{aligned} W_2 &= (\hat{S}_1 - S_c)\left[\hat{S}_1 + \hat{S}_2 - S_c - S_c^T\right]^{-1} \\ &= \left[(\hat{S}_1 - S_c)^{-1} + (\hat{S}_2 - S_c^T)^{-1}\right]^{-1}(\hat{S}_2 - S_c^T)^{-1} \end{aligned} \tag{25}$$

One can easily show that the resulting covariance matrix will be given by

$$\hat{S} = \hat{S}_1 - (\hat{S}_1 - S_c)\left[\hat{S}_1 - S_c + \hat{S}_2 - S_c^T\right]^{-1}(\hat{S}_1 - S_c^T) \tag{26}$$

3.4.2 Unknown correlation

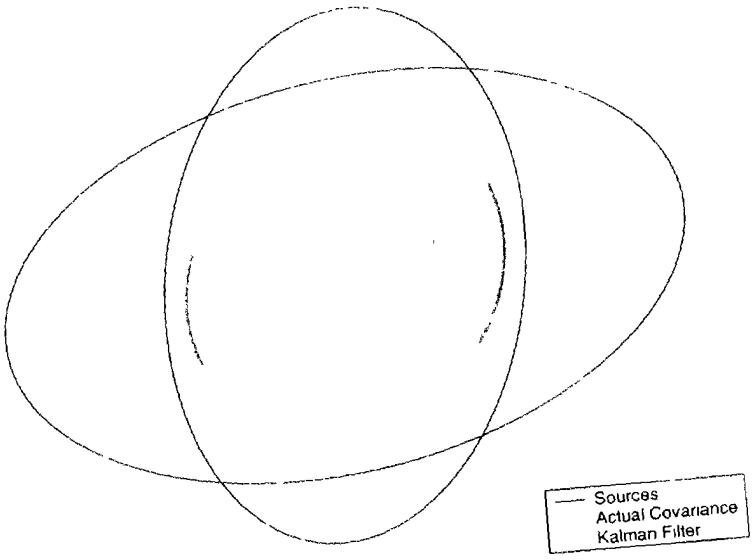
Unfortunately, if the correlation information is missing, the Kalman filter cannot theoretically be applied. In most situations, to allow for the use of the Kalman filter, the independence is often assumed and the correlation is simply ignored in the fusion process.

Figures 3 (a) & (b) show the actual error covariance ellipsoid (or k_σ contour) that results from the fusion of two correlated sources, for different cross-correlation strong, assuming that this cross-correlation is known. It can be noticed that the actual covariance ellipsoid will always lie within the region defined by the intersection of the covariance ellipsoids of the fused sources, whatever the degree of correlation between these sources is. From Figure 3 (b), it is clear that ignoring the cross-correlation, in the fusion process, leads to an underestimation of the actual error covariance for positive values of the correlation coefficient ϱ.

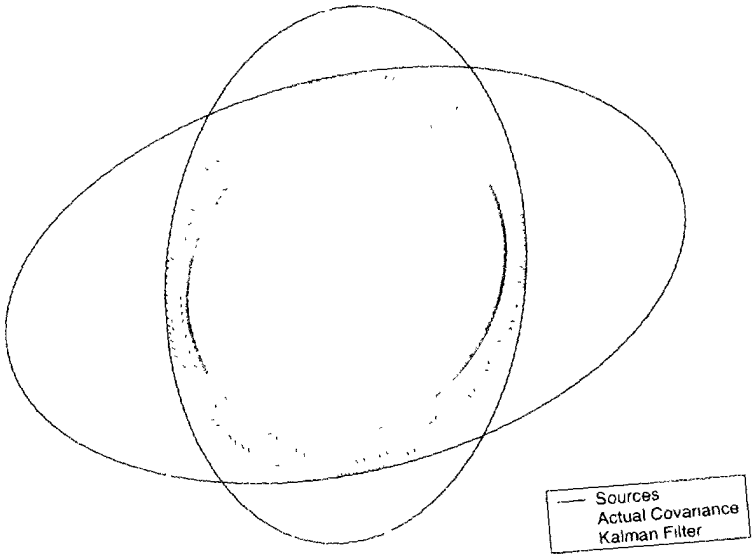
This means that the Kalman filter overestimates its own performance, which may result in a divergence phenomenon. One common solution, to avoid this inconsistency problem, consists in increasing artificially the estimated covariance matrix, by introducing an empirically determined parameter. This palliative solution is also known as "the fudge factor" [13]. Since there is no rigorous method for choosing this parameter, the stability and the reliability of the Kalman filter can be greatly compromised.

To overcome this limitation, some methods use an approximation of the missing cross covariance matrix. An example of algorithms for computing the cross-covariance matrix is given by the weighted covariance method [17, 18]. It has however been proven by Simukai [19] that it is not possible to maintain a consistent cross-covariance matrix, in any arbitrary fusion architecture. This makes the weighted covariance only applicable in "small" fusion networks with a known data flow. This also the case of the methods based on the redundancy removal, like the tracklet fusion [20, 21] and the information filter [22]. Since the aim of this report is not the comparison of the fusion algorithms, the above-mentioned methods will not be considered in the sequel. For comparative study of the these methods, in the context of the track-level fusion, the reader is referred to [23].

The covariance intersection method, presented in detail in Chapter 4, allows for exploiting the full power of decentralized architectures, by using a slightly different update rule from the one of the Kalman filter. Without any assumption about the cross covariance, it brings an interesting solution to the problem of the fusion in presence of an unknown and un-modeled correlation.



(a) Negative correlation $-.9 \leq \varrho < 0$



(b) Positive correlation $0 < \varrho \leq .9$

Figure 3: *1_σ contour (for $x \in R^2$) of the actual covariance matrix obtained by the fusion of two correlated sources (for different correlation coefficient values) and estimated covariance matrix yielded by the Kalman filter (under the independence assumption, i.e., $\varrho = 0$)*

4. Covariance intersection

The covariance intersection [14] is a recent method that can represent, in many situations, a good alternative to the Kalman filter, since it makes possible a consistent fusion of data provided by correlated sources. This fusion is performed without considering the cross-correlation information nor assuming the independence of the sources The "intersection" terminology is related here to the geometric analogy, based upon the covariance k_σ contour plots (see Section 2.3).

By using a Kalman-like fusion rule, the actual error covariance ellipsoid will always lie within the region defined by the intersection of the covariance ellipsoids of the fused sources, whatever the degree of correlation between these sources is (see Figures 3 (a) & (b)). Based on this observation, and in order to guarantee the consistency, the covariance intersection computes a covariance matrix that will always enclose the intersection region. The latter represents, in fact, an upper bound for the actual covariance.

4.1 Fusion rule

In the case of two sources, $(\hat{s}_1, \hat{S}_1)$ and $(\hat{s}_2, \hat{S}_2)$, the fusion rule used by the covariance intersection is given by a convex combination of both of the estimates and the inverse of the covariance matrices. The resulting combined estimate can be written as

$$\hat{s} = \omega W_1 \hat{s}_1 + (1-\omega) W_2 \hat{s}_2 \tag{27}$$

where $\omega \in [0,1]$ is a weight factor. This parameter offers an additional degree of freedom, whose value is determined according to the error norm (of the combined estimate) one wants to minimize (see Appendix B). This may, for instance, be the determinant (the product of the eigenvalues), the trace (the sum of the eigenvalues), $\mathcal{L}_\infty$ (the maximum eigenvalue), ...etc. The minimization problem, with respect to W_1 and W_2, gives a similar solution to the Kalman gains, namely,

$$W_1 = \left[\omega \hat{S}_1^{-1} + (1-\omega)\hat{S}_2^{-1}\right]^{-1} \hat{S}_1^{-1} \tag{28}$$

$$W_2 = \left[\omega \hat{S}_1^{-1} + (1-\omega)\hat{S}_2^{-1}\right]^{-1} \hat{S}_2^{-1} \tag{29}$$

and the corresponding fusion algorithm can then be expressed as

$$\hat{S}^{-1} = \omega \hat{S}_1^{-1} + (1-\omega)\hat{S}_2^{-1} \tag{30}$$

$$\hat{S}^{-1}\hat{s} = \omega \hat{S}_1^{-1}\hat{s}_1 + (1-\omega)\hat{S}_2^{-1}\hat{s}_2 \tag{31}$$

One can easily show that, in the case of equally weighted sources ($\omega = .5$), the covariance intersection yields the same mean (estimate) as the Kalman filter, that is

$$\hat{s}_{CI} = \hat{s}_{KF} \tag{32}$$

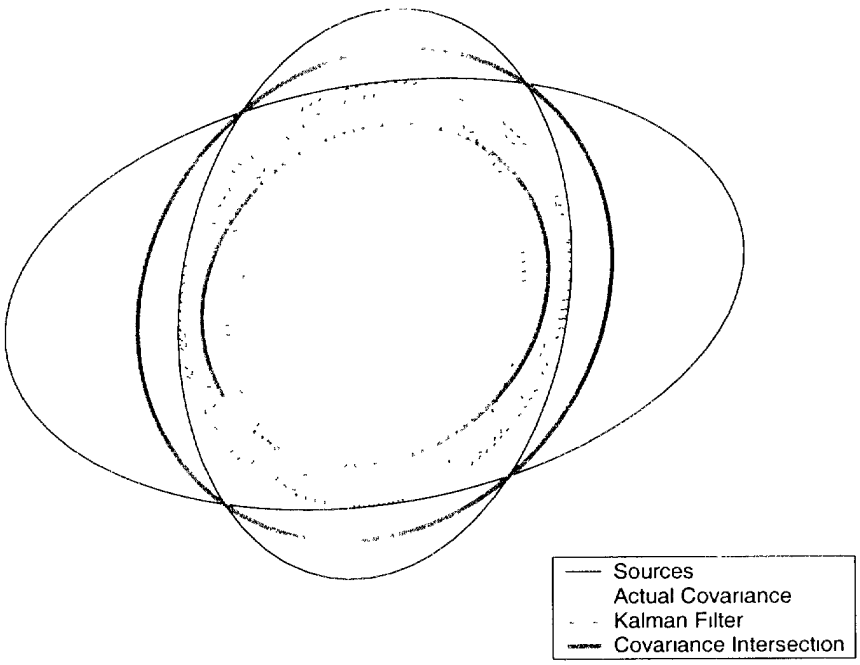


Figure 4: *1_σ contour (for $x \in R^2$) of the actual covariance matrix obtained by the fusion of two correlated sources (for different correlation coefficient values $-9 \leq \varrho \leq 9$) and estimated covariance matrix yielded by the Kalman filter (under the independence assumption, i e , $\varrho = 0$) and the covariance intersection update rules, respectively*

Its corresponding covariance matrix is however twice the Kalman filter one,

$$\hat{S}_{CI} = 2\hat{S}_{KF} \tag{33}$$

4.2 Consistency

It has been proven [14, 24] that, without assuming the independence, the combined estimate is guaranteed to be consistent, given the consistency of the two sources. This means that, if

$$\hat{S}_1 \geq S_1 \quad \text{and} \quad \hat{S}_2 \geq S_2 \tag{34}$$

then

$$\hat{S} \geq S \tag{35}$$

This statement remains true whatever is the correlation intensity ϱ (see Figure 4) and the parameter ω value (see Figure 5). This correlation-independent consistency represents the unique and major advantage the covariance intersection has over the Kalman filter. The covariance intersection update rule is however sub-optimal in the sense that it results in a covariance matrix that will always be larger than the one estimated by the Kalman filter, even if there is actually no correlation (see Figure 4).

4.3 Multisource fusion

As for the Kalman filter, the generalization to the n-source case, of the fusion rule given by the equations (30)–(31), is straightforward.

$$\hat{S}^{-1} = \omega_1 \hat{S}_1^{-1} + \omega_2 \hat{S}_2^{-1} + \cdots + \omega_n \hat{S}_n^{-1} \tag{36}$$

$$\hat{S}^{-1}\hat{s} = \omega_1 \hat{S}_1^{-1}\hat{s}_1 + \omega_2 \hat{S}_2^{-1}\hat{s}_2 + \cdots + \omega_n \hat{S}_n^{-1}\hat{s}_n \tag{37}$$

subject to

$$\sum_{j=1}^{n} \omega_j = 1 \tag{38}$$

4.4 Comparison

As previously stated, from Figure 4, one can notice that the covariance matrix estimated by the covariance intersection method is always larger than the actual one, which means that the estimate yielded by the covariance intersection will always be consistent. This is, unfortunately, not the case of the estimate given by the Kalman filter. The case of actually independent sources, and correlated sources with a known cross correlation, are the only situations where the Kalman filter is consistent. In these cases, the Kalman filter will provide a solution which is superior to the covariance intersection, since the latter, even if it is also consistent, results in a much larger covariance estimate.

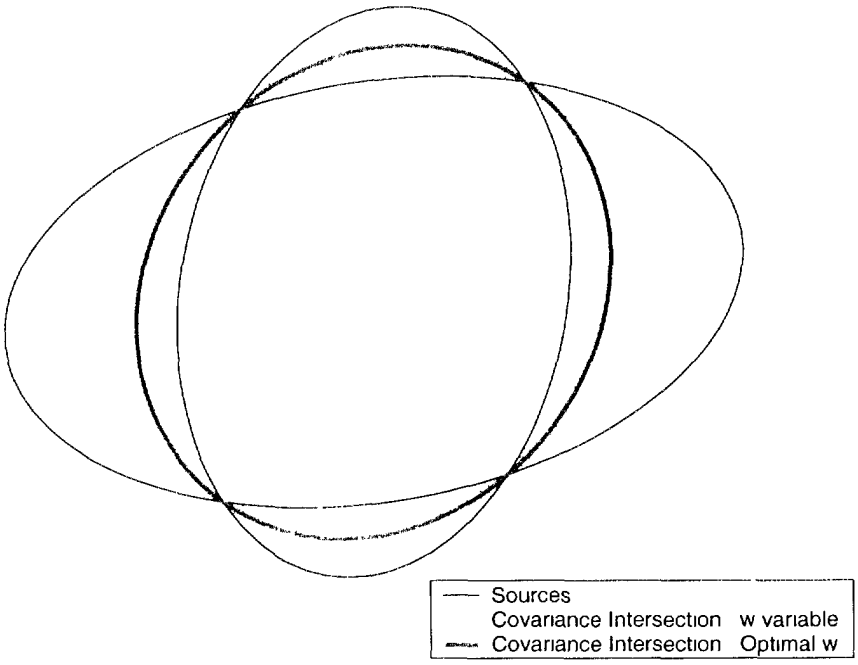


Figure 5: *1_σ contour (for $x \in R^2$) of the combined estimate yielded by the fusion of two correlated sources, using the covariance intersection update rule for different values of the parameter ω $(0 \leq \omega \leq 1)$*

In all cases where the degree of dependence is supposed unknown, and the sources are therefore assumed independent, the covariance intersection is superior in the sense that it still yields a consistent estimation. The Kalman filter gives, as shown by Figures 3 and 4, an underestimation of the actual covariance matrix, for postive values of ϱ. From Figure 4, one can notice that, without any assumption about the correlation between the sources, the covariance intersection was able to provide a consistent fusion rule. The shown results correspond to two correlated sources, whose correlation coefficient ϱ is taken within the range $-.9 \leq \varrho \leq .9$.

Finally, the effect of the parameter ω on the performance of the covariance intersection is observed in Figure 5. The combined covariance reduces to the covariance of the first source, for $\omega = 0$, and to the covariance of the second source, for $\omega = 1$. For $0 < \omega < 1$ the resulting covariance is given by the convex combination of the two covariance matrices. For the case of centred sources[3], the estimation is still consistent independently of the value of the parameter ω. Note, also, that for both variable ϱ and variable ω cases, the actual covariance always lies within the intersection region defined by the source covariance matrices.

4.5 Non-centred sources

In all the presented scenarios, the data sources are taken centred, with however different covariance matrices. In such a case, the fusion algorithm will only help in reducing the uncertainty in the estimation, *viz.*, the covariance. The estimate value will remain unchanged throughout the estimation process. For completeness, the more general case of non-centred sources is presented, for the same scenarios, in Appendix A.

4.6 Alternative approach

Since the intersection region represents (in all directions) an upper limit for the actual error covariance matrix, and the covariance intersection method overestimates it, the latter results in a significant decrease in performance. This is why the method is said sub-optimal. To avoid this large overestimation of the covariance, a new filter, called the largest ellipsoid method, is presented in Chapter 5.

[3] Which provide the same mean.

5. Largest ellipsoid

As in the case of the covariance intersection, the design of this filter is also based on the estimation of the intersection region of the covariance matrices. Instead of overestimating this region, the new filter will however slightly underestimate it. This will have no consequence on the consistency of the fusion, since the intersection region represents an upper limit. This means that the actual covariance will almost always lie within the region defined by the estimated covariance. The very rare cases, where the actual covariance may partly lie outside the estimated one, are those of highly correlated sources. And even in those cases, the difference is very small and only due the orientations of the matrices and not to their sizes (trace, determinant, etc.).

Since the k_σ contour defined by the actual covariance matrix is an ellipsoid, the proposed filter computes the "largest ellipsoid" contained within the intersection region. The matrix orientation problems may however make the direct computation of this ellipsoid very difficult, and even impossible (see Figure 6).

In the case where the matrices have the same orientation, *i.e.*, their eigenvectors are two-by-two parallel; the ellipsoid computation reduces to a comparison between the eigenvalues associated with each pair of parallel eigenvectors. The eigenvectors of the intersection ellipsoids are identical to those of the fused matrices. Nevertheless, problems arise in the more general case, where the matrices do not have the same orientation, as illustrated in Figure 6, where besides the ellipsoids, two segments are plotted for each matrix. Each segment is associated with an eigenvector (and its corresponding eigenvalue) of the considered matrix. This plots help in getting a better understanding of the matrix orientation incompatibility problem. Computing the size and the orientation of the intersection ellipsoid is not as straightforward as in the case of compatible matrices. To solve the orientation incompatibility problem, geometrical transformations must be applied to the two fused matrices.

5.1 Geometrical transformations

The two fused matrices are brought, via geometrical transformations, within a space where they have compatible orientations. This allows for performing a simple comparison between the eigenvalues associated with their parallel eigenvectors. The resulting matrix, corresponding to the intersection ellipsoid, is returned back to the original space by applying the reverse of the applied transformations.

Rotation

The first transformation consists in rotating the fused matrices to make the base (*i.e.*, eigenvectors) of the matrix $\hat{S}_1$ aligned[4] with the base (see Figure 7) of the Cartesian

[4]This is an arbitrary choice, and the matrix $\hat{S}_2$ could have been chosen. This would have led to an identical result.

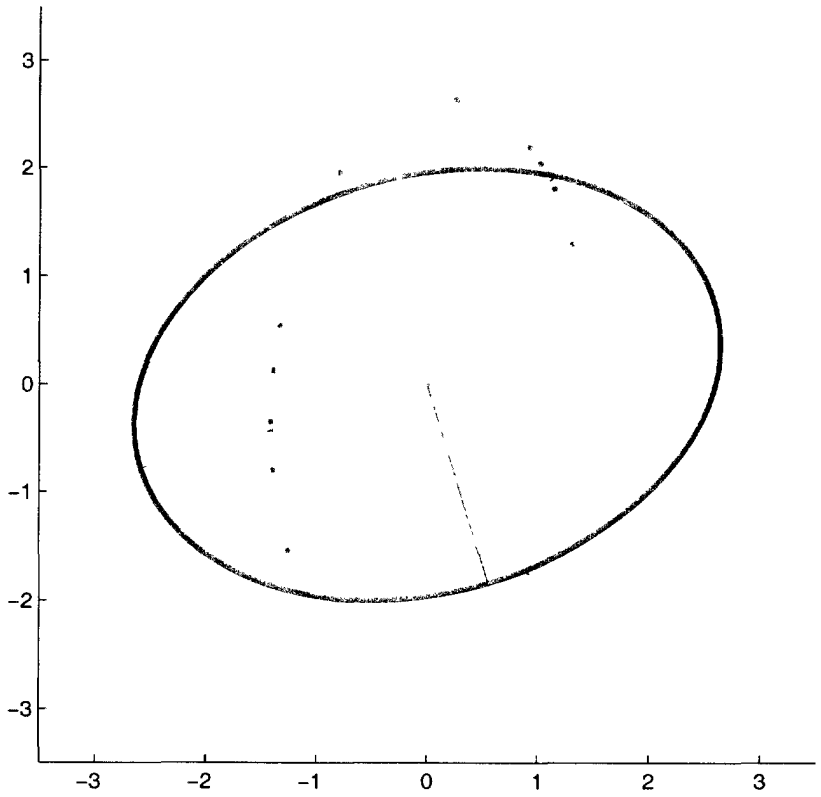


Figure 6: *Eigenvalues and eigenvectors of the two fused error covariance matrices $\hat{S}_1$ & $\hat{S}_2$*

plan. The following matrix represents this transformation.

$$\boldsymbol{T}_r = \begin{bmatrix} \boldsymbol{v}_{1_1}^T & \boldsymbol{v}_{1_2}^T & \ldots\ldots & \boldsymbol{v}_{1_n}^T \end{bmatrix}^T \tag{39}$$

where $\boldsymbol{v}_{1_j}$ is the j^{th} eigenvector of $\hat{\boldsymbol{S}}_1$. The resulting matrices are noted

$$\hat{\boldsymbol{S}}_1^r = \boldsymbol{T}_r \hat{\boldsymbol{S}}_1 \boldsymbol{T}_r^T \tag{40}$$

$$\hat{\boldsymbol{S}}_2^r = \boldsymbol{T}_r \hat{\boldsymbol{S}}_2 \boldsymbol{T}_r^T \tag{41}$$

Scaling

The second geometrical transformation is given by

$$\boldsymbol{T}_s = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & \sqrt{\frac{\lambda_{1_1}}{\lambda_{1_2}}} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \sqrt{\frac{\lambda_{1_1}}{\lambda_{1_n}}} \end{bmatrix} \tag{42}$$

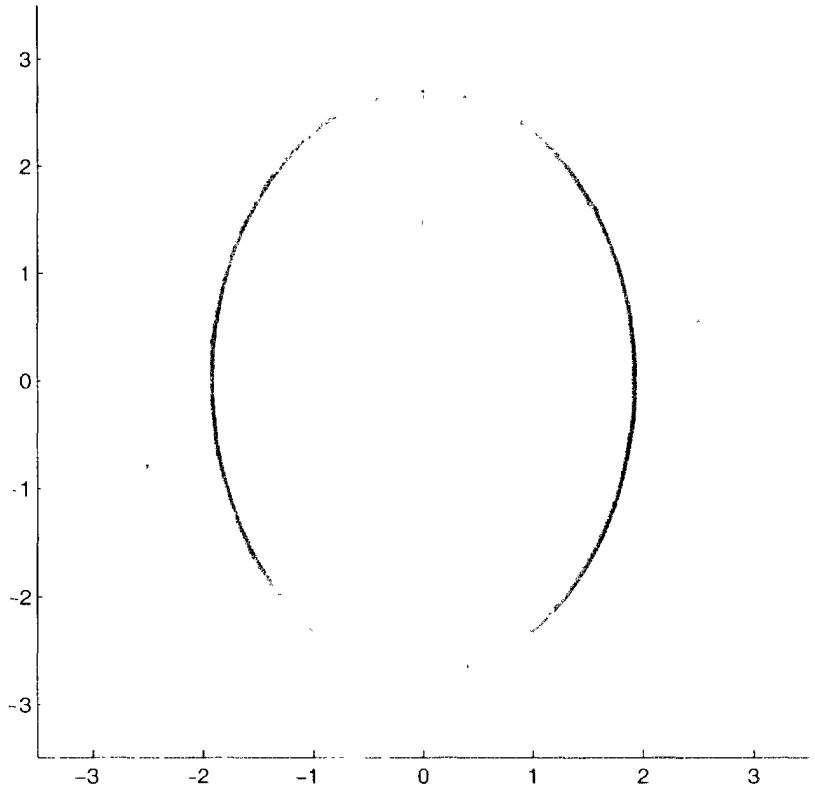


Figure 7: *Eigenvalues and eigenvectors of the rotated error covariance matrices $\hat{S}_1'$ & $\hat{S}_2'$*

where λ_{1_j} is the j^{th} eigenvalue of $\hat{S}_1'$. This is a scaling transformation that makes equal all the eigenvalues of the matrix:

$$\hat{S}_1^{sr} = T_s \hat{S}_1' T_s^T \tag{43}$$

$$= T_s T_r \hat{S}_1 T_r^T T_s'^T \tag{44}$$

As shown in Figure 8, this brings the matrices to the space, where the σ-contours defined by $\hat{S}_1^{sr}$ reduce to circles. Since for circles, all the directions are similar, one can take, as comparison directions, the eigenvectors

$$\begin{bmatrix} v_{2_1}^T & v_{2_2}^T & \cdots & v_{2_n}^T \end{bmatrix}^T \tag{45}$$

of the transformed matrix

$$\hat{S}_2^{sr} = T_s \hat{S}_2^r T_s^T \tag{46}$$

$$= T_s T_r \hat{S}_2 T_r^T T_s'^T \tag{47}$$

whose σ-contours are still ellipsoids. The eigenvectors of $\hat{S}_2^{sr}$ will therefore serve as eigenvectors of the intersection ellipsoid. The corresponding eigenvalues are given by

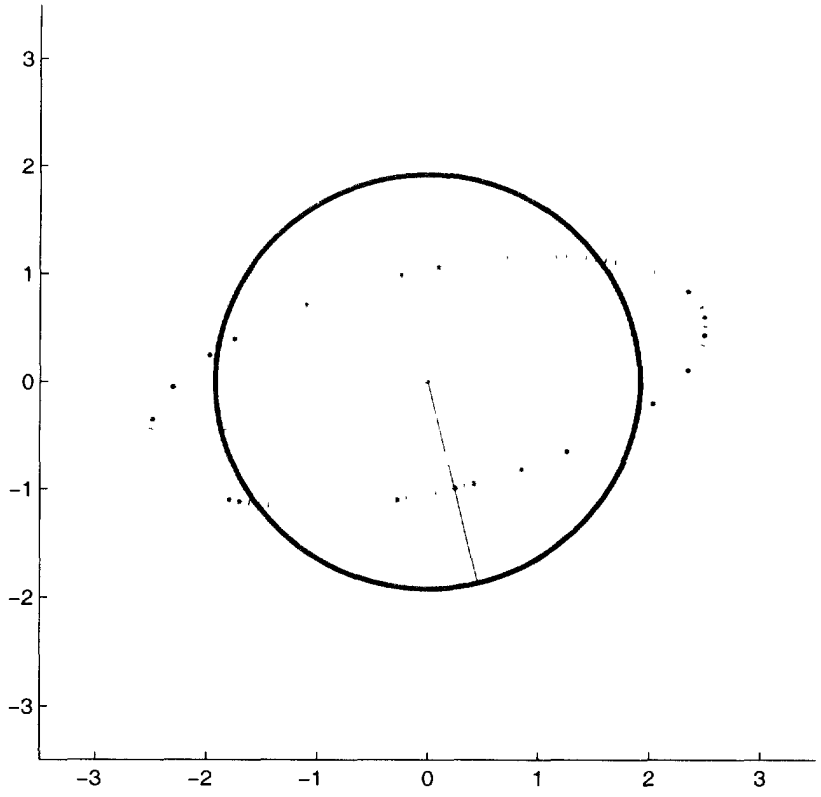


Figure 8: *Eigenvalues and eigenvectors of the rotated and scaled error covariance matrices* $\hat{S}_1^{sr}$ *&* $\hat{S}_2^{sr}$

the diagonal elements of the following matrix

$$\boldsymbol{D}_{min} = \begin{bmatrix} \min(\lambda_{1_1}, \lambda_{2_1}) & 0 & \dots & 0 \\ 0 & \min(\lambda_{1_1}, \lambda_{2_2}) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & . & \min(\lambda_{1_1}, \lambda_{2_n}) \end{bmatrix} \tag{48}$$

where λ_{2_j} is the j^{th} eigenvalue of $\hat{\boldsymbol{S}}_2^{sr}$. Hence, in the transformed space, the intersection ellipsoid is given by (see Figure 9).

$$\hat{\boldsymbol{E}}^{sr} = \begin{bmatrix} \boldsymbol{v}_{2_1} & \dots & \boldsymbol{v}_{2_n} \end{bmatrix} \boldsymbol{D}_{min} \begin{bmatrix} \boldsymbol{v}_{2_1}^T \\ \vdots \\ \boldsymbol{v}_{2_n}^T \end{bmatrix} \tag{49}$$

whose corresponding matrix in the original space (see Figure 10) can be obtained by applying the reverse transformation, as follows

$$\begin{aligned}\hat{\boldsymbol{E}} &= \boldsymbol{T}_r^{-1}\boldsymbol{T}_s^{-1}\hat{\boldsymbol{E}}^{sr}\boldsymbol{T}_s^{-T}\boldsymbol{T}_r^{-T} \\ &= \boldsymbol{T}_r^{-1}\boldsymbol{T}_s^{-1}\begin{bmatrix}\boldsymbol{v}_{2_1} & \cdots & \boldsymbol{v}_{2_n}\end{bmatrix}\boldsymbol{D}_{min}\begin{bmatrix}\boldsymbol{v}_{2_1}^T \\ \vdots \\ \boldsymbol{v}_{2_n}^T\end{bmatrix}\boldsymbol{T}_s^{-T}\boldsymbol{T}_r^{-T}\end{aligned} \tag{50}$$

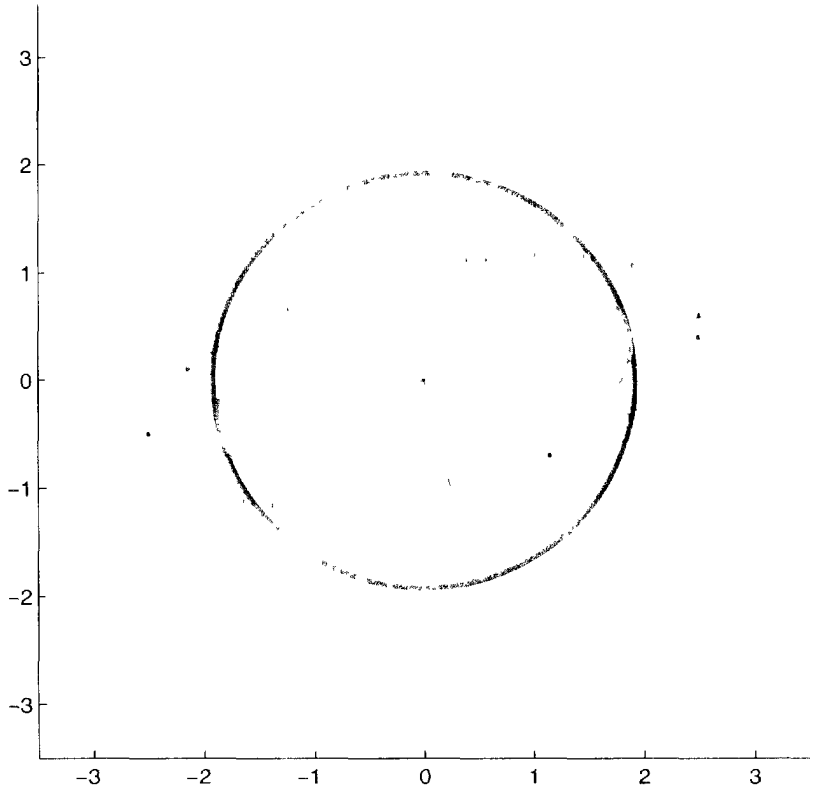


Figure 9: *Intersection ellipsoid in the transformed space*

Besides the computation of the largest ellipsoid matrix, a Kalman filter fusion is run. The resulting estimate (17) is kept, while the covariance matrix (16) is dropped and replaced by the computed one (50).

5.2 Comparison

In order to allow for a better comparison of the proposed fusion rule with the Kalman filter and the covariance intersection, the corresponding 1_σ contours are plotted in Figure 11 The contours of the actual error covariance matrix, for different values of the

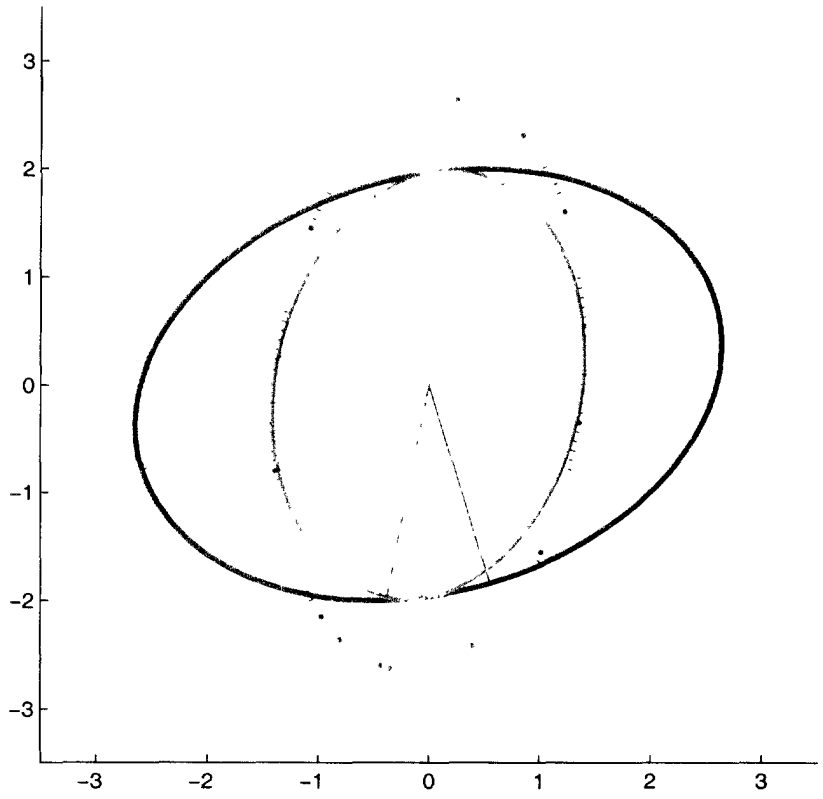


Figure 10: *Intersection ellipsoid in the original space*

cross-correlation coefficient ($-1 \leq \varrho \leq 1$), are also plotted to show the consistency of the largest ellipsoid-based fusion. One can easily notice the significant difference between the sizes of the ellipsoids resulting, respectively, from the covariance intersection and the largest ellipsoid. As stated previously, the Kalman filter is still, for the considered values of ϱ, suffer from inconsistency. The reduction by the largest ellipsoid fusion of the estimated covariance matrix size results in an increased performance (faster convergence and smaller estimation error). While the fusion is still consistent, the estimated covariance matrix is tighter (*i.e.*, closer the actual one) than the one yielded by the covariance intersection algorithm. The presented methods are illustrated and compared, in Chapter 6, using the target's tracking problem.

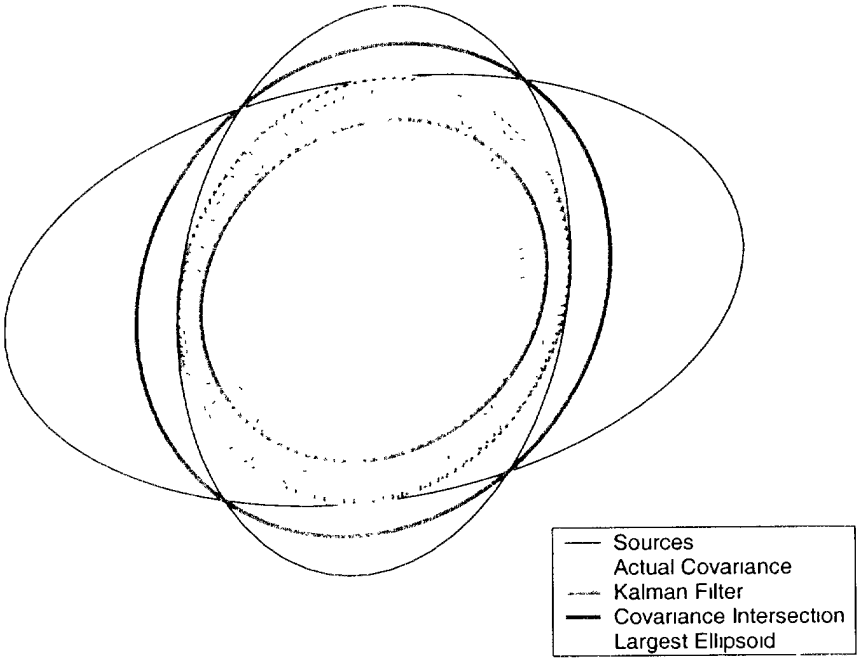


Figure 11: *1_σ contour of the combined estimate, yielded by the largest ellipsoid method vs the Kalman filter and the covariance intersection for correlated sources*

6. Application to the track-level fusion

The methods presented in the previous chapters are illustrated and compared in this one using the target's tracking problem whose fusion network architecture is given by Figure 12. In the tracking problem, one is interested in providing an improved estimate of the state vector, such that the target's position can be depicted as correctly as possible. In a multi-sensor environment, the tracking problem may, as explained below, present both of the independent and correlated source fusion cases. But since the independent source case admits the standard the Kalman filter as an optimal solution, the emphasis will be put on the more challenging correlated case.

The tracked target is assumed to be moving in a 2D space, where the acceleration acts as an input[5]. The state vector to be estimated is therefore composed of the target's coordinates, *viz.*, the position and the linear velocity. With the following state variable notation

$$\boldsymbol{x} = \begin{bmatrix} p_x \\ p_y \\ \dot{p}_x \\ \dot{p}_y \end{bmatrix} \quad \text{and} \quad \boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{51}$$

the equations of a such a target can be expressed as

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \Gamma \boldsymbol{v}_k \tag{52}$$

where $\boldsymbol{v}_i$ are random variables that reflect the unforeseeable variation of the acceleration, in both directions, and

$$\boldsymbol{F}_k = \begin{bmatrix} 1 & 0 & h & 0 \\ 0 & 1 & 0 & h \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{53}$$

h is the time increment. The matrix Γ, in (52), depends on the model used to represent the discrete-time nature of the process noise $\boldsymbol{v}_k$. Two examples are given below. The first one, which is given by

$$\Gamma = \begin{bmatrix} h^2/2 & 0 \\ 0 & h^2/2 \\ h & 0 \\ 0 & h \end{bmatrix} \tag{54}$$

[5]Since the acceleration may change unforeseeably, it is often modeled as a random variable, and so it will be here.

represents to the pulse model, while the following one

$$\Gamma = \begin{bmatrix} \frac{\sqrt[3]{h^2}}{2\sqrt{3}} & \frac{\sqrt[3]{h^2}}{2} & 0 & 0 \\ 0 & 0 & \frac{\sqrt[3]{h^2}}{2\sqrt{3}} & \frac{\sqrt[3]{h^2}}{2} \\ 0 & \sqrt{h} & 0 & 0 \\ 0 & 0 & 0 & \sqrt{h} \end{bmatrix} \tag{55}$$

is used by the Brownian motion model. Notice that the latter requires four random variables instead of two. If the process noises are assumed to have the same standard deviation δ_v, the pulse model results in the following process noise covariance matrix

$$Q = \delta_v^2 \Gamma\Gamma^T \tag{56}$$

$$= \delta_v^2 \begin{bmatrix} \frac{h^4}{4} & 0 & \frac{h^3}{2} & 0 \\ 0 & \frac{h^4}{4} & 0 & \frac{h^3}{2} \\ \frac{h^3}{2} & 0 & h^2 & 0 \\ 0 & \frac{h^3}{2} & 0 & h^2 \end{bmatrix} \tag{57}$$

while the covariance matrix for the Brownian motion is given by

$$Q = \delta_v^2 \Gamma\Gamma^T \tag{58}$$

$$= \delta_v^2 \begin{bmatrix} \frac{h^3}{3} & 0 & \frac{h^2}{2} & 0 \\ 0 & \frac{h^3}{3} & 0 & \frac{h^2}{2} \\ \frac{h^2}{2} & 0 & h & 0 \\ 0 & \frac{h^2}{2} & 0 & h \end{bmatrix} \tag{59}$$

6.1 Fusion architecture

For any given sensor suite configuration, there can be many different ways to combine data from the sensors, offering as many architectural options to the data fusion system designer. The architecture of such a system can range from a highly centralized to a highly distributed. In the case illustrated in Figure 12, one possible solution consists in maintaining sensor-level tracks using local sensor information at each sensor site, finding (in a central fusion node) the sensor tracks that potentially represent the same target and then combining these tracks into global tracks. This architecture is typically referred to as "Track-Level Fusion". The primary alternative architecture assumes that all of the raw sensor measurements (*i.e.*, sensor contacts) are sent directly to the centralized fusion node to be combined into global tracks. This architecture is typically referred to as "Contact-Level Fusion".

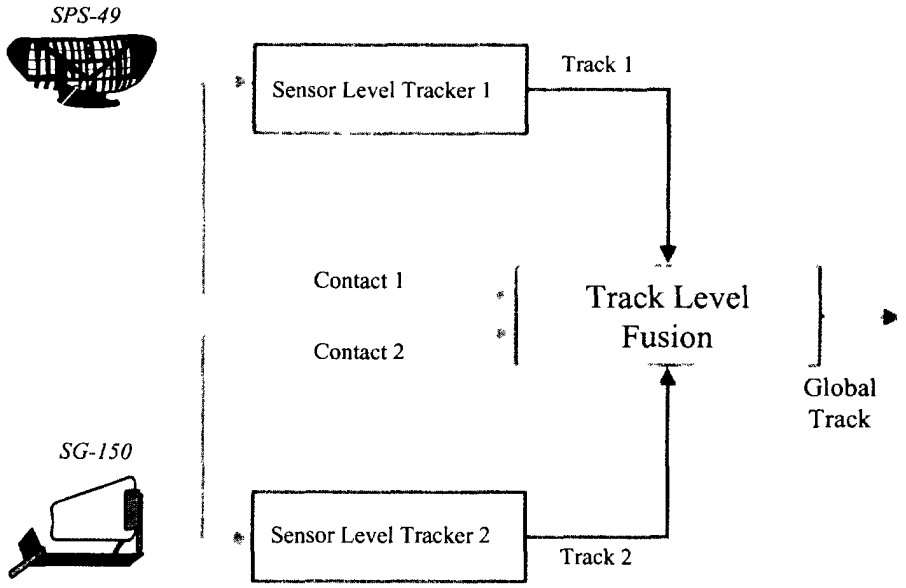


Figure 12: *Fusion network architecture*

6.1.1 Contact-level fusion

The contact-level fusion problem is first considered. This corresponds to the case of actually independent sources, since the contacts and the estimation noises, at the sensor level, are not correlated. The target's position (in both directions), is the only measured variable. This can be represented by the following observation equation

$$\boldsymbol{z}_{i_{k+1}} = \boldsymbol{H}_i \boldsymbol{x}_{k+1} \tag{60}$$

where $k+1$ is the observation time, i is the sensor number $(=1,2)$ and $\boldsymbol{H}_i$ is the observation matrix that is given, for both sensors, by

$$\boldsymbol{H}_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{61}$$

This means that, based on this partial observation, the fusion algorithm will have to provide an estimate for both of the position and the velocity of the target.

6.1.2 Track-level fusion

The track-level fusion architecture allows each sensor to perform a maximum amount of pre-processing to generate sensor output decisions. The resulting target track data is normally stored in a track file. Hence, each sensor

individually maintains its own track file based exclusively upon its own measurement data processed by the local tracker. These sensor-level tracks are then transmitted to a central fusion process responsible for both finding the sensor tracks that likely represent the same target, and for combining or fusing these tracks into composite tracks to form a master track file. Since the sensor-level tracks are state vectors, the observation equation at central node can be expressed as

$$\boldsymbol{z}_{k+1} = \boldsymbol{H}\boldsymbol{x}_{k+1} \tag{62}$$

where the observation matrix $\boldsymbol{H}$ is this time given by

$$\boldsymbol{H}_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{63}$$

The track-level fusion approach has the advantage of reduced communication requirements when compared with a centralized one, and therefore requires much less I/O bandwidth (reduced data-bus loading) to transmit the data between the sensors and the central node. The local tracks are periodically transferred to the central processor rather than the copious measurement data. If necessary, the sensor-level tracks may also be communicated less frequently than the arrival of the sensor data. The approach also has the advantage of reduced computational loading (in any single processor). In military applications, due to the distributed tracking capabilities, the decentralized sensor fusion results in increased survivability when compared with centralized systems [4, 10]. Moreover, if one sensor becomes degraded, its observations will not affect the sensor-level tracks of the other sensors (i.e., the good sensor-level tracks will not be corrupted by the bad data). By checking the sensor-level tracks with the central tracks, one may be able to detect any errors in the sensors. Then, when the sensor with poor data is finally recognized, the central-level tracks can be formed using only sensor-level tracks from un-degraded sensors [25].

Correlation

Because the process noise introduced by the target behavior is observed by all the sources tracking a common entity, Bar-Shalom [5] has shown that the tracks may be correlated, and that this correlation must be considered in the fusion process. The dependence between the estimation errors from the two track files arises from the common process noise entering into the state equation of the two tracking filters (*i.e.*, the common error source due to the target dynamics), given that the two information processors follow the same target. For example, a sudden target maneuver can lead to a bias error for both

tracking filters. The fact that the two measurement noise sequences processed by these different filters can be assumed independent is not sufficient to ensure the independence of the estimation errors in the two track files.

After track-to-track associations have been determined, the local tracks that correspond to the same targets as seen by different sensors must eventually be combined at a later stage. An often-employed, and overly simplistic, solution to this problem is achieved by choosing the least uncertain set of measurements and disregarding the remaining information. This is known as track selection or "best track approach". The pitfalls of such a naive technique are an inferior accuracy and reduced spatial and temporal coverage [26]. Hence, if sensor-level tracks are maintained, they must be combined at some point if significant benefit is to be derived from the multi-sensor fusion approach. The result is central-level tracks that are updated with sensor-level track data instead of with sensor report data. If a central-level track is updated with a sensor-level track, the usual assumption (valid for the case of raw measurements with uncorrelated measurement error) of error independence from one update period to another is not valid.

Since it deals with correlated sources, the track-level fusion problem is more challenging than the sensor-level one. In this case, the methods discussed in Chapters 3 to 5 are applied and compared. Due to its proven optimality, the contact fusion approach is taken as a reference, in the performance evaluation. Nevertheless, the performance is not the unique comparison criterion. The consistency is another important property that helps in choosing the appropriate approach for a given situation. To compare the different algorithms, the traces of the actual and the estimated error covariance matrices of the central tracker are plotted. All the plotted traces are normalized with respect to the trace of the estimated error covariance matrix yielded by the contact fusion algorithm.

6.2 Results

The presented results are obtain via 200 Monte-Carlo runs. The used covariance matrix corresponds to the pulse model, with $\delta_v = 12$ and $h = .1$. The sensor noise standard deviations are given, respectively, by $\delta_{v_x} = 5$ and $\delta_{v_y} = 4$ for the first sensor, and $\delta_{v_x} = 6$ and $\delta_{v_y} = 3$, for the second one. Notice that the sensors are not assumed identical, and the positions in each direction (x or y) is measured with a different accuracy.

To obtain a comparison means between the presented methods, two parameters are defined (see Table 1). The first one ϱ_c corresponds, for each considered method, to the ratio of the trace of the estimated covariance matrix to the trace of the actual one. This gives an direct measure of the consistency of the fusion algorithms. Note that a method is inconsistent, if $\varrho_c \leq 1$. The second parameter ϱ_p, which is ≥ 1, deals with the

performance. It is defined by the ratio of the trace of the actual covariance matrix (yielded by each method) to the actual covariance matrix yielded by the centralized contact fusion. Note that the more ϱ_p is close to 1, the more the corresponding method is performent.

The Kalman filter assumes the independence between the sensor-level tracks, and between each of the sensor-level tracks and the central track. This obviously leads to inconsistency and an increase in the estimation error. This phenomenon is clearly shown in Figure 13, where the fusion algorithm underestimates (-65%) the actual error covariance matrix ($\varrho_c = 0.35$). Also, the estimation error increases considerably ($\varrho_p = 1.27$). As shown in Figure 14, the covariance intersection allows for avoiding the inconsistency of the simple fusion. The actual covariance matrix is indeed smaller than the assumed one ($\varrho_c = 1.12$). This consistency is however obtained at the expense of a significant decrease in performance ($\varrho_p = 1.20$).

The results given by Figures 15 & 16 are conclusive, since, while the estimate is still consistent ($\varrho_c = 1.10$), the assumed covariance, is much closer the centralized contact-level fusion one, than that obtained by the covariance intersection. Indeed, the actual error is only 10% lager than the one obtained with the centralized contact-level fusion, while it is 20% in the case of the covariance intersection. The performance ratio is given by $\varrho_p = 1.10$. The simulation results are summarized in Table 1. It is clear that, the only consistent approaches are the covariance intersection and the largest ellipsoid.

Method	ϱ_p	ϱ_c
Contact fusion	1.00	1.01
Kalman filter	1.25	0.35
Covariance intersection	1.20	1.12
Largest ellipsoid	1.10	1.10

Table 1: *Performance/Consistency comparison of the fusion methods*

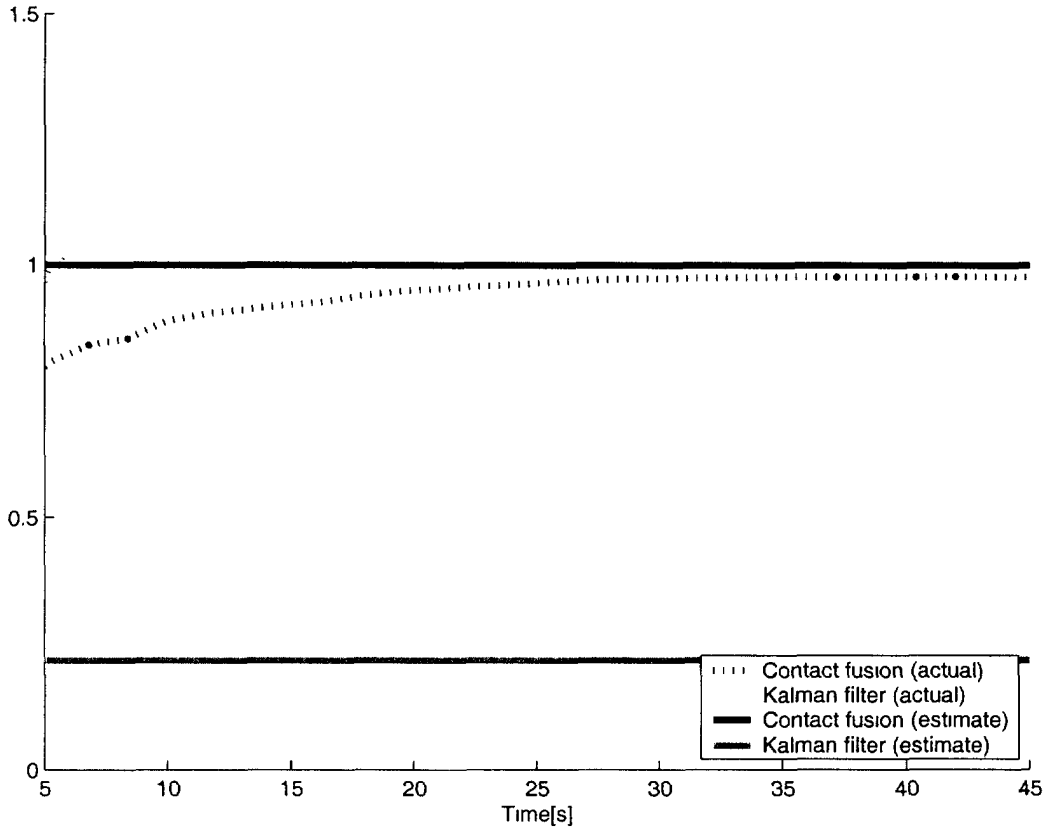


Figure 13: *Kalman filter track-level fusion*

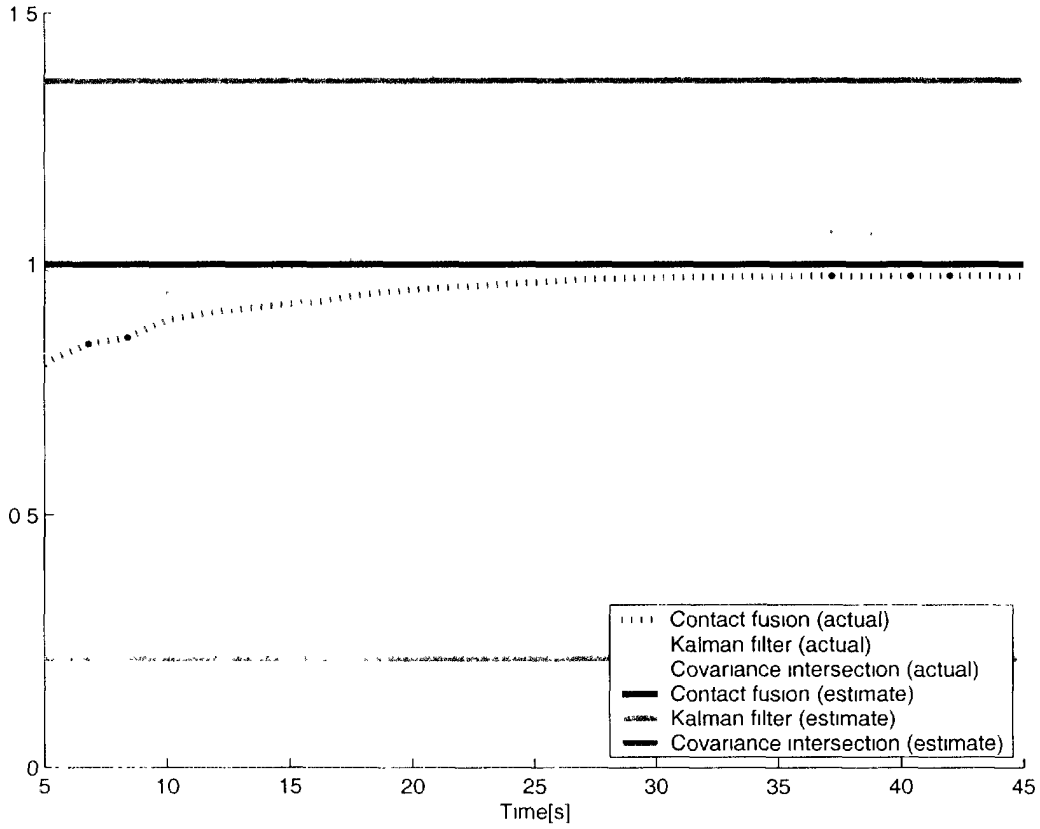


Figure 14: *Covariance intersection track-level fusion*

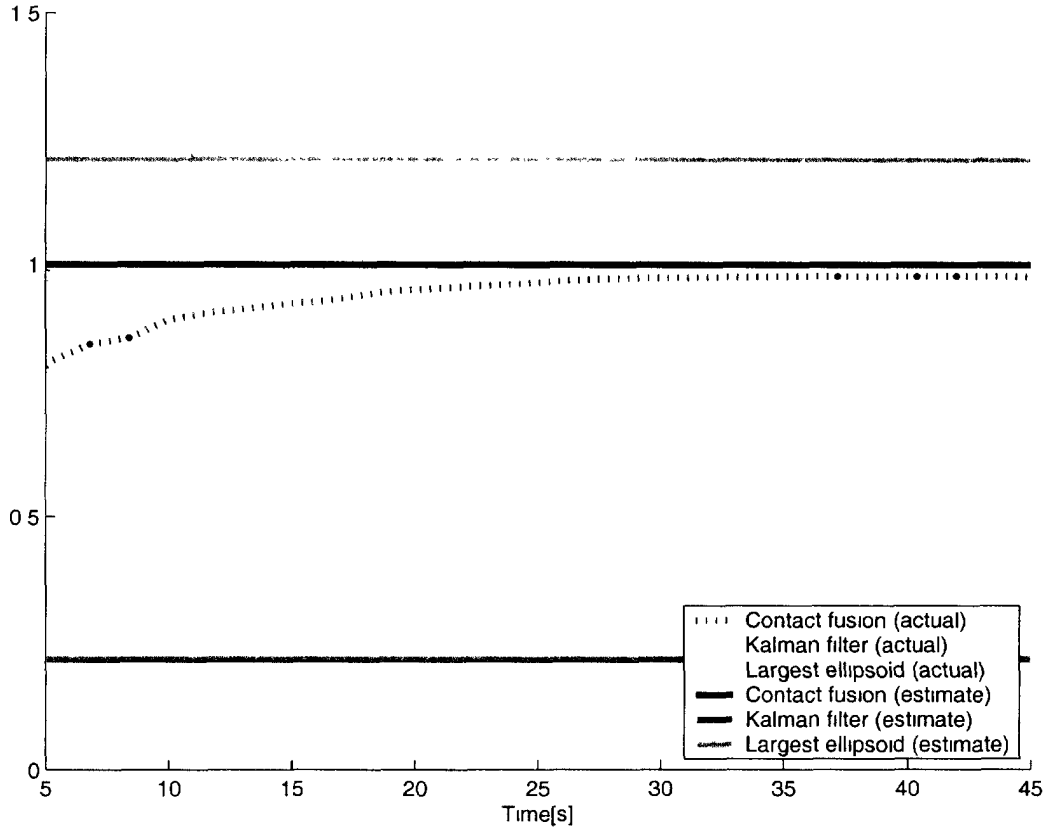


Figure 15: *Largest ellipsoid track fusion vs contact fusion*

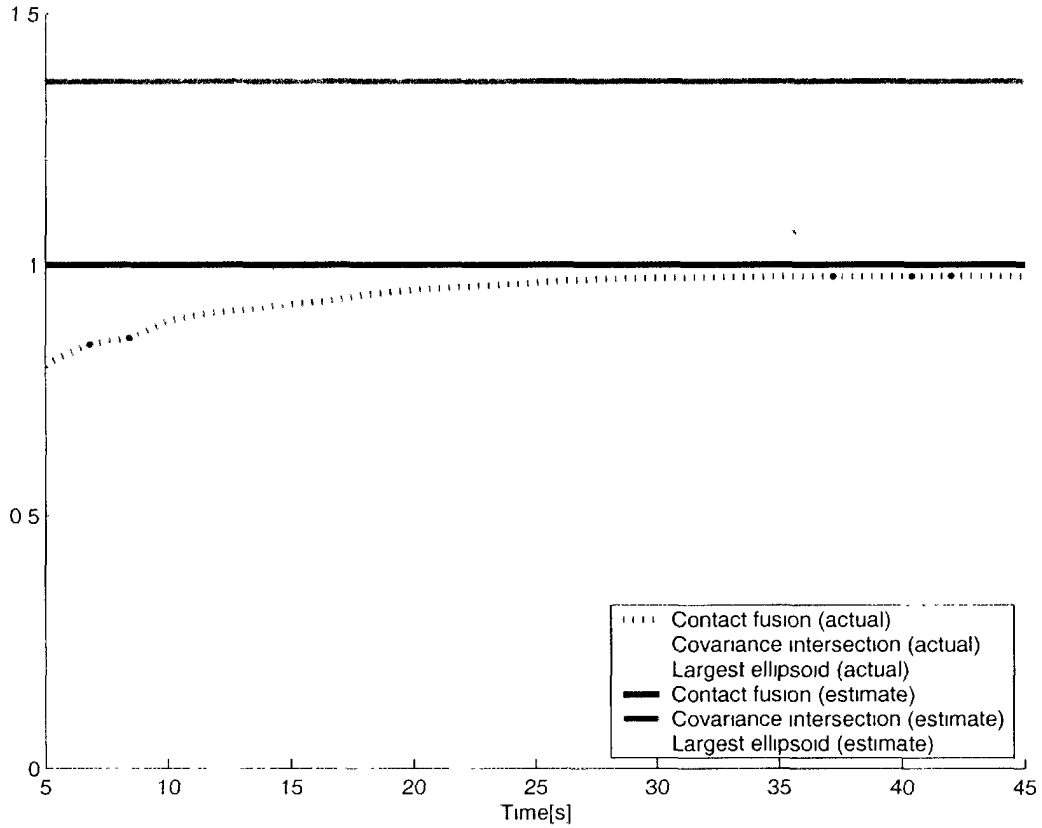


Figure 16: *Largest ellipsoid vs covariance intersection track fusion*

7. Conclusion

Data fusion algorithms govern how information about a given system is extracted from multiple source data. Data fusion provides the decision maker with an efficient tool to manage the information he might receive from a variety of sources and improve his current situation awareness by producing an "as accurate as possible" explanatory picture of the battle environment. The data fusion process is subdivided into five levels, where each succeeding level deals with a higher level of information abstraction. In this report, the emphasis is put on L1DF, where, two key algorithms for data fusion, *viz.*, the Kalman filter and the covariance intersection, are presented and their limitations are highlighted. An alternative solution, which represents a good compromise, is then proposed. In the case of actually independent sources, or correlated sources with a known cross covariance matrix, the Kalman filter yields the best performance. If the correlation information is missing or incomplete, the filter will lead to an inconsistent estimate. The covariance intersection avoids this inconsistency phenomenon, by using a more conservative fusion rule. Unlike the Kalman filter, the covariance intersection update strategy does not make any assumption about the source independence. This property allows for the covariance intersection to fuse correlated sources with an unknown cross covariance. Nevertheless, this results in a decrease in performance, leading to an only sub-optimal solution.

The new proposed Largest Ellipsoid filter offers a good alternative to the covariance intersection method. As in the latter, this consistent fusion approach is also based on the concept of intersection. Nevertheless, it leads to a tighter estimate, since it does not overestimate the intersection region, but slightly underestimates it. This has no consequence on the consistency property of the fusion, since the underestimated region represents an upper bound for the actual covariance.

References (U)

1. Roy, J.M.J., Bossé, É., and Des Groseilliers, L. (1995). State of the Art in Local Area Sensor Data Fusion for Naval Command and Control Afloat. DREV TR 9410. UNCLASSIFIED.

2. Hall, D. L. and Linn, R. J. (1990). A Taxonomy of Multi-Sensor Data Fusion Techniques. *In Proceedings of the 1990 Joint Service Data Fusion Symposium, Vol. 1*, pp. 593–610.

3. Waltz, E. and Llinas, J. (1990). Multi-sensor Data Fusion, Boston: Artech House.

4. Bar-Shalom, Y. (Editor) (1992). Multitarget-Multisensor Tracking : Applications and Advances, Artech House.

5. Bar-Shalom, Y. and Li, X.-R. (1995). Multitarget-Multisensor Tracking: Principles and Techniques, Storrs, CT: Yaakov Bar-Shalom.

6. Roy, J. and Bossé, É (1995). Definition of a Performance Evaluation Methology for Sensor Data Fusion Systems. DREV TR 9423. UNCLASSIFIED.

7. Roy, J., Bossé, É, and Duclos-Hindié, N. (1998). Performance Comparison of Contact-level and Track-level Sensor Fusion Architectures. *Otp. Eng.*, **37**(2), 434–440.

8. Antony, R.T. (1995). Principles of Data Fusion Automation, Norwood, MA: Artech House.

9. Hall, D. (1992). Mathematical Techniques in Multisensor Data Fusion, Norwod, MA: Artech House.

10. Blackman, S. (1986). Multi-Target Tracking with Radar Applications, Norwod, MA: Artech House.

11. Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME, Journal of Basic Engineering*, **82**, 34–45.

12. Maybeck, P. (1979). Stochastic Models, Estimation and Control, Vol. 1. Academic Press.

13. Bar-Shalom, Y. and Fortman, T. (1988). Tracking and Data Association, Vol. 179 of *Mathematics in Science and Engineering*. Academic Press.

14. Uhlmann, J. K. (1996). General Data Fusion for Estimates with Unknown Cross Covariances. *In Proceedings of the SPIE AeroSense Conference*, pp. 165–173.

15. Steinberg, A., Bowman, C., and White, F (1998). Revision to the JDL data fusion model. In *Joint NATO/IRIS Conference*, Quebec City.

16. H., Jazwinski A. (1970). Stochastic Processes and Filtering Theory, Academic Press.

17. Bar-Shalom, Y. (1981). Develpment of Track-to-Track Correlation Problem. *IEEE Transactions on Automatic Control*, **2**(26), 571–572.

18. Bar-Shalom, Y. and Campo, L. (1986). The Effect of Common Noise on the Two-Sensor Fused-Track Covariance. *IEEE Transactions on Aerospace and Electronic Systems*, **6**(22), 803–805.

19. Simukai, U. (1995). Network Management in Decentralized Sensing Systems. Ph.D. thesis. University of Oxford.

20. Chee-Yee, C. (1998). Distributed Architecture for Data Fusion. *In Proceedings of the International Conference on Multisource-Multisensor Information Fusion*, pp. 84–91.

21. Miller, M. D., Drummond, O. E., and Perrella, A. J. (1998). Tracklets and Covariance Truncation Options for Threater Missile Tracking. *In Proceedings of the International Conference on Multisource-Multisensor Information Fusion*, pp. 165–173.

22. Chee-Yee, C., Mori, S., and Chang, K. (1985). Information Fusion in Distributed Sensor Networks. *In American Control Conference*, pp. 830–835.

23. Benaskeur, A. and Roy, J. (2001). A Comparative Study of Data Fusion Algorithms for Target Tracking. DRDV - Valcartier TR 2001-224. UNCLASSIFIED.

24. Julier, S. and Uhlmann, J. K.. A Non-divergent Estimation Algorithm in the Presence of Unknown Correlation. www.ait.nrl.navy.mil/ people/uhlmann/CovInt.html.

25. Roy, J., Bossé, É, and Duclos-Hindié, N. (1996). Quantitative comparison of sensor fusion architectural approaches in an algorithm-level testbed. In Drummond, O. E., (Ed.), *Signal and Data Processing of Small Tragets*, SPIE, pp. 373–384. Orlando, FL.

26. Bruder, S. and Farooq, M. (1990). Sensor Fusion Techniques and a Hybrid Multi-Sensor Multi-Target Tracker. Department of Electrical and Computer Engineering, Royal Military College of Canada. (Technical Report No. 90/3). Kingston, Ontario.

27. Y., Nesterov and A., Nemirovsky (1994). Interior-point Polynomial Methods in Convex Programming. *Studies in Applied Mathematics. SIAM, Philadelphia, PA*, **13**.

Annex A
Fusion of non-centred sources

The centred sources used, in Chapters 3 to 5, to analyze the consistency of fusion algorithms, are replaced here by the more realistic non-centred ones. As shown in Figures A.1 and A.5, the results for the Kalman filter and the largest ellipsoid are very similar to the case of centred sources. The only difference lies in the estimated value (*i.e.*, the centre of the ellipsoid) that is affected, this time, by the action of the filters. The consistency/inconsistency properties remain however the same.

A major difference is noticed in the case of a variable ω. Since the actual covariance does not, this time, always lie in the region defined by the intersection of the covariance of the fused sources, all the ellipsoids defined by the covariance intersection (*i.e.* for different values of ω) do not enclose the actual covariance. Indeed, as shown in Figure A.2, depending on the value of the parameter ω, the actual covariance may be partly outside the intersection region. Nevertheless, this observation does not mean that the covariance intersection filter will underestimate the actual covariance. As can be observed in Figure A.2, the estimate covariance contours are still larger than the actual ones.

Figure A.3 shows better the covariance intersection fusion rule obtained by computing the optimal value of the parameter ω, which is clearly consistent. Figure A.4 compares the Kalman filter to the covariance intersection, while Figure A.5 gives a comparison of the three methods. This figure shows clearly the superiority of the proposed approach. Even though the computation is based on non-centred data sources, it results in what would have been the largest ellipsoid contained in the intersection region if the sources were centred. In this case, the computed ellipsoid will naturally be larger than the intersection region, because the latter is much smaller than the actual covariance. The consistency property of the filter is, however, still preserved since the estimated error covariance matrix is always larger than the actual one.

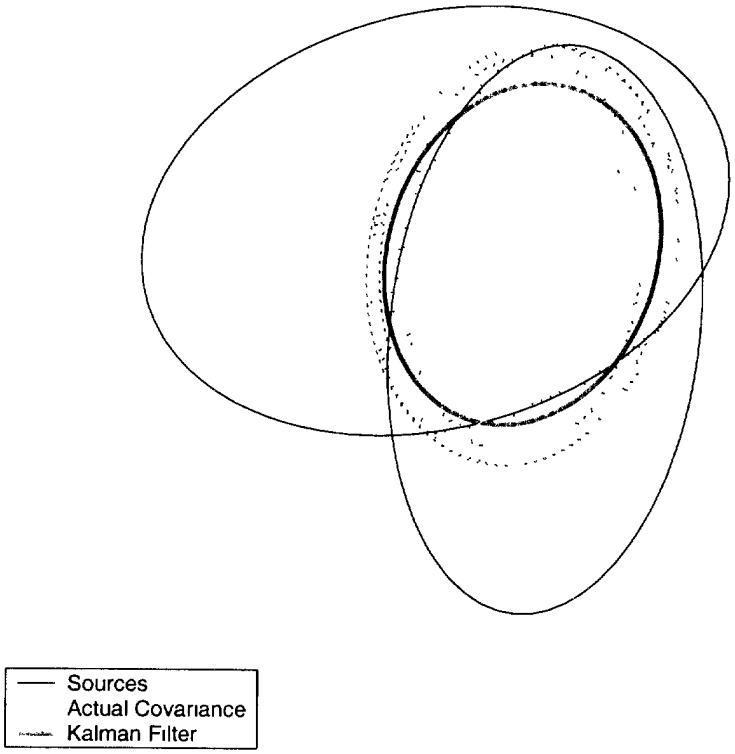


Figure A.1: *1_σ contour of the combined estimate yielded by the Kalman filter update rule and the actual covariance for two correlated (non-centred) sources, with variable correlation coefficient*

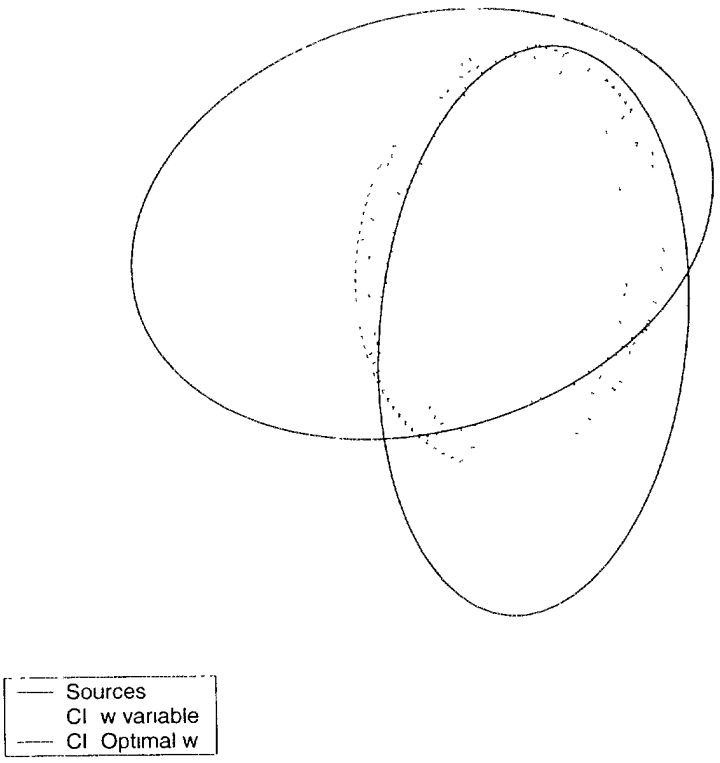


Figure A.2: *1_σ contour of the combined estimate yielded by the covariance intersection update rule (with variable ω) and the actual covariance for two correlated (non centred) sources with variable correlation coefficient*

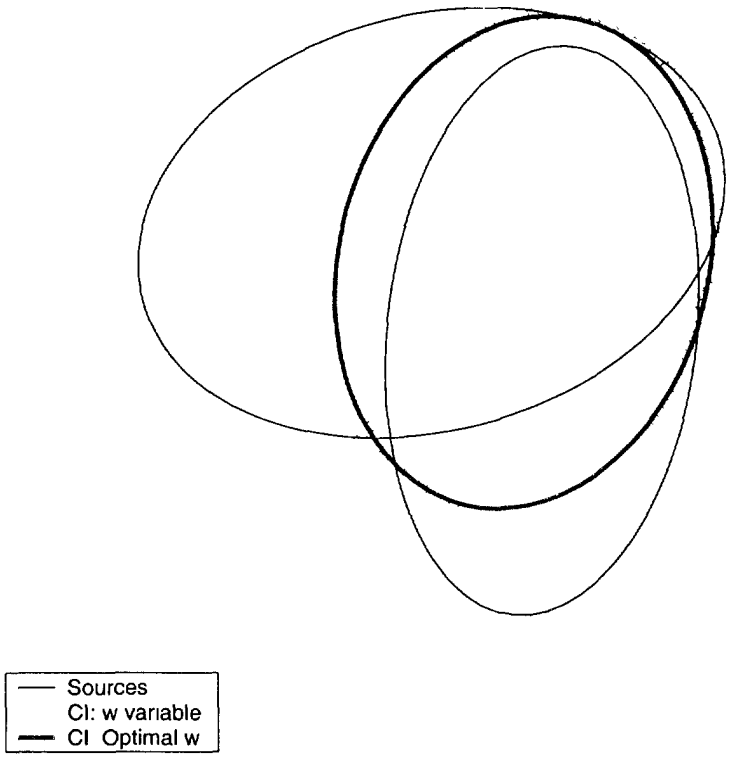


Figure A.3: *1_σ contour of the combined estimate yielded the covariance intersection update rule, with a variable parameter ω, for two (non-centred) sources*

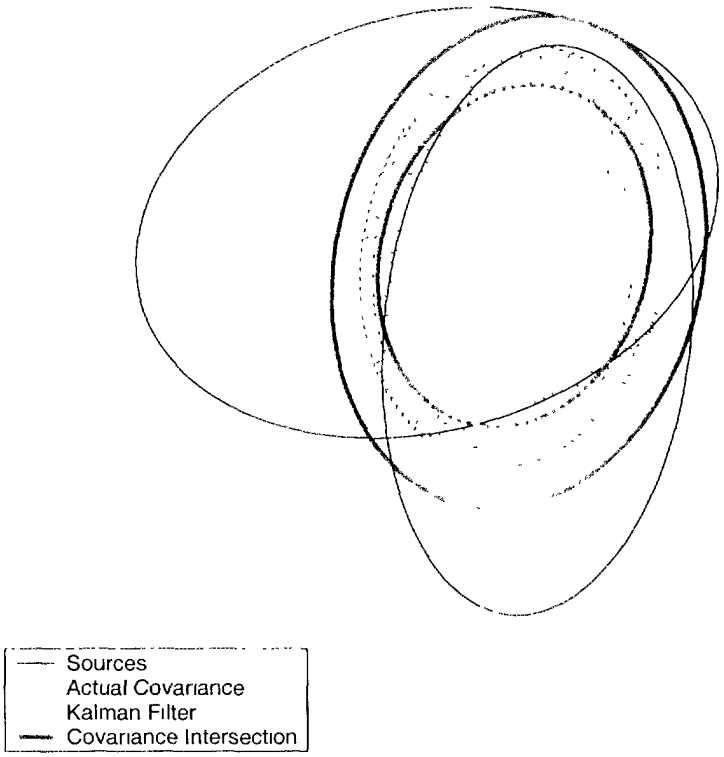


Figure A.4: *1_σ contour of the combined estimate yielded respectively by the Kalman filter and the covariance intersection update rules, and the actual covariance for two correlated (non centred) sources, with variable correlation coefficient*

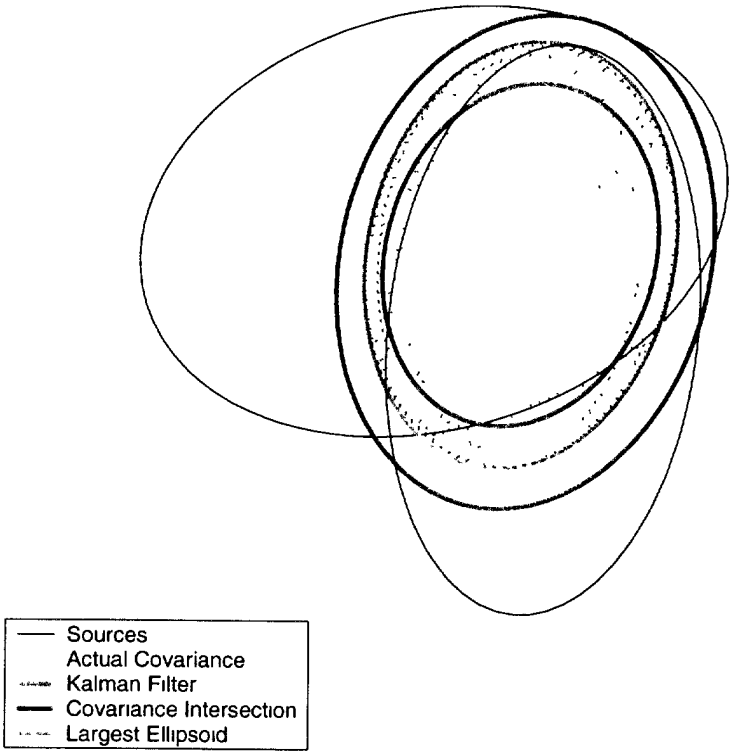


Figure A.5: *1_σ contour of the combined estimate, yielded by the largest ellipsoid vs the Kalman filter and the covariance intersection for two correlated (non-centred) sources*

Annex B
Optimization with respect to ω

The parameter ω offers, for the covariance intersection method, an additional degree of freedom, whose value can be determined to minimize a certain error norm of the resulting covariance matrix

$$\begin{aligned}\hat{\boldsymbol{P}}_{\hat{\boldsymbol{x}}}(\omega) &= \left[\omega\hat{\boldsymbol{P}}_{\hat{\boldsymbol{x}}_1}^{-1} + (1-\omega)\hat{\boldsymbol{P}}_{\hat{\boldsymbol{x}}_2}^{-1}\right]^{-1} \\ &= \left[\hat{\boldsymbol{P}}_{\hat{\boldsymbol{x}}_2}^{-1} + \omega\left(\hat{\boldsymbol{P}}_{\hat{\boldsymbol{x}}_1}^{-1} - \hat{\boldsymbol{P}}_{\hat{\boldsymbol{x}}_2}^{-1}\right)\right]^{-1} \\ &\triangleq \left[\boldsymbol{G}_0 + \omega\boldsymbol{G}_1\right]^{-1}\end{aligned}$$

B.1 Determinant optimization

For the covariance intersection method, and from the expression of $\hat{\boldsymbol{P}}_{\hat{\boldsymbol{x}}}$, the determinant appears to be the "natural" norm to be minimized. The underlying convex optimization problem[6] consists then in finding ω_{opt} such that

$$\begin{aligned}\omega_{opt} &= \arg\min_{\omega} \det\left(\hat{\boldsymbol{P}}_{\hat{\boldsymbol{x}}}\right) \\ &= \arg\min_{\omega} \ln\det\left(\hat{\boldsymbol{P}}_{\hat{\boldsymbol{x}}}\right) \\ &= \arg\min_{\omega} \varphi(\omega)\end{aligned}$$

where

$$\varphi(\omega) = -\ln\det\left(\boldsymbol{G}_0 + \omega\boldsymbol{G}_1\right)$$

with, as a constraint, $0 \leq \omega \leq 1$. Since $\boldsymbol{G}_0 = \boldsymbol{G}_0^T$ is definite positive (a covariance matrix), one can rewrite the cost function as follows

$$\begin{aligned}\varphi(\omega) &= -\ln\det\left[\boldsymbol{G}_0^{1/2}\left(\boldsymbol{I} + \omega\boldsymbol{G}_0^{-1/2}\boldsymbol{G}_1\boldsymbol{G}_0^{-1/2}\right)\boldsymbol{G}_0^{1/2}\right] \\ &= -\ln\det(\boldsymbol{G}_0) - \ln\det\left(\boldsymbol{I} + \omega\boldsymbol{G}_0^{-1/2}\boldsymbol{G}_1\boldsymbol{G}_0^{-1/2}\right)\end{aligned}$$

The last term, in the above equation, can be rewritten as

$$\begin{aligned}\ln\det\left(\boldsymbol{I} + \omega\boldsymbol{G}_0^{-1/2}\boldsymbol{G}_1\boldsymbol{G}_0^{-1/2}\right) &= \ln\left[\prod_{i=1}^{N}(1+\omega\lambda_i)\right] \\ &= \sum_{i=1}^{N}\ln(1+\omega\lambda_i)\end{aligned}$$

[6] Also known as *Analytic Centering* problem

where λ_i is the i^{th} eigenvalue of $G_0^{-1/2} G_1 G_0^{-1/2}$. This result allows for rewriting the cost function as

$$\varphi(\omega) = -\ln\det(G_0) - \sum_{i=1}^{N} \ln\left(1 + \omega\lambda_i\right)$$

The two first derivatives of φ, with respect to ω, can be obtained analytically, and there is no need to approximate them numerically.

Gradient

Indeed, the gradient of the cost function is given by

$$\begin{aligned} g(\omega) &= \frac{\partial\varphi}{\partial\omega} \\ &= -\sum_{i=1}^{N} \frac{\lambda_i}{1+\omega\lambda_i} \\ &= -\sum_{i=1}^{N} \frac{eig(G_0^{-1/2} G_1 G_0^{-1/2})}{eig(I + \omega G_0^{-1/2} G_1 G_0^{-1/2})} \\ &= -\sum_{i=1}^{N} eig\left[G_1 G_0^{-1/2} (I + \omega G_0^{-1/2} G_1 G_0^{1/2})^{-1} G_0^{-1/2}\right] \\ &= -\sum_{i=1}^{N} eig(G_1 \hat{P}_{\hat{x}}) \\ &= -Trace(G_1 \hat{P}_{\hat{x}}) \end{aligned}$$

Hessian

Similar calculations lead to the following Hessian expression

$$\begin{aligned} H(\omega) &= \frac{\partial^2\varphi}{\partial\omega^2} \\ &= \sum_{i=1}^{N} \left[\frac{\lambda_i}{1+\omega\lambda_i}\right]^2 \\ &= \sum_{i=1}^{N} eig\left[(G_1 \hat{P}_{\hat{x}})^2\right] \\ &= Trace\left[(G_1 \hat{P}_{\hat{x}})^2\right] \end{aligned}$$

Since the Hessian is always definite positive, the optimum will be a minimum (Second Order Necessary Condition).

Multisource fusion

In the case of a batch fusion, that is,

$$\hat{P}_{\hat{x}}^{-1}(\omega) = G_0 + \sum_{i=1}^{n} \omega_i G_i$$

The gradient and the Hessian will be given respectively by

$$g(i) = (\nabla\varphi)_i = -Trace(G_i \hat{P}_{\hat{x}})$$

and

$$H(i,j) = (\nabla^2\varphi)_{ij} = Trace(G_i \hat{P}_{\hat{x}} G_j \hat{P}_{\hat{x}})$$

Implementation

If there were no constraint on ω, an analytic solution to the above-stated optimization problem could have been found, by setting $g(\omega) = 0$. Since ω_{opt} must be within $[0, 1]$, a numerical solution will be used. The implementation details will be given later.

B.2 Trace minimization

The trace of the covariance matrix is another interesting cost function to be minimized. As in the determinant case, for practical reasons, the minimization will be applied to the logarithm of the trace. The optimization problem is then defined as follows

$$\begin{aligned}
\omega_{opt} &= \arg\min_{\omega} \operatorname{trace}\left(\hat{P}_{\hat{x}}\right) \\
&= \arg\min_{\omega} \operatorname{trace}\left[G_0 + \omega G_1\right]^{-1} \\
&= \arg\min_{\omega} \ln \operatorname{trace}\left[G_0 + \omega G_1\right]^{-1}
\end{aligned}$$

with, as a constraint, $0 \leq \omega \leq 1$. By using

$$\operatorname{trace} A^{-1} = \operatorname{trace} A \det A^{-1}$$

the minimization problem can be rewritten in the following form

$$\begin{aligned}
\omega_{opt} &= \arg\min_{\omega} \left\{ \ln \operatorname{trace}\left(G_0 + \omega G_1\right) - \ln \det\left(G_0 + \omega G_1\right) \right\} \\
&\triangleq \arg\min_{\omega} \varphi(\omega)
\end{aligned}$$

Gradient

$$
\begin{aligned}
g(\omega) &= \frac{\partial \varphi}{\partial \omega} \\
&= \frac{\partial \ln \operatorname{trace}(G_0 + \omega G_1)}{\partial \operatorname{trace}(G_0 + \omega G_1)} \frac{\partial \operatorname{trace}(G_0 + \omega G_1)}{\partial \omega} - \frac{\partial \ln (G_0 + \omega G_1)}{\partial \omega} \\
&= \operatorname{trace} G_1 \left[\operatorname{trace}(G_0 + \omega G_1)\right]^{-1} - \operatorname{trace} G_1 (G_0 + \omega G_1)^{-1} \\
&= \operatorname{trace} G_1 \left[\operatorname{trace} \hat{P}_{\hat{x}}^{-1}\right]^{-1} - \operatorname{trace} G_1 \hat{P}_{\hat{x}}
\end{aligned}
$$

Hessian

In the same way, the Hessian will be given by

$$
H(\omega) = -\left\{\operatorname{trace} G_1 \left[\operatorname{trace} \hat{P}_{\hat{x}}^{-1}\right]^{-1}\right\}^2 + \operatorname{trace}\left[(G_1 \hat{P}_{\hat{x}})^2\right]
$$

Multi-source fusion

As in the case of the determinant optimization, in the case of a batch fusion, the gradient and the Hessian will be given respectively by

$$
\begin{aligned}
g(i) &= (\nabla \varphi)_i \\
&= \operatorname{trace} G_i \left[\operatorname{trace} \hat{P}_{\hat{x}}^{-1}\right]^{-1} - \operatorname{trace}(G_i \hat{P}_{\hat{x}})
\end{aligned}
$$

and

$$
\begin{aligned}
H(i,j) &= (\nabla^2 \varphi)_{ij} \\
&= -\operatorname{trace} G_i \operatorname{trace} G_j \left[\operatorname{trace} \hat{P}_{\hat{x}}^{-1}\right]^{-2} + \operatorname{trace}(G_i \hat{P}_{\hat{x}} G_j \hat{P}_{\hat{x}})
\end{aligned}
$$

Implementation

Newton's method, with appropriate step length selection, can be used to efficiently compute ω_{opt}, starting from a feasible initial point. The following algorithm is considered

$$
\omega_{k+1} = \omega_k - \alpha_k H^{-1}(\omega_k) g(\omega_k)
$$

where α_k is the damping factor of the k^{th} iteration, which is given by [27]

$$
\alpha_k = \begin{cases} 1 & \text{if} \quad \delta_k \leq 1/4 \\ \dfrac{1}{1+\delta_k} & \text{otherwise} \end{cases}
$$

and

$$\delta_k = \sqrt{\boldsymbol{g}^T \boldsymbol{H}^{-1} \boldsymbol{g}}$$

is the Newton decrement of φ at ω.

Distribution list

INTERNAL DISTRIBUTION

TR 2001-223

1 - Director General

1 - Deputy Director General

3 - Document Library

1 - Head/DSS

1 - Head/IKM

1 - Head/SOS

1 - A. Benaskeur (author)

1 - J.M.J. Roy

1 - M. Blanchette

1 - S. Paradis

1 - M. Bélanger

1 - A. Guitouni

1 - I. Abi-Zeid

1 - J. Berger

1 - A. Boukhtouta

1 - R. Breton

1 - A.C. Bourry-Brisset

1 - M. Allouche

1 - H.Irandoust

1 - M. Gagnon

1 - P. Maupin

1 - C. Daigle

INTERNAL DISTRIBUTION (cont'd)

TR 2001-223

1 - A.L. Jousselme

1 - LCol C. Beaumont

1 - LCdr D. Morrissey

1 - Lt(N) E. Woodliffe

1 - Maj. G. Clairoux

1 - Maj. M. Gareau

1 - R. Charpentier

1 - D. Gouin

1 - G. Thibault

1 - J. Bédard

1 - J. Dumas

EXTERNAL DISTRIBUTION

TR 2001-223

1 - DRD KIM

1 - DRD KIM (unbound copy)

1 - DRDC

3 - DGOR

4 - DGJFD

2 - D ST CCIS

2 - D ST L

2 - D ST A

2 - D ST M

2 - D ST HP

2 - D MRS

1 - D MRS 6

1 - D MRS 6-2

2 - D DCEI

1 - D LCSPM

2 - DAR

1 - DAR-4

1 - DAR-3

2 - DMSS

2 - DMSS - 6

2 - DMSS - 8

1 - DSTA - 3 (3d)

1 - DSTM - 2 (1b)

EXTERNAL DISTRIBUTION (cont'd)

TR 2001-223

1 - DSTM - 3 (1a)

1 - DSTM - 5 (1c)

1 - DSTL - 2

2 - DLR

1 - ADM (IM)

1 - CFEC

7 - DRDC - Toronto:
Attn: R. Pigeau
K. Hendy
J. Baranski
F. Lichacz

4 - DRDC - Atlantic:
Attn: J.S. Kennedy
B. McArthur
D. Peters
LCdr B. MacLennan

2 - DRDC - Ottawa:
Attn: M. Rey
B. Bridgewater
P. Yansouni
C. McMillan

1 - PMO MHP

1 - PMO CADRE

1 - PMO AURORA

1 - Canadian Forces Command and Staff College Toronto
Attn: Commanding Officer

1 - CF Maritime Warfare School CFB Halifax
Halifax, Nova Scotia
Attn: Commanding Officer

SANS CLASSIFICATION
COTE DE SÉCURITÉ DE LA FORMULE
(plus haut niveau du titre, du résumé ou des mots-clefs)

FICHE DE CONTRÔLE DU DOCUMENT

1. PROVENANCE (le nom et l'adresse)	2. COTE DE SÉCURITÉ (y compris les notices d'avertissement, s'il y a lieu)
Abder Rezak Benaskeur **Centre de Recherche pour la Défense Valcartier** **2495, boul. Pie-Xi Nord** **Val-Bélair Qc, G3J 1X5 Canada**	

3. TITRE (Indiquer la cote de sécurité au moyen de l'abréviation (S, C, R ou U) mise entre parenthèses, immédiatement après le titre.)

A consistent filter for robust decentralized data fusion (U)

4. AUTEURS (Nom de famille, prenom et initiales. Indiquer les grades militaires, ex. Bleau, Maj. Louis E)

Abder Rezak Benaskeur et Jean Roy

5 DATE DE PUBLICATION DU DOCUMENT (mois et année)	6a. NOMBRE DE PAGES	6b NOMBRE DE REFERENCES
10/2001	**65**	**27**

7 DESCRIPTION DU DOCUMENT (La catégorie du document, par exemple rapport, note technique ou mémorandum Indiquer les dates lorsque le rapport couvre une periode définie.)

Rapport Technique

8 PARRAIN (le nom et l'adresse)

9a. NUMÉRO DU PROJET OU DE LA SUBVENTION (Spécifier si c'est un projet ou une subvention)	9b. NUMÉRO DE CONTRAT
11ba18	

10a. NUMÉRO DU DOCUMENT DE L'ORGANISME EXPÉDITEUR	10b. AUTRES NUMÉROS DU DOCUMENT
	N/A

11. ACCÈS AU DOCUMENT (Toutes les restrictions concernant une diffusion plus ample du document, autres que celles inhérentes à la cote de sécurité.)

- ☒ Diffusion illimitée
- ☐ Diffusion limitée aux entrepreneurs des pays suivants (spécifier)
- ☐ Diffusion limitée aux entrepreneurs canadiens (avec une justification)
- ☐ Diffusion limitée aux organismes gouvernementaux (avec une justification)
- ☐ Diffusion limitée aux ministères de la Défense
- ☐ Autres

12. ANNONCE DU DOCUMENT (Toutes les restrictions à l'annonce bibliographique de ce document. Cela correspond, en principe, aux données d'accès au document (11). Lorsqu'une diffusion supplémentaire (à d'autres organismes que ceux précisés à la case 11) est possible, on pourra élargir le cercle de diffusion de l'annonce.)

SANS CLASSIFICATION
COTE DE LA SÉCURITÉ DE LA FORMULE
(plus haut niveau du titre, du résumé ou des mots-clefs)

dcd03f ver (06-1999)

SANS CLASSIFICATION

COTE DE LA SÉCURITÉ DE LA FORMULE
(plus haut niveau du titre, du resumé ou des mots-clefs)

13. SOMMAIRE (Un résumé clair et concis du document. Les renseignements peuvent aussi figurer ailleurs dans le document Il est souhaitable que le sommaire des documents classifiés soit non classifié Il faut inscrire au commencement de chaque paragraphe du sommaire la cote de sécurité applicable aux renseignements qui s'y trouvent, à moins que le document lui-même soit non classifié Se servir des lettres suivantes: (S), (C), (R) ou (U) Il n'est pas nécessaire de fournir ici des sommaires dans les deux langues officielles à moins que le document soit bilingue)

14 MOTS-CLÉS, DESCRIPTEURS OU RENSEIGNEMENTS SPECIAUX (Expressions ou mots significatifs du point de vue technique, qui caractérisent un document et peuvent aider a le cataloguer. Il faut choisir des termes qui n'exigent pas de cote de sécurité Des renseignements tels que le modèle de l'equipement, la marque de fabrique, le nom de code du projet militaire, la situation geographique, peuvent servir de mots-cles. Si possible, on doit choisir des mots-clés d'un thésaurus, par exemple le "Thesaurus of Engineering and Scientific Terms (TESTS)" Nommer ce thésaurus. Si l'on ne peut pas trouver de termes non classifiés, il faut indiquer la classification de chaque terme comme on le fait avec le titre)

518345

CA 56-529

SANS CLASSIFICATION

COTE DE SÉCURITÉ DE LA FORMULE
(plus haut niveau du titre, du résumé ou des mots-clefs)

dcd03f ver (06-1999)