July 2006 Volume 1 Number 1 ISSN 1557-6418

JAIF 06

Journal of Advances in Information Fusion

A semi-annual archival publication of the International Society of Information Fusion

Regular Papers **Page**

From the Editor-In-Chief

With this inaugural issue, ISIF introduces JAIF as its flagship journal. JAIF is a peer-reviewed journal that...

ISIF

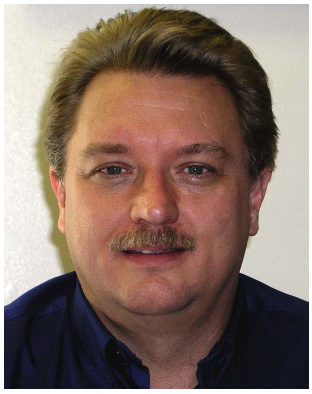
INTERNATIONAL SOCIETY OF INFORMATION FUSION

The International Society of Information Fusion (ISIF) seeks to be the premier professional society and global information resource for multidisciplinary approaches for theoretical and applied INFORMATION FUSION technologies. Technical areas of interest include target tracking, detection theory, applications for information fusion methods, image fusion, fusion systems architectures and management issues, classification, learning, data mining, Bayesian and reasoning methods.

Journal of Information Advances in Information Fusion (ISSN 1557-6418) is published semi-annually by the International Society of Information Fusion. The responsibility for the contents rests upon the authors and not upon ISIF, the Society, or its members. ISIF is California Nonprofit Public Benefit Corporation at P.O. Box 4631, Mountain View, California 94040.

From the Editor in Chief:

July 2006



Moore's law tells us that the number of semiconductor devices on an integrated circuit doubles every 24 months and this law has reliably predicted that growth since 1960 [1]. This growth in semiconductor density has led to the popular interpretation of Moore's law as the computer capability and memory double every 18 months [2]. This rapid growth in computing and data storage has already given us the ability to collect more information than a human can process in many applications, and it appears that the trend will not end soon. Some are predicting the continued shrinkage in memory devices to the size of a white blood cell in 2020 [3]. Furthermore, systems-on-package promises to leapfrog Moore's Law by combining ICs with micrometer-scale thin-film versions of discrete components and embedding them in a new type of package [4]. With this growth in computing and memory in smaller packages, sensors are becoming cheaper and more plentiful allowing the visions of more sophisticated and automated systems to be realized, and effective information processing plays a critical role in those systems. Examples of that processing might include rapid and efficient distillation of a massive data set to a few salient features or fusion of the information from multiple sources into a common representational form.

With the pressing demand and interest in information processing methods, scientists and engineers organized the first International Conference on Information Fusion (Fusion 1998) in Las Vegas, Nevada on July 6–9, 1998. With 146 papers and 161 attendees, the first Fusion conference was considered a great success, and the International Society for Information Fusion (ISIF) was founded to be the premier global information community for multidisciplinary approaches for theoretical and applied information fusion technologies. The second Fusion conference was an even greater success with 190 papers and 211 attendees. After strong sponsorship by ONERA and Thomson-CSF in 2000 and Lockheed-Martin Canada in 2001, the fourth Fusion conference became a self-supporting conference in 2002.

After five years of success of Fusion conferences as summarized in the accompanying table, the ISIF

Board of Directors voted in 2003 to establish a peer-reviewed archival journal in the area of information fusion in the name of Journal of Advances in Information Fusion (JAIF). In 2003, editors were appointed for JAIF and a web-based review system was established at http://jaif.msubmit.net to handle the peer review of manuscripts electronically. This system facilitates the review of manuscripts for authors and archives the reviewers' comments and editorial decisions for all manuscripts. Researchers are invited to volunteer to be a referee by registering as an author at the web site.

With this inaugural issue, ISIF introduces JAIF as its flagship journal. JAIF is a peer-reviewed, semi-annual archival journal that will be published electronically and distributed via the internet. JAIF has established high standards for the peer review process through an editorial board with strong academic and industrial backgrounds. Prior to publication, each manuscript will require a review from at least three referees and manuscript corrections that address any shortcoming identified by the referees and editors. The inside of the front cover gives the scope of JAIF and the editorial board and their associated technical areas. Additional information on the editorial process and board is available at http://www.isif.org.

William Dale Blair
Editor in Chief

Ten Years of the International Conference on Information Fusion (Fusion)

Year	Location, Date Chair	Number of Papers	Number of Attendees
1st	Las Vegas, Nevada, July 6–9, 1998 Dongping Daniel Zhu, Zaptron Systems, Inc.	146	161
2nd	Sunnyvale, California, July 6–8, 1999 Dongping Daniel Zhu, Zaptron Systems, Inc.	190	211
3rd	Paris, France, July 10–13, 2000 Jean Dezert, ONERA	173	253
4th	Montréal, Quebec, Canada, August 7–10, 2001 Pierre Valin, Lockheed-Martin, Canada	146	261
5th	Annapolis, Maryland, July 7–11, 2002 X. Rong Li, University of New Orleans	232	289
6th	Cairns, Queensland, Australia, July 8–11, 2003 Subhash Challa, University of Technology, Sydney	204	256
7th	Stockholm, Sweden, June 28–July 1, 2004 Per Svensson, Swedish Defence Research Agency	171	300
8th	Philadelphia, Pennsylvania, July 25–29, 2005 John Sudano, Lockheed-Martin	181	332
9th	Florence, Italy, July 10–13, 2006 Stefano Coraluppi, NATO Undersea Research Ctr. Peter Willett, University of Connecticut	263	397
10th	Quebec City, Quebec, Canada, July 9–12, 2007 Eloi Bossé, DRDC Valcartier	—	—

REFERENCES

[1] R. Schaller
Moore's law: Past, present, and future.
IEEE Spectrum, June 1997, 53–59.

[2] Moore's Law
www.wikipedia.org.
Jan. 31, 2007.

[3] P. Ball
A switch in time.
Nature, Vol. 445, Jan. 25, 2007, 362–363.

[4] R. R. Tummaia
Moore's law meets its match.
IEEE Spectrum, June 2006, 44–49.

Multisensor Track-to-Track Association for Tracks with Dependent Errors

Y. BAR-SHALOM
University of Connecticut

H. CHEN
University of New Orleans

The problem of track-to-track association has been considered until recently in the literature only for pairwise associations. In view of the extensive recent interest in multisensor data fusion, the need to associate simultaneously multiple tracks has arisen. This is due primarily to bandwidth constraints in real systems, where it is not feasible to transmit detailed measurement information to a fusion center but, in many cases, only local tracks. As it has been known in the literature, tracks of the same target obtained from independent sensors are still dependent due to the common process noise [2]. This paper derives the exact likelihood function for the track-to-track association problem from multiple sources, which forms the basis for the cost function used in a multidimensional assignment algorithm that can solve such a large scale problem where many sensors track many targets. While a recent work [14] derived the likelihood function under the assumption that the track errors are independent, the present paper incorporates the (unavoidable) dependence of these errors.

Manuscript received November 11, 2004; revised September 21, 2005; recommended by Editor Dr. Shozo Mori.

Authors' adresses: Y. Bar-Shalom, Dept. of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269-2157, E-mail: (ybs@ee.uconn.edu); H. Chen, Dept. of Electrical Engineering, University of New Orleans, New Orleans, LA 70148, E-mail: (hchen2@uno.edu).

1. INTRODUCTION

In this paper we consider the problem of associating tracks represented by their local estimates and covariances from S sources. These sources are processors that use data from corresponding local sensor systems. While different sensors have, typically, independent measurement errors, the local state estimation errors for the same target will be dependent due to the common prior or process noise. This dependence is characterized by the crosscovariances of the local estimation errors—see [2], Sec. 8.4. The association presented in [2], while accounting for the crosscovariances of the track errors, is limited to pairs of tracks, i.e., it is suitable for the situation of two lists (sources) of tracks. Consequently, if this is used when there are more than two lists of local tracks, the results will depend on the order in which the lists are considered. This order-dependence can be avoided only by simultaneous consideration of all S-tuples when there are S lists.

While a recent work [14] derived the likelihood function under the assumption that the state estimation errors are independent, the present paper incorporates the (unavoidable) dependence of these errors. Earlier work on fusion of multiple tracks can be found in [13]. This work also addressed the issue of dependence among tracks due to prior communication. The general fusion of crosscorrelated tracks was derived in [11]. A recent comparison of different fusion techniques can be found in [15].

The goal of this paper is to derive a likelihood-ratio based cost function suitable for the use of a multidimensional assignment (S-D, see, e.g., [3], Ch. 2) to decide which tracks should be fused. The cost function should allow simultaneous consideration of S tracks corresponding to the same target (one from each source) or any subset of this.

First we shall derive the likelihood function of the hypothesis that S tracks are from the same target, i.e., that they have a common origin. This derivation is based on [17] where it was presented for the purpose of sensor bias estimation for $S = 2$ sensors and it accounted for the dependence of the track estimation errors across sensors. More recently [14] developed the likelihood function for the association of tracks from an arbitrary number of sensors, but under the assumption that their track (local state estimation) errors are independent. This assumption, however, is not satisfied when the target state equations have process noise which is necessary to model motion uncertainty.

These likelihood functions are, however, not dimensionless since they are joint pdfs (probability density functions) of state vectors. As indicated in [4], Sec. 1.4.2, the pdf of a vector consisting of position and velocity in an n-dimensional Cartesian space has its physical dimension given by the inverse of the product of the physical dimensions of its components, i.e., $(\text{length})^{-n} \cdot (\text{length/time})^{-n}$. Consequently,

the joint pdf of S such vectors has physical dimension of $(\text{length})^{-2Sn} \cdot (\text{time})^{Sn}$. Therefore, one cannot compare the likelihood functions of the hypothesis that S tracks have a common origin with the hypothesis that, say, a subset of them, consisting of $M < S$ tracks, have a common origin, i.e, one has an incompatibility. The remedy for this incompatibility problem is to use dimensionless likelihood ratios obtained by dividing a common-origin likelihood function with the likelihood function of the hypothesis that these tracks are all of different origin. The latter likelihood function will consist of a diffuse pdf (a uniform distribution in the augmented (product) state space—see [4], Sec. 2.3.4), as detailed later. Using these likelihood ratios one can compare all the hypotheses regardless of how many tracks of common origin are assumed in them.

The methodology of this paper, based on likelihood ratios and a diffuse prior for the target state estimates, allows for a systematic way of handling incomplete track-to-track associations across sensors and was presented in preliminary form in [6]. Subsequently, an application to a practical problem was given in [1].

The rest of the paper is organized as follows. The likelihood function of a set of tracks is derived in Section 2. The likelihood ratios for the track-to-track association are presented in Section 3. The assignment with the negative log-likelihood ratios as cost function is discussed in Section 4. An investigation of the assignment accuracy, the sensitivity to the crosscorrelation, and a tracking example are presented in Section 5. Conclusions are in Section 6.

2. THE LIKELIHOOD FUNCTION OF A SET OF TRACKS

Consider the situation where there are S sensors, each with its list of tracks represented by the estimates $\hat{x}_i^{j_i}$ in the same state space, with errors that are zero-mean jointly Gaussian with covariances $P_i^{j_i}$, $i = 1,\ldots,S$, pertaining to a common time (not indicated for simplicity), where subscript i denotes the sensor based on whose data the (local) track has been obtained and superscript $j_i = 1,\ldots,N_i$ denotes the indices of the tracks at sensor i. The error crosscovariances for tracks representing the same target are discussed later.

The likelihood function of the common origin hypothesis $\mathcal{H}_{l_1,\ldots,l_S}$ for the tracks represented by the local estimates $\hat{x}_i^{l_i}$, $i = 1,\ldots,S$, i.e., that they represent the same target is the joint pdf of the "track data" conditioned on the hypothesis

$$\Lambda(\mathcal{H}_{l_1,\ldots,l_S}) = p(\hat{x}_S^{l_S},\ldots,\hat{x}_1^{l_1} \mid \mathcal{H}_{l_1,\ldots,l_S}). \tag{1}$$

Note that in the above we use the fact that the track estimates are sufficient statistics—a consequence of the Gaussian assumption. On the other hand, there is no assumption of independence of the track estimation errors. As it is known, the estimation errors for the same target obtained at independent sensors (with the measurement noises independent across the sensors) are correlated and this is quantified by the crosscovariance matrices (see [2], Sec. 8.4.2). Otherwise, these errors are assumed independent.

The likelihood function (1) can be rewritten by moving the first (or any other) track estimate into the conditioning set, as follows

$$\Lambda(\mathcal{H}_{l_1,\ldots,l_S}) = p(\hat{x}_S^{l_S},\ldots,\hat{x}_2^{l_2} \mid \mathcal{H}_{l_1,\ldots,l_S},\hat{x}_1^{l_1})p(\hat{x}_1^{l_1} \mid \mathcal{H}_{l_1,\ldots,l_S}). \tag{2}$$

Since $\mathcal{H}_{l_1,\ldots,l_S}$ does not contribute any information to the marginal pdf of a single track, it can be dropped from the last term above. Furthermore, the marginal pdf of a track estimate can be taken as diffuse (uniformly distributed in a region of the state space $\mathcal{V}$, whose volume is V, assumed large enough to qualify for a diffuse prior), i.e.,

$$p(\hat{x}_1^{l_1} \mid \mathcal{H}_{l_1,\ldots,l_S}) = p(\hat{x}_1^{l_1}) = \frac{1}{V} \tag{3}$$

because, in the absence of any information (which is our assumption), a track estimate can be anywhere in the state space. This is in accordance to Bayes' postulate [8, 10]. The diffuse prior has to have a support only "sufficiently larger" than the estimates' pdf. Furthermore, this diffuse prior assumption is only for the marginal (unconditional) pdf of a track estimate. The conditional pdf of any track estimate given another estimate with the same origin is not diffuse anymore and is determined by the statistical properties of their estimation errors which are not assumed independent—their correlation can be due to the common process noise as well as to a common prior.

With this, (2) becomes

$$\Lambda(\mathcal{H}_{l_1,\ldots,l_S}) = \frac{1}{V} p(\hat{x}_S^{l_S},\ldots,\hat{x}_2^{l_2} \mid \mathcal{H}_{l_1,\ldots,l_S},\hat{x}_1^{l_1}). \tag{4}$$

Note that V^{-1}, while having a physical dimension (that makes (4) have the same dimension as (1)), is really a constant whose exact value only scales the final result.

Consider first the case of common origin of two tracks, l_i and l_j from sensors i and j, respectively. Now, under the Gaussian assumption, if $\hat{x}_i^{l_i}$ originated from the same target as $\hat{x}_j^{l_j}$, then, with the diffuse prior assumption, one has (see Appendix; this result was presented in [2], Sec. 8.3.3, but without proof)

$$E[\hat{x}_i^{l_i} \mid \mathcal{H}_{l_i,l_j},\hat{x}_j^{l_j}] = \hat{x}_j^{l_j} \tag{5}$$

and

$$\begin{aligned} &E[(\hat{x}_i^{l_i} - \hat{x}_j^{l_j})(\hat{x}_j^{l_j} - \hat{x}_j^{l_j})' \mid \mathcal{H}_{l_i,l_j},\hat{x}_j^{l_j}] \\ &\quad = E[(\hat{x}_i^{l_i} - x^l - (\hat{x}_j^{l_j} - x^l))(\hat{x}_i^{l_i} - x^l - (\hat{x}_j^{l_j} - x^l))' \mid \mathcal{H}_{l_i,l_j}] \\ &\quad = P_i^{l_i} + P_j^{l_j} - P_{i,j}^{l_i,l_j} - (P_{i,j}^{l_i,l_j})' \end{aligned} \tag{6}$$

where x^l is the common true state of these tracks, which is irrelevant. The crosscovariance $P_{i,j}^{l_i,l_j}$ is given by a Lyapunov type recursion (see [2], Sec. 8.4).[1]

The general likelihood function (4) for common origin of the tracks $l_1,\ldots,l_S$ is obtained as follows. The pdf from (4) can be written as

$$p(\hat{x}_S^{l_S},\ldots,\hat{x}_2^{l_2} \mid \mathcal{H}_{l_1,\ldots,l_S},\hat{x}_1^{l_1}) = \mathcal{N}\left(\begin{bmatrix}\hat{x}_2^{l_2}\\ \vdots \\ \hat{x}_S^{l_S}\end{bmatrix}; \begin{bmatrix}\hat{x}_1^{l_1}\\ \vdots \\ \hat{x}_1^{l_1}\end{bmatrix}, \begin{bmatrix} E[(\hat{x}_2^{l_2}-\hat{x}_1^{l_1})(\hat{x}_2^{l_2}-\hat{x}_1^{l_1})' \mid \mathcal{H}_{l_1,\ldots,l_S}] & \cdots & E[(\hat{x}_2^{l_2}-\hat{x}_1^{l_1})(\hat{x}_S^{l_S}-\hat{x}_1^{l_1})' \mid \mathcal{H}_{l_1,\ldots,l_S}] \\ \cdots & \cdots & \cdots \\ E[(\hat{x}_S^{l_S}-\hat{x}_1^{l_1})(\hat{x}_2^{l_2}-\hat{x}_1^{l_1})' \mid \mathcal{H}_{l_1,\ldots,l_S}] & \cdots & E[(\hat{x}_S^{l_S}-\hat{x}_1^{l_1})(\hat{x}_S^{l_S}-\hat{x}_1^{l_1})' \mid \mathcal{H}_{l_1,\ldots,l_S}] \end{bmatrix}\right). \tag{10}$$

Thus for tracks l_i and l_j one has

$$\begin{aligned} p(\hat{x}_i^{l_i} \mid \mathcal{H}_{l_i,l_j},\hat{x}_j^{l_j}) &= \mathcal{N}[\hat{x}_i^{l_i};\hat{x}_j^{l_j},P_i^{l_i}+P_j^{l_j}-P_{i,j}^{l_i,l_j}-(P_{i,j}^{l_i,l_j})'] \\ &= \mathcal{N}[\hat{x}_i^{l_i}-\hat{x}_j^{l_j};0,P_i^{l_i}+P_j^{l_j}-P_{i,j}^{l_i,l_j}-(P_{i,j}^{l_i,l_j})'] \end{aligned} \tag{7}$$

where $\mathcal{N}[x;\bar{x},P]$ denotes the Gaussian pdf with argument x, mean $\bar{x}$ and covariance P. Then the joint likelihood function of common origin for the tracks l_i and l_j is

$$\begin{aligned} \Lambda(\mathcal{H}_{l_i,l_j}) &= \frac{1}{V}p(\hat{x}_i^{l_i} \mid \mathcal{H}_{l_i,l_j},\hat{x}_j^{l_j}) \\ &= \frac{1}{V}\mathcal{N}[\hat{x}_i^{l_i}-\hat{x}_j^{l_j};0,P_i^{l_i}+P_j^{l_j}-P_{i,j}^{l_i,l_j}-(P_{i,j}^{l_i,l_j})']. \end{aligned} \tag{8}$$

Note that the test statistic (normalized distance squared)

$$D(\hat{x}_i^{l_i},\hat{x}_j^{l_j}) = (\hat{x}_i^{l_i}-\hat{x}_j^{l_j})'[P_i^{l_i}+P_j^{l_j}-P_{i,j}^{l_i,l_j}-(P_{i,j}^{l_i,l_j})']^{-1}(\hat{x}_i^{l_i}-\hat{x}_j^{l_j}) \tag{9}$$

has been known in the literature for some time (e.g., [2], Sec. 8.4.3) for the association of pairs of tracks.[2] While originally this distance was introduced heuristically, it can be seen to follow directly from (8) as a likelihood test. The first rigorous derivation of (9) was given in [17] in the context of sensor bias estimation. The derivation given above is, however, much simpler and, more importantly, it generalizes to S tracks.

Then, similarly to (7), the mean is shifted into the argument and this yields the likelihood function

$$\Lambda(\mathcal{H}_{l_1,\ldots,l_S}) = \frac{1}{V}\mathcal{N}[\hat{\mathbf{x}}_{1,S};0,\mathbf{P}_{1,S}] \tag{11}$$

where

$$\hat{\mathbf{x}}_{1,S} \triangleq \begin{bmatrix}\hat{x}_2^{l_2}-\hat{x}_1^{l_1}\\ \vdots \\ \hat{x}_S^{l_S}-\hat{x}_1^{l_1}\end{bmatrix} \tag{12}$$

is a stacked $(S-1)n_x$ vector (with n_x the dimension of x), whose covariance, defined within (10) has the diagonal blocks

$$\begin{aligned} (\mathbf{P}_{1,S})_{i-1,i-1} &= E[(\hat{x}_i^{l_i}-\hat{x}_1^{l_1})(\hat{x}_i^{l_i}-\hat{x}_1^{l_1})' \mid \mathcal{H}_{l_1,\ldots,l_S}] \\ &= P_1^{l_1}+P_i^{l_i}-P_{1,i}^{l_1,l_i}-(P_{1,i}^{l_1,l_i})', \\ & \qquad i=2,\ldots,S \end{aligned} \tag{13}$$

and the offdiagonal blocks

$$\begin{aligned} (\mathbf{P}_{1,S})_{i-1,j-1} &= E[(\hat{x}_i^{l_i}-\hat{x}_1^{l_1})(\hat{x}_j^{l_j}-\hat{x}_1^{l_1})' \mid \mathcal{H}_{l_1,\ldots,l_S}] \\ &= P_1^{l_1}-P_{1,j}^{l_1,l_j}-(P_{1,i}^{l_1,l_i})'+P_{i,j}^{l_i,l_j}, \\ & \qquad i,j=2,\ldots,S. \end{aligned} \tag{14}$$

Note that with the (invertible) transformation

$$\hat{\mathbf{y}}_{1,S} = \begin{bmatrix} 1 & -1 & 0 & \cdots \\ 0 & 1 & -1 & 0 \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & 0 & 1 & -1 \\ \cdots & 0 & 0 & 1 \end{bmatrix} \hat{\mathbf{x}}_{1,S} = \begin{bmatrix} \hat{x}_2^{l_2}-\hat{x}_3^{l_3} \\ \hat{x}_3^{l_3}-\hat{x}_4^{l_4} \\ \vdots \\ \hat{x}_{S-1}^{l_{S-1}}-\hat{x}_S^{l_S} \\ \hat{x}_S^{l_S}-\hat{x}_1^{l_1} \end{bmatrix} \tag{15}$$

one can see that (11) is really symmetric in the sense that it has an equivalent symmetric form even if it appears not to be symmetric at first sight. This is due to the fact that the determinant of the above transformation is unity.

[1] Previous communication is difficult to account for in the correlation but not impossible—this would require restarting (after every communication) the iteration of the Lyapunov equation (8.4.2-3) in [2] that yields the crosscovariance.

[2] The importance of using the crosscovariances is twofold: ignoring the crosscorrelations (which are positive, as discussed in Section 5) the distance (9) is smaller than it should be and the covariance of the fused estimate is optimistic (see [2], Sec. 8.4.5).

REMARKS Note that the expression of the likelihood function (11) follows from the way in which (4) is written, namely as the joint pdf of the local track estimates from sensors $S,\ldots,2$ (written for convenience with the indices decreasing) conditioned on the track estimate from sensor 1. Equation (4) can be rewritten in the chain rule form as

$$\Lambda(\mathcal{H}_{l_1,\ldots,l_S}) = \frac{1}{V}\prod_{i=2}^{S} p(\hat{x}_i^{l_i} \mid \hat{x}_{i-1}^{l_{i-1}},\ldots,\hat{x}_2^{l_2},\hat{x}_1^{l_1}) = \frac{1}{V}\prod_{i=2}^{S} p(\hat{x}_i^{l_i} \mid \hat{x}^{F^{i-1}}) \tag{16}$$

where $\hat{x}^{F^{i-1}}$ is the fused state estimate from the first $i-1$ local tracks.

It was this last form that was derived in [14] under the assumption that the local track errors are uncorrelated. While (16) holds also for correlated tracks since no uncorrelatedness assumption was needed in its derivation above, its evaluation is relatively simple only under the assumption that the local track errors are uncorrelated. Otherwise, for the realistic situation of correlated track errors it becomes quite complicated. Consequently, expression (11) is believed to be the practical one when the crosscovariances are taken into consideration.

Note that the local track estimate from sensor 1 is chosen in the conditioning of (2) only for notational simplicity. One can use any local estimate as the reference track to obtain (11) with similar derivation.

3. THE LIKELIHOOD RATIOS FOR GENERAL TRACK-TO-TRACK ASSOCIATION

The likelihood ratio of the common origin hypothesis $\mathcal{H}_{l_1,\ldots,l_S}$ for the tracks represented by the local estimates $\hat{x}_i^{l_i}$, $i = 1,\ldots,S$, i.e., that all these tracks represent the same target is obtained next. The numerator is given by (11) while the denominator, which is the likelihood of all being of different origin (hypothesis $\bar{\mathcal{H}}_{l_1,\ldots,l_M}$), is obtained in a similar manner to (2) as follows

$$\Lambda(\bar{\mathcal{H}}_{l_1,\ldots,l_S}) = p(\hat{x}_S^{l_S},\ldots,\hat{x}_2^{l_2} \mid \bar{\mathcal{H}}_{l_1,\ldots,l_S},\hat{x}_1^{l_1}) p(\hat{x}_1^{l_1} \mid \bar{\mathcal{H}}_{l_1,\ldots,l_S}) = \prod_{s=2}^{S} p(\hat{x}_s^{l_s} \mid \bar{\mathcal{H}}_{l_1,\ldots,l_S},\hat{x}_1^{l_1}) p(\hat{x}_1^{l_1} \mid \bar{\mathcal{H}}_{l_1,\ldots,l_S}). \tag{17}$$

Analogously to (3),

$$p(\hat{x}_1^{l_1} \mid \bar{\mathcal{H}}_{l_1,\ldots,l_S}) = p(\hat{x}_1^{l_1}) = \frac{1}{V}. \tag{18}$$

As shown in [7], [10], the role of the pdf of a false/extraneous measurement in the likelihood ratio is played by the spatial density of these measurements under the assumption that they are Poisson distributed. This was obtained from the rigorous Bayesian derivation of the Multiple Hypothesis Tracker. Consequently, assuming the extraneous tracks in the present problem to be Poisson distributed in the state space with spatial density[3] μ_{ex}, one has

$$p(\hat{x}_s^{l_s} \mid \bar{\mathcal{H}}_{l_1,\ldots,l_S},\hat{x}_1^{l_1}) = \mu_{\text{ex}}. \tag{19}$$

Using (18) and (19) in (17) yields

$$\Lambda(\bar{\mathcal{H}}_{l_1,\ldots,l_S}) = \frac{\mu_{\text{ex}}^{S-1}}{V}. \tag{20}$$

Finally, combining the above with (11) yields the likelihood ratio

$$\mathcal{L}(\mathcal{H}_{l_1,\ldots,l_S} : \bar{\mathcal{H}}_{l_1,\ldots,l_S}) = \frac{\Lambda(\mathcal{H}_{l_1,\ldots,l_S})}{\Lambda(\bar{\mathcal{H}}_{l_1,\ldots,l_S})} = \frac{\frac{1}{V}\mathcal{N}[\hat{\mathbf{x}}_{1,S};0,\mathbf{P}_{1,S}]}{\frac{1}{V}\mu_{\text{ex}}^{S-1}} = \frac{\mathcal{N}[\hat{\mathbf{x}}_{1,S};0,\mathbf{P}_{1,S}]}{\mu_{\text{ex}}^{S-1}} \tag{21}$$

which is, clearly, a dimensionless quantity.

Next consider the likelihood ratio of an incomplete assignment consisting of tracks from the lists corresponding to the subset of sensors with indices $\mathcal{S}_i = \{s_1,s_2,\ldots,s_M\}$, where $1 \le s_1 < s_2 < \cdots < s_M \le S$. The entire set of list (sensor) indices is denoted as $\mathcal{S}$.

Assume that the probability of a target having a ("detected") track in the list of sensor s is P_{D_s} and that these track detection events are independent across sensors.[4]

Then the likelihood ratio of this assignment is [7]

$$\begin{aligned}\mathcal{L}(\mathcal{H}_{l_{s_1},\ldots,l_{s_M}} : \bar{\mathcal{H}}_{l_{s_1},\ldots,l_{s_M}}) &= V^{-1}\mathcal{N}[\hat{\mathbf{x}}_{\mathcal{S}_i};0,\mathbf{P}_{\mathcal{S}_i}]\left[\prod_{s\in\mathcal{S}_i} P_{D_s}\right]\mu_{\text{ex}}^{-S+M} \\ &\quad\times\left[\prod_{s\in\bar{\mathcal{S}}_i}(1-P_{D_s})\right][V^{-1}\mu_{\text{ex}}^{-S+1}]^{-1} \\ &= \mu_{\text{ex}}^{M-1}\mathcal{N}[\hat{\mathbf{x}}_{\mathcal{S}_i};0,\mathbf{P}_{\mathcal{S}_i}]\left[\prod_{s\in\mathcal{S}_i} P_{D_s}\right]\left[\prod_{s\in\bar{\mathcal{S}}_i}(1-P_{D_s})\right].\end{aligned} \tag{22}$$

The above follows by including in the numerator the probabilities of the events (assumed independent) that the tracks belonging to the hypothesized target have been detected by the sensors in $\mathcal{S}_i$ but not by the

[3] Since the true targets are typically not homogeneously distributed in the space, this should be taken as the local density of the extraneous (true and false) tracks.

[4] This is clearly only a convenient mathematical assumption—in practice the situation can be much more complex: these probabilities depend on the target locations, sensor modes, their fields of view, obscuration conditions, etc.

sensors in $\bar{\mathcal{S}}_i$. For the tracks corresponding to the sensors in $\bar{\mathcal{S}}_i$, their pdfs are the "extraneous" ones, μ_{ex}. In the denominator we have the probability densities of the tracks assuming they are not of common origin, modeled as having pdfs μ_{ex}. The pdfs of the tracks corresponding to the sensors in $\bar{\mathcal{S}}_i$ cancel between the numerator and denominator.

The first argument of the Gaussian density in (22) is, similarly to (12), given by

$$\hat{\mathbf{x}}_{\mathcal{S}_i} \stackrel{\Delta}{=} \begin{bmatrix} \hat{x}_{s_2}^{l_{s_2}} - \hat{x}_{s_1}^{l_{s_1}} \\ \vdots \\ \hat{x}_{s_M}^{l_{s_M}} - \hat{x}_{s_1}^{l_{s_1}} \end{bmatrix} \tag{23}$$

and $\mathbf{P}_{\mathcal{S}_i}$ is its covariance matrix with blocks given by expressions similar to (13)–(14).

4. THE USE OF THE LIKELIHOOD RATIOS IN ASSIGNMENT

We first consider the assignment formulation for track-to-track association from two sensors. Assume sensor 1 has a list of N_1 tracks and sensor 2 has a list of N_2 tracks. Define the binary assignment variable χ_{ij} as

$$\chi_{ij} = \begin{cases} 1 & \text{track } i \text{ from sensor 1 and track } j \\ & \text{from sensor 2 are from the same target,} \\ 0 & \text{otherwise.} \end{cases} \tag{24}$$

Denote by $\mathcal{L}_{ij}$ the likelihood ratio of the two tracks being from the same target vs. from two different targets which is the two sensor case of (22). If we assume that the track association events among different track pairs are independent, then the 2-D assignment formulation finds the most likely (joint) track-to-track association hypothesis by solving the following constrained optimization[5]

$$\min_{\chi_{ij}} \sum_{i=0}^{N_1} \sum_{j=0}^{N_2} c_{ij} \chi_{ij} \tag{25}$$

subject to

$$\sum_{i=0}^{N_1} \chi_{ij} = 1, \qquad j = 1, \ldots, N_2 \tag{26}$$

$$\sum_{j=0}^{N_2} \chi_{ij} = 1, \qquad i = 1, \ldots, N_1 \tag{27}$$

$$\chi_{ij} \in \{0,1\}, \qquad i = 0,1,\ldots,N_1, \quad j = 0,1,\ldots,N_2 \tag{28}$$

[5]Each list of tracks from a sensor is augmented by a "dummy element" with index 0, which stands for "no track," to allow for incomplete associations, while keeping the assignment problem complete.

where

$$c_{ij} = -\ln \mathcal{L}_{ij}. \tag{29}$$

This can be solved using the Auction or JVC algorithm [19]. As shown in [12] this can also be solved optimally using linear programming by relaxing the integer constraint.

The extension to multidimensional assignment (S-D) is as follows. Assume there are S sources ($S \geq 3$) where source S_i has a list of N_i tracks. Define the binary assignment variable $\chi_{i_1 i_2 \ldots i_S}$ as

$$\chi_{i_1 i_2 \ldots i_S} = \begin{cases} 1 & \text{tracks } i_1, i_2, \ldots, i_S \text{ are from the same target,} \\ 0 & \text{otherwise.} \end{cases} \tag{30}$$

We allow a subset of indices $\{i_1, i_2, \ldots, i_s\}$ to be zero in the assignment variable meaning that no track will be from the target in the corresponding list of the sources. Denote by $\mathcal{L}_{i_1 i_2 \ldots i_S}$ the likelihood ratio of the track association hypothesis vs. all tracks being from different targets which is given by (22). The S-D assignment formulation finds the most likely hypothesis by solving the following constrained optimization

$$\min_{\chi_{i_1 i_2 \ldots i_S}} \sum_{i_1=0}^{N_1} \sum_{i_2=0}^{N_2} \cdots \sum_{i_S=0}^{N_S} c_{i_1 i_2 \ldots i_S} \chi_{i_1 i_2 \ldots i_S} \tag{31}$$

subject to

$$\begin{aligned} \sum_{i_2=0}^{N_2} \cdots \sum_{i_S=0}^{N_S} \chi_{j i_2 \ldots i_S} &= 1, \qquad j = 1, \ldots, N_1 \\ \sum_{i_1=0}^{N_1} \sum_{i_3=0}^{N_3} \cdots \sum_{i_S=0}^{N_S} \chi_{i_1 j i_3 \ldots i_S} &= 1, \qquad j = 1, \ldots, N_2 \\ &\cdots \\ \sum_{i_1=0}^{N_1} \cdots \sum_{i_{S-1}=0}^{N_{S-1}} \chi_{i_1 i_2 \ldots i_{S-1} j} &= 1, \qquad j = 1, \ldots, N_S \end{aligned} \tag{32}$$

and

$$\begin{aligned} &\chi_{i_1 i_2 \ldots i_S} \in \{0,1\}, \\ &i_1 = 0,1,\ldots,N_1, \quad i_2 = 0,1,\ldots,N_2 \quad ,\ldots, \quad i_S = 0,1,\ldots,N_S. \end{aligned} \tag{33}$$

In (31) the assignment cost is

$$c_{i_1 i_2 \ldots i_S} = -\ln \mathcal{L}_{i_1 i_2 \ldots i_S} \tag{34}$$

where the likelihood ratio $\mathcal{L}_{i_1 i_2 \ldots i_S}$ (written here with a simpler index-only notation) can be computed using (22). The above constrained integer programming is, in general, NP hard. However, efficient algorithms exist to find a suboptimal solution via Lagrangian relaxation (see, e.g., [19]).

5. SIMULATION RESULTS

5.1 Evaluation of the Association Accuracy and Sensitivity

We want to study the track association accuracy for a different number of local sensors with various crosscorrelation coefficients. To make it simple, we assume that the local estimates are scalars with unity variances. The crosscorrelation coefficients between two local estimates is denoted by ρ. We choose various values of ρ, namely, 0, 0.1, 0.3, 0.5, when the local tracks correspond to the same target.

The null hypothesis H_0 is that all the local estimates correspond to the same target with its location uniformly distributed within the surveillance region of length $V = 10$

$$H_0 = \{\text{“same target”}\}. \tag{35}$$

The hypothesis H_1 is that all local estimates correspond to different targets with their locations uniformly distributed within the surveillance region

$$H_1 = \{\text{“different targets”}\}. \tag{36}$$

In this case, the separation of two targets is random and it depends on the volume of the surveillance region and no further prior knowledge is assumed.

Note that with the relatively small region V the targets, even if they are different, can be close enough to appear as they were the same, i.e., it is difficult to discriminate between the two hypotheses because they are not easily distinguishable. Consequently, even the most powerful test will not be very powerful is this situation.

The test based on (22) is used to compute the receiver operating characteristic (ROC) curves for the cases of N local tracks from the same target, i.e., the curves of the power of the test

$$P_D = P\{\text{“}H_0\text{”} \mid H_0\} \tag{37}$$

where “H_0” denotes “accept H_0,” vs. the false alarm probability

$$P_{FA} = P\{\text{“}H_0\text{”} \mid H_1\}. \tag{38}$$

Fig. 1 shows the ROC curves for the track association test with 2, 3, and 4 local track estimates and various crosscorrelation coefficients. One thousand random realizations are used for each hypothesis with fixed ρ and N to compute these curves. We can see that the test power increases as N increases for fixed V since the H_0 hypothesis becomes more distinguishable when more targets are uniformly distributed within the surveillance region. The crosscorrelation between the local track estimates is beneficial in terms of the test power under a given false alarm rate for all cases. As ρ increases, the alternative hypothesis (“different targets”) becomes more distinguishable from the null hypothesis (“same target”) because common origin tracks will then be closer to each other (in terms

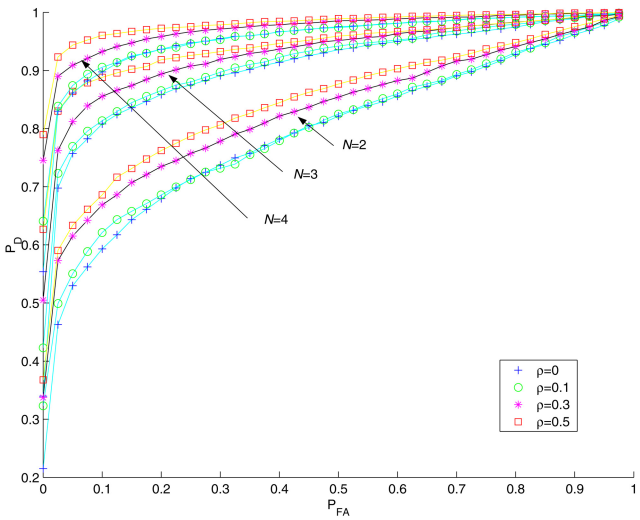


Fig. 1. ROC curves for the track association test with a different number of local estimates and various crosscorrelation coefficients.

of their normalized distance—see (9), which improves the decision accuracy. However, once H_0 is declared, the variance of the fused track estimate is larger than when they are uncorrelated [11].

Consider a case where one uses the test assuming $\rho = 0$. The threshold is determined for a certain maximum allowed miss probability of H_0, that is, $1 - P_D$. If the true crosscorrelation coefficient is, e.g., $\rho = 0.5$, the actual P_D will be higher than the one calculated under $\rho = 0$. At the same time, the actual P_{FA} will also be higher.

For two tracks (each with unity variance, for simplicity) assuming $\rho = 0$, the (chi-square) test statistic used is

$$D_0 = (\hat{x}_1 - \hat{x}_2)^2/2 \tag{39}$$

and the “design” probability of falsely rejecting the null hypothesis is

$$P\{D_0 > \tau_0 \mid H_0\} = 1 - P_D^0 \tag{40}$$

based on the chi-square distribution with one degree of freedom

$$D_0 \sim \chi_1^2. \tag{41}$$

However, since $\rho = 0.5$, (41) does not hold, Instead, under H_0,

$$D = (\hat{x}_1 - \hat{x}_2)^2/[2(1-\rho)] \sim \chi_1^2. \tag{42}$$

Thus the test statistic used, D_0, is

$$D_0 = D/2 \tag{43}$$

i.e., half of what it should have been. Consequently, the statistic D_0 will be more inclined to accept the “same target” hypothesis than the correct statistic D, i.e., P_{FA}, as well as P_D, will increase. Because the test statistic used is a scaled version of the correct one, the test assuming $\rho = 0$ uses effectively a threshold that is

double of what it would have been with $\rho = 0.5$. Thus the ROC for the test assuming $\rho = 0$ is the one with the true $\rho = 0.5$ but the operating point on it is different than the "design operating point."

This can be illustrated on Fig. 1. Assume $N = 3$ and the design operating point (on the $\rho = 0$ ROC curve) is $P_D = 0.83$, $P_{FA} = 0.025$. The actual operating point for this test is on the $\rho = 0.5$ ROC curve at $P_D = 0.86$, $P_{FA} = 0.05$. Note the sensitivity of the actual FA rate to ignoring the crosscorrelation: it is twice the design value.

5.2 A Multisensor Tracking Example

We consider a target tracking scenario where three sensors are located at $(-50, 0)$ km, $(0, 187)$ km, and $(50, 0)$ km, respectively. All three sensors measure the target range and bearing with the same standard deviations of the measurement error given by $\sigma_r = 50$ m and $\sigma_b = 2$ mrad. The sampling interval of sensors 1 and 2 is $T_1 = T_2 = 2$ s while the sampling interval of sensor 3 is $T_3 = 5$ s.

The two targets in the scenario considered are initially at $(0, 86.6)$ km and $(0.4, 86.6)$ km, respectively. Both targets move in parallel with a speed of 300 m/s. The motion of the two targets is characterized as follows. Both targets initially move south-east on a course of approximately 135°. Then at $t = 15$ s both targets make a course change with a constant turn rate of 4°/s (acceleration of about 2.1 g over a duration T_{man} of about 11 s) and head east. Both targets make a second course change at $t = 35$ s with a constant turn rate of 4°/s and head north-east. The trajectories of the two targets are shown in Fig. 2 where the true target positions are indicated at the time instances at which a measurement is made by one of the three sensors. The total time for the two targets to complete the designated trajectories is 60 s. Note that the target ranges are around 100 km at the beginning for all sensors, where the standard deviation of the crossrange measurement is around 200 m. Thus the tracker has measurement origin uncertainty when updating the target state estimates. The true target motion has a random acceleration from a white process noise with power spectral density (PSD) $q = 1$ m^2/s^3 in each realization. We assume that the two targets have unity detection probability by each sensor and there are no false measurements, i.e., each sensor has both tracks and no false tracks—in this case there are no incomplete associations to consider (see [1] for a problem with incomplete associations). The results presented are based on 100 Monte Carlo runs.

Two tracking configurations for performance comparison are implemented as follows.

i) A centralized estimator which uses an IMM with two models and sequentially updates the target state with measurements from sensors 1–3. This IMM estimator has a discretized continuous white noise acceleration (DCWNA) model (see [4], Sec. 6.2.2) with low process noise power q_l to capture the uniform target motion and a DCWNA model with high process noise PSD q_h to capture the two turns. We use $q_l = 1$ m^2/s^3 and $q_h = 8000$ m^2/s^3 which, for $T_{\text{man}} = 11$ s, corresponds to a target average acceleration of $\sqrt{q_h/T_{\text{man}}} \approx 2.6$ g. The process noise is the same in east and north of the Cartesian coordinates and uncorrelated between these coordinates. The transition between the modes is modelled according to a continuous time Markov chain with the expected sojourn times ([4], Sec. 11.7.3) in these modes given by $1/\lambda_1$ and $1/\lambda_2$, respectively. These correspond to exponential sojourn time distributions with parameters λ_1 and λ_2, respectively. The transition probability matrix between the two models (generalized version of (11.6.7-1) in [4]) from any time t_1 to time t_2 is [18]

$$\Pi(t_2, t_1) = \frac{1}{\lambda_1 + \lambda_2} \begin{bmatrix} \lambda_2 + \lambda_1 e^{-(\lambda_1+\lambda_2)T} & \lambda_1 - \lambda_1 e^{-(\lambda_1+\lambda_2)T} \\ \lambda_2 - \lambda_2 e^{-(\lambda_1+\lambda_2)T} & \lambda_1 + \lambda_2 e^{-(\lambda_1+\lambda_2)T} \end{bmatrix} \tag{44}$$

where $T = |t_2 - t_1|$. For the scenario used in simulation, we chose $\lambda_1 = \frac{1}{20}$ s^{-1} and $\lambda_2 = \frac{1}{10}$ s^{-1}. For the centralized IMM estimator, 2-D assignment is used to solve the measurement-to-track association problem and the most likely hypothesis is chosen for the filter update.

ii) In the decentralized tracking configuration each sensor uses an IMM estimator and the fusion center fuses the local estimates every $T_F = 10$ s using all local state estimates and the corresponding covariances with approximate crosscovariances. For each local IMM estimator, 2-D assignment is used to solve the measurement-to-track association problem and the most likely hypothesis is chosen for the filter update. The track-to-track association is based on the most likely hypothesis obtained by solving the 3-D assignment. If the local tracks are declared as from the same target, then the track-to-track fusion is car-

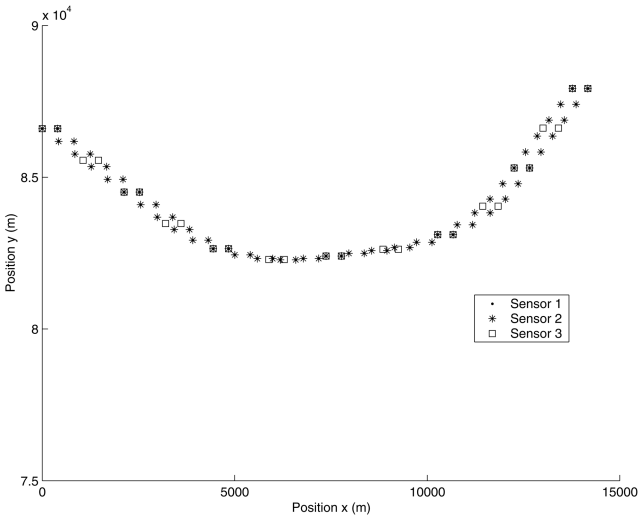


Fig. 2. Target trajectories with true positions at the times when measurements are made by the sensors.

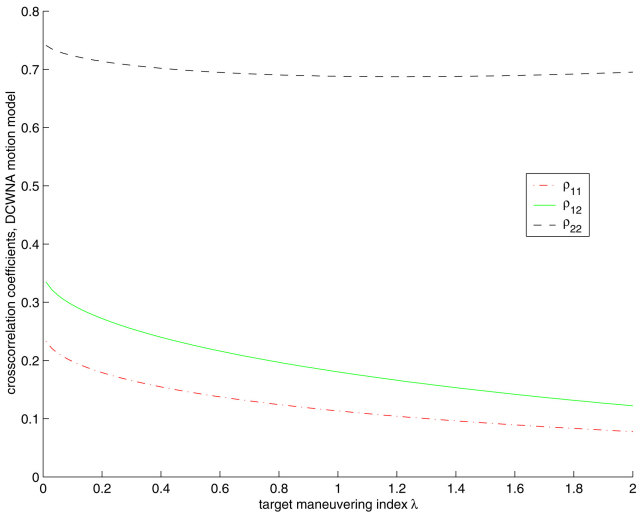


Fig. 3. Crosscorrelation coefficients vs. target maneuvering index for DCWNA model. ρ_{11}: position-position, ρ_{12}: position-velocity, ρ_{22}: velocity-velocity.

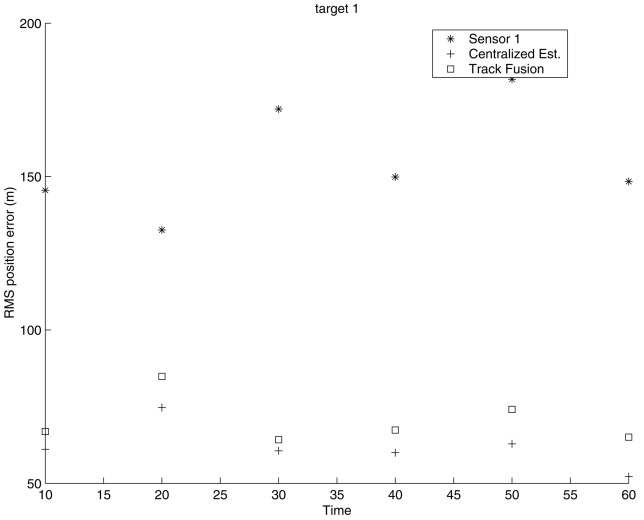


Fig. 4. Comparison of the RMS position errors for centralized IMM estimator (configuration (i)), track fusion from three IMM estimators (configuration (ii)) for target 1; local IMM estimator from sensor 1 also shown.

ried out with an approximate crosscovariance matrix, as in [11].

The combined estimates and their covariances generated by the IMM were used in the association and the corresponding state errors were approximated as Gaussian. The crosscovariance used at the fusion center is calculated using the fixed crosscorrelation coefficients detailed below. Assuming equal variances of the measurement error for both sensors, we can solve the Lyapunov equation for the steady state discretized continuous-time white noise acceleration (DCWNA) model ([2], Sec. 6.2.2). The resulting crosscorrelation coefficients between the estimation errors from the two local trackers are shown in Fig. 3 for the target maneuvering index[6] within (0.05, 2). In the simulation, we used the following fixed values for the crosscorrelation coefficients: $\rho_{11} = 0.15$ (position-position), $\rho_{12} = 0.25$ (position-velocity) and $\rho_{22} = 0.7$ (velocity-velocity) to obtain an approximate crosscovariance matrix between the local track pairs (see [11]) which was then used in the optimal track-to-track fusion algorithm.

Figs. 4 and 7 show the RMS position errors at the fusion center for the above two tracking configurations as well as that by sensor 1 alone for target 1 and target 2. Figs. 5 and 8 show the corresponding RMS velocity errors for target 1 and target 2. We can see that the track fusion of three local IMM estimators (configuration (ii)) has the RMS errors close to that of the centralized estimator (configuration (i)). Thus the proposed assignment solution to the track-to-track association is very effective when the consistency of the local tracks is good. Figs. 6 and 9 show the normalized estimation

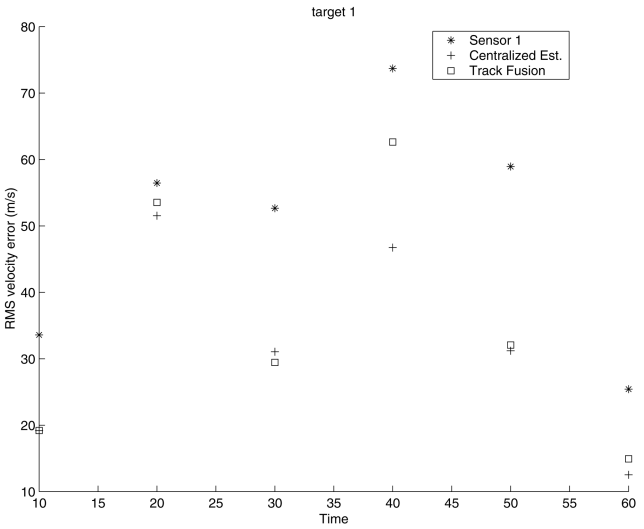


Fig. 5. Comparison of the RMS velocity errors for centralized IMM estimator (configuration (i)), track fusion from three IMM estimators (configuration (ii)) for target 1; local IMM estimator from sensor 1 also shown.

error squared (NEES, see [4], Sec. 5.4.2) at the fusion center for the above two tracking configurations as well as that by sensor 1 alone. We can see that the distributed track fusion yields larger NEES than the centralized estimator during the target turns. Thus caution has to be exercised when fusing the local estimates that are not credible on their own NEES statistics.[7]

[6]The target maneuvering index for a DCWNA model is given by $\sqrt{qT^3/\sigma_w}$ where q is the process noise PSD, T the sampling interval and σ_w the measurement noise standard deviation [4], Sec. 6.5.4.

[7]While, for maneuvering targets, the IMM estimator is superior, in terms of its NEES consistency, compared to a fixed model Kalman filter due to its adaptability, this adaptation takes about 2 sampling intervals, during which it can experience short-term inconsistency (see [4], Sec. 11.7).

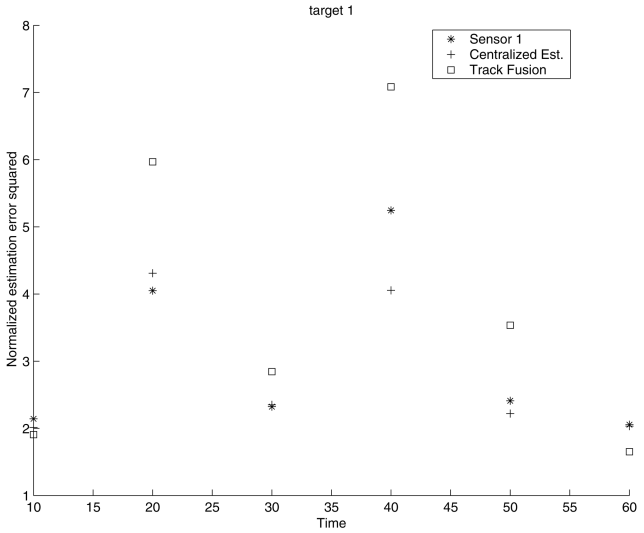


Fig. 6. Comparison of the NEES for centralized IMM estimator (configuration (i)), track fusion from three IMM estimators (configuration (ii)) for target 1; local IMM estimator from sensor 1 also shown.

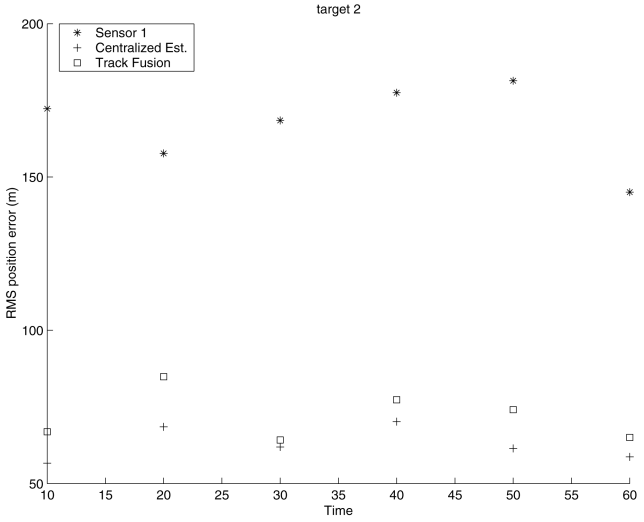


Fig. 7. Comparison of the RMS position errors for centralized IMM estimator (configuration (i)), track fusion from three IMM estimators (configuration (ii)) for target 2; local IMM estimator from sensor 1 also shown.

The ML assignment for track-to-track association from the 3 sensors over the 100 runs yielded in all runs the correct association.

6. SUMMARY AND CONCLUSIONS

In this paper the problem of track-to-track association from an arbitrary number of sources was considered where tracks of the same target obtained from different sensors have dependent estimation errors. The exact likelihood function for the track-to-track association problem from multiple sources was derived. This forms the basis for the likelihood ratio cost function used in a multidimensional assignment algorithm that can solve such a large scale data association problem. Simulation

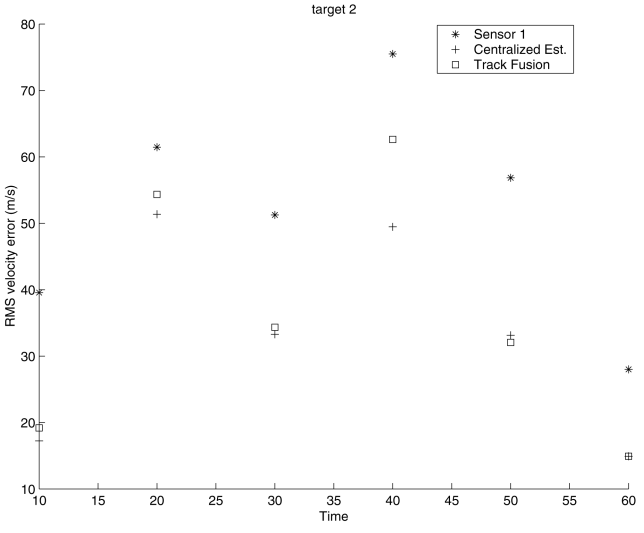


Fig. 8. Comparison of the RMS velocity errors for centralized IMM estimator (configuration (i)), track fusion from three IMM estimators (configuration (ii)) for target 2; local IMM estimator from sensor 1 also shown.

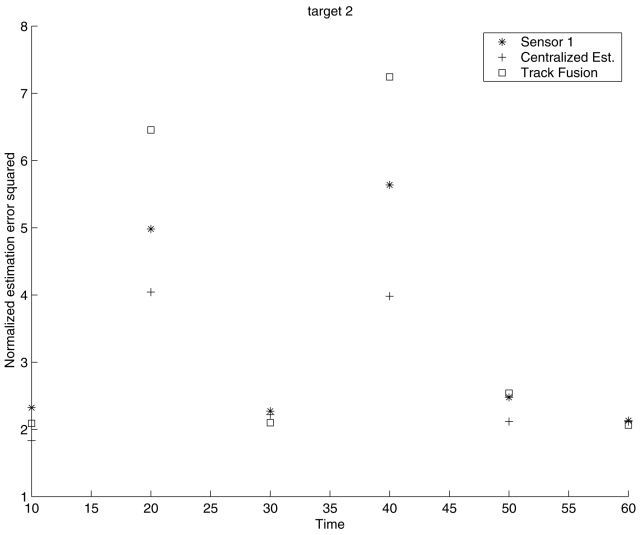


Fig. 9. Comparison of the NEES for centralized IMM estimator (configuration (i)), track fusion from three IMM estimators (configuration (ii)) for target 2; local IMM estimator from sensor 1 also shown.

results using a two-target three-sensor tracking scenario show that the estimation errors of the distributed track fusion with the assignment solution to the track association problem are only slightly larger than those of the centralized estimator. These results are in line with those of [5].

Acknowledgment

The authors are indebted to the reviewers and Editor for their constructive comments that helped improve the paper. Stimulating discussions with L. Kaplan, which led to the derivations presented in the Appendix, are gratefully acknowledged.

Appendix. The pdf of a State Estimate Conditioned on Another State Estimate

Under the common origin hypothesis $\mathcal{H}_{l_i,l_j}$ one has

$$\hat{x}_j^{l_j} = x - \tilde{x}_j^{l_j} \tag{45}$$

and

$$\hat{x}_i^{l_i} = x - \tilde{x}_i^{l_i} \tag{46}$$

where x is the common true state.

Equations (45)–(46) yield

$$\hat{x}_i^{l_i} = \hat{x}_j^{l_j} + \tilde{x}_j^{l_j} - \tilde{x}_i^{l_i}. \tag{47}$$

If the prior (unconditional) pdf of a state estimate $\hat{x}_j^{l_j}$ is diffuse (noninformative or improper [4]), it follows from (45) that the prior of the true state x is also diffuse because

1) x and $\tilde{x}_j^{l_j}$ are independent,

2) the error $\tilde{x}_j^{l_j}$ has a proper prior pdf, and

3) in order for the convolution of the pdfs of x and $\tilde{x}_j^{l_j}$ to yield a diffuse pdf for $\hat{x}_j^{l_j}$ (as assumed), the (prior) pdf of x has to be also diffuse.

Consequently, $\tilde{x}_j^{l_j}$ is independent of $\hat{x}_j^{l_j}$ since there is no inference one can make on $\tilde{x}_j^{l_j}$ from $\hat{x}_j^{l_j}$ because their relationship contains x, which has a diffuse prior pdf.

Thus

$$E[\tilde{x}_j^{l_j} \mid \hat{x}_j^{l_j}] = 0 \tag{48}$$

and, similarly

$$E[\tilde{x}_i^{l_i} \mid \hat{x}_j^{l_j}] = 0. \tag{49}$$

The conditional expectation of (47) can the be written using (48)–(49) as

$$E[\hat{x}_i^{l_i} \mid \mathcal{H}_{l_i,l_j}, \hat{x}_j^{l_j}] = E[\hat{x}_j^{l_j} + \tilde{x}_j^{l_j} - \tilde{x}_i^{l_i} \mid \mathcal{H}_{l_i,l_j}, \hat{x}_j^{l_j}] = \hat{x}_j^{l_j} \tag{50}$$

which proves (5). Equation (6) follows in a similar manner.

Finally, because all the state errors are assumed Gaussian, the conditional pdf of a state estimate in terms of another state estimate (7) follows.

REFERENCES

[1] J. Areta, Y. Bar-Shalom and M. Levedahl
A hierarchical benchmark association problem in missile defense.
In *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, #5913-38, San Diego, CA, Aug. 2005.

[2] Y. Bar-Shalom and X. R. Li
Multitarget-Multisensor Tracking: Principles and Techniques.
Storrs, CT: YBS Publishing, 1995.

[3] Y. Bar-Shalom and W. Dale Blair (Eds.)
Multitarget-Multisensor Tracking: Applications and Advances, vol. III.
Dedham, MA: Artech House, 2000.

[4] Y. Bar-Shalom, X. R. Li and T. Kirubarajan
Estimation with Applications to Tracking and Navigation: Algorithms and Software for Information Extraction.
New York: Wiley, 2001.

[5] Y. Bar-Shalom
Hierarchical tracking for the real world.
In *Proceedings of SPIE Conference on Signal Processing, Sensor Fusion and Target Recognition*, #5429-65, Orlando, FL, Apr. 2004. To appear in *IEEE Transactions on Aerospace and Electronic Systems*, **42**, 3 (July 2006).

[6] Y. Bar-Shalom and H. Chen
Multisensor track-to-track association for tracks with dependent errors.
In *Proceedings of the 43rd IEEE Conference on Decision and Control*, Nassau, Bahamas, Dec. 2004.

[7] Y. Bar-Shalom, S. S. Blackman and R. J. Fitzgerald
The dimensionless score function for measurement to track association.
In *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, #5913-50, San Diego, CA, Aug. 2005. To appear in *IEEE Transactions on Aerospace and Electronic Systems*, 2006.

[8] T. Bayes
An essay towards solving a problem in the doctrine of chances.
Phil. Trans., **52**, 1764, 370–418.

[9] S. Blackman and R. Popoli
Design and Analysis of Modern Tracking Systems.
Dedham, MA: Artech House, 1999.

[10] H. Blom
The Innovative Ideas of Bayes.
Private Communication, 2005.

[11] H. Chen, T. Kirubarajan and Y. Bar-Shalom
Performance limits of track-to-track fusion vs. centralized estimation: theory and application.
IEEE Transactions on Aerospace and Electronic Systems, **39**, 2 (Apr. 2003), 386–400.

[12] H. Chen, K. R. Pattipati, T. Kirubarajan and Y. Bar-Shalom
Data association with possibly unresolved measurements using linear programming.
In *Proceedings of the 5th ONR/GTRI Workshop on Target Tracking*, Newport, RI, June 2002.

[13] C-Y. Chong, S. Mori and K. C. Chang
Distributed multitarget multisensor tracking.
Y. Bar-Shalom (Ed.), *Multitarget-Multisensor Tracking: Advanced Applications*, vol. I, Dedham, MA: Artech House, 1990.

[14] L. M. Kaplan and W. D. Blair
Assignment costs for multiple sensor track-to-track association.
In *Proceedings of International Conference on Information Fusion*, Stockholm, Sweden, July 2004.

[15] L. M. Kaplan, W. D. Blair and Y. Bar-Shalom
Simulation studies of multisensor track association and fusion methods.
In *Proceedings of 2006 IEEE/AIAA Aerospace Conference*, Big Sky, MT, Mar. 2006.

[16] B. F. La Scala and A. Farina
Choosing a track association method.
Information Fusion, **3**, 2 (2002), 119–133.

[17] X. D. Lin, Y. Bar-Shalom and T. Kirubarajan
Multisensor bias estimation using local tracks without a priori association.
In *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, #5204-70, San Diego, CA, Aug. 2003.

[18] A. Papoulis and S. U. Pillai
Probability, Random Variables and Stochastic Processes.
New York: McGraw Hill, 2002.

[19] K. R. Pattipati, et al.
Survey of assignment techniques for multitarget tracking.
Y. Bar-Shalom and W. Dale Blair (Eds.), *Multitarget-Multisensor Tracking: Applications and Advances*, vol. III, Dedham, MA: Artech House, 2000, Ch. 2.

Yaakov Bar-Shalom was born on May 11, 1941. He received the B.S. and M.S. degrees from the Technion, Israel Institute of Technology, in 1963 and 1967 and the Ph.D. degree from Princeton University in 1970, all in electrical engineering.

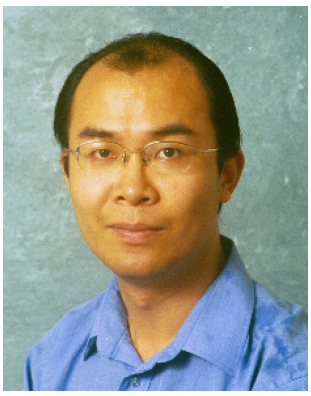
From 1970 to 1976 he was with Systems Control, Inc., Palo Alto, California. Currently he is Board of Trustees Distinguished Professor in the Department of Electrical and Computer Engineering and Marianne E. Klewin Professor in Engineering at the University of Connecticut. He is also director of the ESP (Estimation and Signal Processing) Lab. During 1976 and 1977 he served as associate editor of the *IEEE Transactions on Automatic Control* and from 1978 to 1981 as associate editor of *Automatica*. He was program chairman of the 1982 American Control Conference, general chairman of the 1985 ACC, and cochairman of the 1989 IEEE International Conference on Control and Applications.

During 1983–87 he served as chairman of the Conference Activities Board of the IEEE Control Systems Society and during 1987–89 was a member of the Board of Governors of the IEEE CSS. He was a member of the Board of Directors of the International Society of Information Fusion (1999–2004) and served as general chairman of FUSION 2000, president of ISIF in 2000 and 2002 and vice president for *Publications* in 2004–06.

In 1987 he received the IEEE CSS Distinguished Member Award. Since 1995 he is a distinguished lecturer of the IEEE AESS and has given numerous keynote addresses at major national and international conferences. He is corecipient of the M. Barry Carlton Award for the best paper in the *IEEE Transactions on Aerospace and Electronic Systems* in 1995 and 2000 and the 1998 University of Connecticut AAUP Excellence Award for Research. In 2002 he received the J. Mignona Data Fusion Award from the DoD JDL Data Fusion Group. He is a member of the Connecticut Academy of Science and Engineering.

His current research interests are in estimation theory and target tracking and has published over 330 papers and book chapters in these areas and in stochastic adaptive control. He coauthored the monograph *Tracking and Data Association* (Academic Press, 1988), the graduate texts *Estimation and Tracking: Principles, Techniques and Software* (Artech House, 1993), *Estimation with Applications to Tracking and Navigation: Algorithms and software for Information Extraction* (Wiley, 2001), the advanced graduate text *Multitarget-Multisensor Tracking: Principles and Techniques* (YBS Publishing, 1995), and edited the books *Multitarget-Multisensor Tracking: Applications and Advances* (Artech House, Vol. I, 1990; Vol. II, 1992; Vol. III, 2000).

He has been elected Fellow of IEEE for "contributions to the theory of stochastic systems and of multi-target tracking." He has been consulting to numerous companies and government agencies, and originated the series titled Multitarget-Multisensor Tracking, short courses offered via UCLA Extension, at Government Laboratories, private companies and overseas.



Huimin Chen received the B.E. and M.E. degrees from Department of Automation, Tsinghua University, Beijing, China, in 1996 and 1998, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, in 2002, all in electrical engineering.

He was a post doctoral research associate at the Physics and Astronomy Department, University of California, Los Angeles, and a visiting researcher with the Department of Electrical and Computer Engineering, Carnegie Mellon University from July 2002 where his research focus was on weak signal detection for single electron spin microscopy. He joined the Department of Electrical Engineering, University of New Orleans in January 2003 as an assistant professor. His research interests are in general areas of signal processing, estimation theory, and information theory with applications to target detection and target tracking.

Joint Particle Filtering of Multiple Maneuvering Targets From Unassociated Measurements

HENK A. P. BLOM
EDWIN A. BLOEM
National Aerospace Laboratory NLR

The problem of maintaining tracks of multiple maneuvering targets from unassociated measurements is formulated as a problem of estimating the hybrid state of a Markov jump linear system from measurements made by a descriptor system with independent, identically distributed (i.i.d.) stochastic coefficients. This characterization is exploited to derive the exact equation for the Bayesian recursive filter, to develop two novel Sampling Importance Resampling (SIR) type particle filters, and to derive approximate Bayesian filters which use for each target one Gaussian per maneuver mode. The two approximate Bayesian filters are a compact and a track-coalescence avoiding version of Interacting Multiple Model Joint Probabilistic Data Association (IMMJPDA). The relation of each of the four novel filter algorithms to the literature is well explained. Through Monte Carlo simulations for a two target example, these four filters are compared to each other and to the approach of using one IMMPDA filter per target track. The Monte Carlo simulation results show that each of the four novel filters clearly outperforms the IMMPDA approach. The results also show under which conditions the IMMJPDA type filters perform close to exact Bayesian filtering, and under which conditions not.

Manuscript received July 7, 2004; revised December 31, 2005 and May 2, 2006.

Refereeing of this contribution was handled by Shozo Mori.

Authors' addesss: National Aerospace Laboratory NLR Amsterdam, The Netherlands, E-mail: (blom@nlr.nl, bloem@nlr.nl).

1. INTRODUCTION

In the literature approximate Bayesian approaches towards maintaining tracks of multiple maneuvering targets from unassociated measurements have focussed on the development of combinations of Interacting Multiple Model (IMM) and Joint Probabilistic Data Association (JPDA) approaches. Initially, combinations of IMM and JPDA have been developed along two heuristic directions. Bar-Shalom et al. [4] heuristically developed an IMMJPDA-Coupled filter for situations where the measurements of two targets are unresolved during periods of close encounter. The filters of the individual targets are coupled through cross-target-covariance terms. The filtering results obtained have not been very encouraging to continue this heuristic approach. De Feo et al. [20] combined JPDA and a rather crude approximation of IMM, under the name IMMJPDA. The first proper combination of IMM and JPDA was developed by Chen and Tugnait [18]. Focus of this development was on showing that fixed-lag IMMJPDA smoothing performed far better than IMMJPDA filtering at the cost of 3 scans delay. In [9], [10] we used the descriptor system approach [8] to develop a track-coalescence-avoiding version of IMMJPDA (for short IMMJPDA*). Moreover, we showed that both IMMJPDA and IMMJPDA* perform much better than just applying IMMPDA filtering per maintained track. In spite of these developments it remains unclear how IMMJPDA and IMMJPDA* filtering performs in comparison with the exact Bayesian filter.

This motivates us to study the Sampling Importance Resampling (SIR) based Particle Filter (PF) paradigm [21, 28, 43] for maintaining tracks of multiple maneuvering targets from unassociated measurements. During the last decade this paradigm has been recognized as a practical means for approximating an exact Bayesian filter arbitrarily well. This has stimulated the development of a large variety of particle filters (e.g. [1, 22, 38, 42]) that typically outperform established approximate non-linear filtering and target track maintenance approaches such as Extended Kalman Filtering, Probabilistic Data Association (PDA), the Interacting Multiple Model (IMM) algorithm, and their combinations.

The extension of these results to multiple target tracking situations has also received significant attention. Early on it was recognized that the JPDA formalism provided a logical starting point for this development. Gordon [26] developed a SIR-PF version by replacing JPDA's Gaussian density by a density the evolution of which is approximated with help of a SIR particle filter. Avitzour [2] developed a more advanced SIR particle filter by using joint-target particles; we refer to this as SIR joint PF. Karlsson and Gustafsson [30] compared the RMS position errors of a SIR joint PF with those of a JPDA filter for maintaining tracks in an example of two perpendicular crossing targets. For this "easy" example the difference in performance

appeared to be small. Salmond et al. [45] showed that a SIR joint PF works well for the initialization of two non-maneuvering targets that start from the same initial position. Gordon et al. [27] developed a SIR joint PF approach for tracking a group of targets, the members of which stay close to each other. Through several complementary studies, efficiency improvements have been developed for these particle filters, e.g. [29, 41, 42, 48]. To track multiple objects for robotic vision, Schultz et al. [46] developed an occlusion extension for SIR PF and showed that this outperformed JPDA on a multi-person tracking problem. Tracking multiple objects with occlusion situations by SIR joint PF for robotic vision has been shown in [33] and [34].

A complementary development in SIR particle filtering is to use sensor measurements at the pixel level as observations. This allows handling the problems of target detection and target tracking in an integrated way, and thus to shortcut the traditional sequence of signal processing first, followed by target detection (thresholding) and then target tracking. The feasibility of a track-before-detect particle filtering approach has been introduced in [15, 44] for a single target. Extensions to multiple targets have been developed in [40] using single target particles, and in [17, 32, 35] using joint particles. For the current paper we assume that track maintenance has to be performed on the basis of detected measurement observations, and that pixel level sensor measurements are not available. Hence, the track-before-detect problem setting goes beyond the scope of the current study.

The aim of this paper is to extend the SIR joint particle filter approach towards track maintenance, to the situation of multiple maneuvering targets and to evaluate for an example how the performance of these particle filters compares with IMMJPDA and IMMJPDA* filtering. This asks for the combination of an SIR joint PF for unassociated measurements with an SIR PF for tracking a suddenly maneuvering target [16, 36, 37]. The basis for this integration is provided by the exact Bayesian filter for this particular problem. We developed such an exact Bayesian characterisation using the descriptor system approach [10, 14]. The current paper extends these results in the sense of incorporating a non-homogeneous false measurement density [39].

The specialty of this exact characterization is that both the mode switching and the data association are performed jointly for all targets and that the false plot density is non-homogeneous. Based on such exact equations, we develop a standard SIR particle filter to evaluate the exact Bayesian equations. A weakness of this standard SIR joint particle filter is that after a resampling step for some of the joint modes there may be hardly any or even no particles left. In theory this can be compensated for by significantly increasing the number of particles. However, a more effective approach is to resample a fixed number of joint particles per joint mode. We refer to this as hybrid SIR joint particle filtering. Through Monte Carlo simulations for a simple example the standard SIR and hybrid SIR joint particle filters are compared with the following three combinations of IMM and PDA:

- An IMMPDA filter, which updates an individual IMM track using MMPDA [25] and implicitly assuming there are no other targets;
- A compact version of IMMJPDA, which we derive in this paper in a systematic way from the exact Bayesian filter equations; and
- The track coalescence avoiding version (IMMJPDA*) of this compact IMMJPDA.

The paper is organized as follows. Section 2 formulates the multi-target track maintenance problem considered. Section 3 embeds this in filtering for a jump linear descriptor system. Section 4 develops an exact Bayesian characterization of the evolution of the conditional density for the state of the multiple targets. Section 5 develops the standard SIR joint particle filter. Section 6 develops the hybrid SIR joint particle filter. Section 7 adopts the IMMJPDA assumptions, and shows the impact on the filter equations relative to those of [18]. Section 8 develops IMMJPDA*. Section 9 illustrates and compares the performance of these filters through Monte Carlo simulation results. As a performance reference we also run single target IMMPDA filters on the same scenario. Finally, Section 10 draws conclusions.

2. MULTITARGET TRACK MAINTENANCE PROBLEM

Consider M targets and assume that the state of the ith target is modelled as a jump linear system:

$$x_t^i = a^i(\theta_t^i)x_{t-1}^i + b^i(\theta_t^i)w_t^i, \qquad i = 1,\ldots,M \tag{1}$$

where x_t^i is the n-vectorial state of the ith target, θ_t^i is the Markovian switching mode of the ith target and assumes values from $\mathbb{M} = \{1,\ldots,N\}$ according to a transition probability matrix Π^i, $a^i(\theta_t^i)$ and $b^i(\theta_t^i)$ are $(n \times n)$- and $(n \times n')$-matrices and w_t^i is a sequence of i.i.d. standard Gaussian variables of dimension n' with w_t^i, w_t^j independent for all $i \neq j$ and w_t^i, (x_0^i, θ_0^i), (x_0^j, θ_0^j) independent for all $i \neq j$. At $t = 0$, the joint density $p_{x_0^i,\theta_0^i}$ is known for each $i \in [1,M]$; typically these are i-variant.

A set of measurements consists of measurements originating from targets and measurements originating from clutter. We assume that a potential measurement originating from target i is also modelled as a jump linear system:

$$z_t^i = h^i(\theta_t^i)x_t^i + g^i(\theta_t^i)v_t^i, \qquad i = 1,\ldots,M \tag{2}$$

where z_t^i is an m-vector, $h^i(\theta_t^i)$ is an $(m \times n)$-matrix and $g^i(\theta_t^i)$ is an $(m \times m')$-matrix, and v_t^i is a sequence of i.i.d. standard Gaussian variables of dimension m' with v_t^i and

v_t^j independent for all $i \neq j$. Moreover v_t^i is independent of x_0^j and w_t^j for all i,j.

Let $x_t \triangleq \mathrm{Col}\{x_t^1,\ldots,x_t^M\}$, $\theta_t \triangleq \mathrm{Col}\{\theta_t^1,\ldots,\theta_t^M\}$, $A(\theta_t) \triangleq \mathrm{Diag}\{a^1(\theta_t^1),\ldots,a^M(\theta_t^M)\}$, $B(\theta_t) \triangleq \mathrm{Diag}\{b^1(\theta_t^1),\ldots,b^M(\theta_t^M)\}$, and $w_t \triangleq \mathrm{Col}\{w_t^1,\ldots,w_t^M\}$. Then we can model the state of our M targets as follows:

$$x_t = A(\theta_t)x_{t-1} + B(\theta_t)w_t \tag{3}$$

with A and B of size $Mn \times Mn$ and $Mn \times Mn'$ respectively, with $\{\theta_t\}$ assuming values from $\mathbb{M}^M$ according to the transition probability matrix $\Pi = [\Pi_{\eta,\theta}]$. If the M targets switch mode independently of each other, then:

$$\Pi_{\eta,\theta} = \prod_{i=1}^{M} \Pi^i_{\eta^i,\theta^i} \tag{4}$$

for every $\eta \in \mathbb{M}^M$ and $\theta \in \mathbb{M}^M$.

Next with $z_t \triangleq \mathrm{Col}\{z_t^1,\ldots,z_t^M\}$, $H(\theta_t) \triangleq \mathrm{Diag}\{h^1(\theta_t^1),\ldots,h^M(\theta_t^M)\}$, $G(\theta_t) \triangleq \mathrm{Diag}\{g^1(\theta_t^1),\ldots,g^M(\theta_t^M)\}$, and $v_t \triangleq \mathrm{Col}\{v_t^1,\ldots,v_t^M\}$, we obtain:

$$z_t = H(\theta_t)x_t + G(\theta_t)v_t \tag{5}$$

with H and G of size $Mm \times Mn$ and $Mm \times Mm'$ respectively.

We next assume that with a non-zero detection probability, P_{d}^i, target i is indeed observed at moment t. In addition to this there may be false measurements, the density of which is not homogeneous. Similar to [39] we assume that the number of false measurements at moment t, F_t, has a Poisson distribution:

$$\begin{aligned} p_{F_t}(F) &= \frac{(\hat{F}_t)^F}{F!}\exp(-\hat{F}_t), \qquad F = 0,1,2,\ldots \\ &= 0, \qquad\qquad\qquad\quad \text{else} \end{aligned} \tag{6a}$$

where $\hat{F}_t$ is the expected number of false measurements. Let f_t denote the column vector of i.i.d. false measurements, then the conditional density of f_t given F_t satisfies:

$$p_{f_t|F_t}(f \mid F) = \prod_{i=1}^{F} p_f(f^i) \tag{6b}$$

where $p_f(\cdot)$ is the (measurable) probability density function of a false measurement. Hence, the local density $\lambda(\cdot)$ of false measurements satisfies:

$$\lambda(f^i) = \hat{F}_t p_f(f^i). \tag{6c}$$

Furthermore we assume that the process $\{F_t, f_t\}$ is a sequence of independent vectors, which are independent of $\{x_t\}$, $\{w_t\}$ and $\{v_t\}$.

At moment $t = 1,2,\ldots,T$ a vector observation y_t is made, the components of which consist of the potential observations z_t^i of the detected targets plus the false measurements $\{F_t, f_t\}$. The multi-target track maintenance problem considered is to estimate x_t, θ_t given observations $Y_t \triangleq \{\mathrm{y}_s; 0 \leq s \leq t\}$ with y_0 representing the initial joint density p_{x_0,θ_0}.

3. STOCHASTIC MODELLING OF OBSERVATION EQUATION

This section characterizes the exact relationship between observation vector y_t and the false and potential observations at moment $t > 0$. For this we largely follow [8]. The measurement vector y_t consists of measurements originating from targets and measurements originating from clutter. Firstly, the relation for measurements originating from targets is identified. Subsequently, the clutter measurements are randomly inserted between the target measurements.

Let $\phi_{i,t} \in \{0,1\}$ be the detection indicator for target i, which assumes the value one with a time invariant probability $P_{\mathrm{d}}^i > 0$, independently of $\phi_{j,t}$, $j \neq i$ and independently of the processes introduced in Section 2. This approach yields the following detection indicator vector ϕ_t of size M:

$$\phi_t \triangleq \mathrm{Col}\{\phi_{1,t},\ldots,\phi_{M,t}\}.$$

Thus, the number of detected targets is $D_t \triangleq \sum_{i=1}^{M} \phi_{i,t}$. Furthermore, we assume that $\{\phi_t\}$ is a sequence of i.i.d. vectors.

In order to link the detection indicator vector with the measurement model, we introduce the following operator Φ: for an arbitrary vector ϕ' of length M' and having $(0,1)$ valued components, we define $D(\phi') \triangleq \sum_{i=1}^{M'} \phi'_i$ and the operator Φ producing $\Phi(\phi')$ as a $(0,1)$-valued matrix of size $D(\phi') \times M'$ of which the ith row equals the ith non-zero row of $\mathrm{Diag}\{\phi'\}$. Next we define, for $D_t > 0$, a vector that contains all measurements originating from targets in a fixed order

$$\tilde{z}_t \triangleq \mathbf{\Phi}(\phi_t)z_t \qquad \text{where} \quad \mathbf{\Phi}(\phi_t) \triangleq \Phi(\phi_t) \otimes I_m$$

with I_m a unit-matrix of size m, and $\otimes$ the Kronecker product, i.e.,

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \otimes I_m \triangleq \begin{bmatrix} aI_m & \vdots & bI_m \\ & \cdots\cdots & \\ cI_m & \vdots & dI_m \end{bmatrix}.$$

In reality, however, we do not know in which order the targets are observed. Hence, we introduce the stochastic $D_t \times D_t$ permutation matrix χ_t, which is independent of the processes introduced in Section 2 and is conditionally independent of $\{\phi_t\}$ given D_t. We also assume that $\{\chi_t\}$ is a sequence of independent matrices. Hence, for $D_t > 0$,

$$\tilde{\tilde{z}}_t \triangleq \boldsymbol{\chi}_t \tilde{z}_t \qquad \text{where} \quad \boldsymbol{\chi}_t \triangleq \chi_t \otimes I_m$$

is a vector that contains all measurements originating from targets at moment t in a random order.

Let the random variable L_t be the total number of measurements at moment t. Thus,

$$L_t = D_t + F_t.$$

We next describe the relationship between the potential measurement vector z_t, the false plot vector f_t and the measurement vector $\mathrm{y}_t \triangleq \mathrm{Col}\{\mathrm{y}_{1,t},\ldots,\mathrm{y}_{L_t,t}\}$, where $\mathrm{y}_{i,t}$ denotes the ith m-vectorial measurement at moment t. Because y_t contains a random mixture of D_t target measurements and $L_t - D_t$ false measurements, the relation between z_t and y_t can be characterized by the following pair of equations for the target and false measurements respectively:

$$\begin{aligned} \boldsymbol{\Phi}(\psi_t)\mathrm{y}_t = \boldsymbol{\chi}_t\boldsymbol{\Phi}(\phi_t)z_t \quad & \text{if} \quad D_t > 0 \\ = \{\} \quad & \text{if} \quad D_t = 0 \end{aligned} \tag{7a}$$

$$\begin{aligned} \boldsymbol{\Phi}(\psi_t^*)\mathrm{y}_t = f_t \quad & \text{if} \quad L_t > D_t \\ = \{\} \quad & \text{if} \quad L_t = D_t \end{aligned} \tag{7b}$$

where ψ_t, ψ_t^*, χ_t are explained below.

First we explain the target measurement (7a). This equation has stochastic i.i.d. coefficients $\boldsymbol{\Phi}(\psi_t)$ and $\boldsymbol{\chi}_t\boldsymbol{\Phi}(\phi_t)$. The detected target measurements in the observation vector y_t are in random order. Hence, the potential detected measurements of targets need to be randomly mixed. To perform this by a simple matrix multiplication, a sequence of independent stochastic permutation matrices $\{\chi_t\}$ of size $D_t \times D_t$ is defined and assumed to be independent of $\{\phi_t\}$. To take into account the measurement vector size m, χ_t needs to be "inflated" to the proper size of $D_t m$ by means of the Kronecker product with I_m. To this end, $\boldsymbol{\chi}_t \triangleq \chi_t \otimes I_m$ with I_m a unit-matrix of size m, and $\otimes$ the Kronecker product. Hence $\boldsymbol{\chi}_t\boldsymbol{\Phi}(\phi_t)z_t$ is a column vector of potential detected measurements of targets in random order.

$\psi_t \triangleq \mathrm{Col}\{\psi_{1,t},\ldots,\psi_{L_t,t}\}$ is the target indicator vector, where $\psi_{i,t} \in \{0,1\}$ is a target indicator at moment t for measurement i, which assumes the value one if measurement i belongs to a detected target and zero if measurement i is false. Because there are as many detected targets as target measurements, the following constraint applies:

$$D(\psi_t) = D(\phi_t). \tag{8}$$

Under this equality constraint, $\{\psi_t\}$ is a sequence of independent vectors that is D_t-conditionally independent of all earlier defined processes.

In order to let ψ_t select the correct measurements by simple matrix multiplication, the matrix operator Φ defined above is used. To take into account the measurement vector size m, $\Phi(\psi_t)$ needs to be "inflated" to the proper size of $D_t m$ by means of the Kronecker product with I_m. To this end, $\boldsymbol{\Phi}(\psi') \triangleq \Phi(\psi') \otimes I_m$ with I_m a unit-matrix of size m, and $\otimes$ the Kronecker product. Hence $\boldsymbol{\Phi}(\psi_t)\mathrm{y}_t$ is a column vector that contains all detected target measurements in y_t.

$\psi_t^* \triangleq \mathrm{Col}\{\psi_{1,t}^*,\ldots,\psi_{L_t,t}^*\}$ is a false indicator vector of size L_t with $\psi_{i,t}^* = 1 - \psi_{i,t}$. To select the false measurements by matrix multiplication, the matrix operator Φ is used again. Hence $\boldsymbol{\Phi}(\psi_t^*)\mathrm{y}_t$ is a column vector that contains all false measurements from y_t.

Finally we develop a characterization for y_t. For this we first verify the following for $L_t > D_t > 0$:

$$\Phi(\psi_t)^T\Phi(\psi_t) + \Phi(\psi_t^*)^T\Phi(\psi_t^*) = I_{L_t \times L_t}.$$

Hence

$$\mathrm{y}_t = [\Phi(\psi_t)^T\Phi(\psi_t) + \Phi(\psi_t^*)^T\Phi(\psi_t^*)]\mathrm{y}_t \quad \text{if} \quad L_t > D_t > 0.$$

Substituting (7a) and (7b) into this equation yields the following model for the observation vector y_t:

$$\begin{aligned} \mathrm{y}_t &= \boldsymbol{\Phi}(\psi_t)^T\boldsymbol{\chi}_t\boldsymbol{\Phi}(\phi_t)z_t + \boldsymbol{\Phi}(\psi_t^*)^T f_t && \text{if} \quad L_t > D_t > 0 \\ &= \boldsymbol{\Phi}(\psi_t)^T\boldsymbol{\chi}_t\boldsymbol{\Phi}(\phi_t)z_t && \text{if} \quad L_t = D_t > 0 \\ &= \boldsymbol{\Phi}(\psi_t^*)^T f_t && \text{if} \quad L_t > D_t = 0 \\ &= \{\} && \text{if} \quad L_t = 0. \end{aligned} \tag{9}$$

Together with equations (3), (4), (5) and (6), equation (9) forms a complete characterization of our tracking problem in terms of stochastic difference equations.

EXAMPLE Assume we maintain tracks of five targets ($M = 5$), of which we detect detect and observe four ($D_t = 4$) together with two false measurements ($F_t = 2$), and with:

$$\chi_t = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \qquad \begin{aligned} \phi_t &= [1\ 0\ 1\ 1\ 1]^T \\ \psi_t &= [1\ 1\ 0\ 1\ 1\ 0]^T \end{aligned}$$

i.e., the 2nd target is not detected, and the 3rd and 6th measurements are false. This implies:

$$\Phi(\phi_t) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\Phi(\psi_t) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\Phi(\psi_t^*) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\chi_t \Phi(\phi_t) = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{\Phi}(\psi_t)^T \boldsymbol{\chi}_t \boldsymbol{\Phi}(\phi_t) z_t = [z_{5,t} \;\; z_{4,t} \;\; 0 \;\; z_{1,t} \;\; z_{3,t} \;\; 0]^T$$

$$\boldsymbol{\Phi}(\psi_t^*)^T f_t = [0 \;\; 0 \;\; f_{1,t} \;\; 0 \;\; 0 \;\; f_{2,t}]^T.$$

Substituting this in (9) yields:

$$\mathrm{y}_t = [z_{5,t} \;\; z_{4,t} \;\; f_{1,t} \;\; z_{1,t} \;\; z_{3,t} \;\; f_{2,t}]^T.$$

4. EXACT FILTER EQUATIONS

In this section a Bayesian characterization of the conditional density $p_{x_t,\theta_t|Y_t}(x,\theta)$ is given where Y_t denotes the σ-algebra generated by measurements y_t up to and including moment t. Subsequently, characterizations are developed for the mode probabilities and the mode conditional means and covariances.

First we introduce an auxiliary indicator matrix process $\tilde{\chi}_t$ of size $D_t \times L_t$, as follows:

$$\tilde{\chi}_t \triangleq \chi_t^T \Phi(\psi_t) \qquad \text{if} \quad D_t > 0. \tag{10}$$

Pre-multiplying the left- and right hand terms in (9) with $\tilde{\boldsymbol{\chi}}_t = \tilde{\chi}_t \otimes I_m$ and subsequent straightforward evaluation yields:

$$\tilde{\boldsymbol{\chi}}_t \mathrm{y}_t = \boldsymbol{\Phi}(\phi_t) H(\theta_t) x_t + \boldsymbol{\Phi}(\phi_t) G(\theta_t) v_t \qquad \text{if} \quad D_t > 0 \tag{11}$$

where the size of $\tilde{\boldsymbol{\chi}}_t$ is $D_t m \times L_t m$ and the size of $\boldsymbol{\Phi}(\phi_t)$ is $D_t m \times M m$.

Notice that (11) is a linear Gaussian descriptor system [19] with stochastic i.i.d. coefficients $\tilde{\boldsymbol{\chi}}_t$ and $\boldsymbol{\Phi}(\phi_t)$ and Markovian switching coefficients $H(\theta_t)$ and $G(\theta_t)$.

From (11), it follows that for $D_t > 0$ all relevant associations and permutations can be covered by $(\phi_t, \tilde{\chi}_t)$-hypotheses. We extend this to $D_t = 0$ by adding the combination $\phi_t = \{0\}^M$ and $\tilde{\chi}_t = \{\}^{L_t}$. Hence, through defining the weights

$$\beta_t(\phi, \tilde{\chi}, \theta) \triangleq \mathrm{Prob}\{\phi_t = \phi, \tilde{\chi}_t = \tilde{\chi}, \theta_t = \theta \mid Y_t\}$$

the law of total probability yields:

$$p_{x_t,\theta_t|Y_t}(x,\theta) = \sum_{\tilde{\chi},\phi} \beta_t(\phi, \tilde{\chi}, \theta) p_{x_t|\theta_t,\phi_t,\tilde{\chi}_t,Y_t}(x \mid \theta, \phi, \tilde{\chi}). \tag{12}$$

And thus, our problem is to characterize the terms in the last summation. This problem is solved in two steps, the first of which is the following Proposition.

PROPOSITION 1 *For any $\phi \in \{0,1\}^M$, such that $D(\phi) \triangleq \sum_{i=1}^M \phi_i \le L_t$, and any $\tilde{\chi}_t$ matrix realization $\tilde{\chi}$ of size $D(\phi) \times L_t$, the following holds true:*

$$p_{x_t|\theta_t,\phi_t,\tilde{\chi}_t,Y_t}(x \mid \theta, \phi, \tilde{\chi}) = \frac{p_{\tilde{z}_t|x_t,\theta_t,\phi_t}(\tilde{\boldsymbol{\chi}} \mathrm{y}_t \mid x, \theta, \phi) \cdot p_{x_t|\theta_t,Y_{t-1}}(x \mid \theta)}{F_t(\phi, \tilde{\chi}, \theta)} \tag{13}$$

$$\beta_t(\phi, \tilde{\chi}, \theta) = F_t(\phi, \tilde{\chi}, \theta) \cdot \prod_{j=1}^{L_t - D(\phi)} \lambda([\Phi(1_{L_t} - \tilde{\chi}^T \tilde{\chi} 1_{L_t}) \mathrm{y}_t]_j) \cdot \prod_{i=1}^{M} [(1 - P_d^i)^{(1-\phi_i)} (P_d^i)^{\phi_i}] \cdot p_{\theta_t|Y_{t-1}}(\theta)/c_t \tag{14}$$

where $\tilde{\boldsymbol{\chi}} \triangleq \tilde{\chi} \otimes I_m$, $1_{L_t} = [1, \ldots, 1]^T$ is an L_t vector with 1-valued elements and $F_t(\phi, \tilde{\chi}, \theta)$ and c_t are such that they normalize $p_{x_t|\theta_t,\phi_t,\tilde{\chi}_t,Y_t}(x \mid \theta, \phi, \tilde{\chi})$ and $\beta_t(\phi, \tilde{\chi}, \theta)$ respectively.

PROOF See Appendix A.

The next step starts with substituting (13) and (14) into (12), which yields:

$$p_{x_t,\theta_t|Y_t}(x,\theta) = \sum_{\tilde{\chi},\phi} \left[\frac{p_{\tilde{z}_t|x_t,\theta_t,\phi_t}(\tilde{\boldsymbol{\chi}} \mathrm{y}_t \mid x, \theta, \phi) \cdot p_{x_t|\theta_t,Y_{t-1}}(x \mid \theta)}{F_t(\phi, \tilde{\chi}, \theta)} \cdot F_t(\phi, \tilde{\chi}, \theta) \cdot \prod_{j=1}^{L_t - D(\phi)} \lambda([\Phi(1_{L_t} - \tilde{\chi}^T \tilde{\chi} 1_{L_t}) \mathrm{y}_t]_j) \cdot \prod_{i=1}^{M} [(1 - P_{\mathrm{d}}^i)^{(1-\phi_i)} (P_{\mathrm{d}}^i)^{\phi_i}] \right] \cdot p_{\theta_t|Y_{t-1}}(\theta)/c_t.$$

Simplifying this and rearranging terms yields:

$$p_{x_t,\theta_t|Y_t}(x,\theta) = \sum_{\tilde{\chi},\phi} \left[p_{\tilde{z}_t|x_t,\theta_t,\phi_t}(\tilde{\boldsymbol{\chi}} \mathrm{y}_t \mid x, \theta, \phi) \cdot p_{x_t,\theta_t|Y_{t-1}}(x,\theta) \cdot \prod_{j=1}^{L_t - D(\phi)} \lambda([\Phi(1_{L_t} - \tilde{\chi}^T \tilde{\chi} 1_{L_t}) \mathrm{y}_t]_j) \cdot \prod_{i=1}^{M} [(1 - P_{\mathrm{d}}^i)^{(1-\phi_i)} (P_{\mathrm{d}}^i)^{\phi_i}]/c_t \right] \tag{15}$$

with

$$p_{\tilde{z}_t|x_t,\theta_t,\phi_t}(\tilde{z} \mid x, \theta, \phi) = N\{\tilde{z}; \boldsymbol{\Phi}(\phi) H(\theta) x, \boldsymbol{\Phi}(\phi) G(\theta) G(\theta)^T \boldsymbol{\Phi}(\phi)^T\}. \tag{16}$$

Define $\tilde{F}_t(\phi,\tilde{\chi},x,\theta) \triangleq p_{\tilde{z}_t|x_t,\theta_t,\phi_t}(\tilde{\chi}\mathrm{y}_t \mid x,\theta,\phi)$. Hence from (16) we get:

$$\tilde{F}_t(\phi,\tilde{\chi},x,\theta) = [(2\pi)^{mD(\phi)}\mathrm{Det}\{\tilde{Q}_t(\phi,\theta)\}]^{-1/2} \cdot \exp\{-\tfrac{1}{2}\tilde{\nu}_t^T(\phi,\tilde{\chi},x,\theta)\tilde{Q}_t(\phi,\theta)^{-1}\tilde{\nu}_t(\phi,\tilde{\chi},x,\theta)\}. \tag{17}$$

where

$$\tilde{\nu}_t(\phi,\tilde{\chi},x,\theta) \triangleq \tilde{\chi}\mathrm{y}_t - \mathbf{\Phi}(\phi)H(\theta)x$$

$$\tilde{Q}_t(\phi,\theta) \triangleq \mathbf{\Phi}(\phi)(G(\theta)G(\theta)^T)\mathbf{\Phi}(\phi)^T.$$

Substituting (17) into (15) and rearranging terms yields

$$p_{x_t,\theta_t|Y_t}(x,\theta) = \frac{1}{c_t}\sum_{\tilde{\chi},\phi}\left[\tilde{F}_t(\phi,\tilde{\chi},x,\theta)\cdot\prod_{j=1}^{L_t-D(\phi)}\lambda([\Phi(1_{L_t}-\tilde{\chi}^T\tilde{\chi}1_{L_t})\mathrm{y}_t]_j) \cdot\prod_{i=1}^{M}[(1-P_{\mathrm{d}}^i)^{(1-\phi_i)}(P_{\mathrm{d}}^i)^{\phi_i}]\right]\cdot p_{x_t,\theta_t|Y_{t-1}}(x,\theta). \tag{18}$$

THEOREM 1 *For any $\phi\in\{0,1\}^M$, such that $D(\phi)\triangleq\sum_{i=1}^M\phi_i\le L_t$, the following recursive equation holds true for the conditional density $p_{x_t,\theta_t|Y_t}(x,\theta)$:*

$$\begin{aligned}p_{x_t,\theta_t|Y_t}(x,\theta) &= \frac{1}{c_t}\sum_{\phi\in\{0,1\}^M}\left[\prod_{i=1}^{M}[(1-P_{\mathrm{d}}^i)^{(1-\phi_i)}(P_{\mathrm{d}}^i)^{\phi_i}]\right.\\ &\cdot\sum_{\tilde{\chi}}N_{mD(\phi)}\{\tilde{\chi}\mathrm{y}_t;\mathbf{\Phi}(\phi)H(\theta)x,\mathbf{\Phi}(\phi)G(\theta)G(\theta)^T\mathbf{\Phi}(\phi)^T\}\\ &\left.\cdot\prod_{j=1}^{L_t-D(\phi)}\lambda([\Phi(1_{L_t}-\tilde{\chi}^T\tilde{\chi}1_{L_t})\mathrm{y}_t]_j)\right]\\ &\cdot\int_{\mathbb{R}^{Mn}}N_{Mn}\{x;A(\theta)x',B(\theta)B(\theta)^T\}\\ &\cdot\sum_{\eta\in\mathbb{M}^M}[\Pi_{\eta\theta}p_{x_{t-1},\theta_{t-1}|Y_{t-1}}(x',\eta)]dx'\end{aligned} \tag{19}$$

with normalization c_t, $N_K\{\cdot;\bar{x},\bar{P}\}$ a K-dimensional Gaussian with mean $\bar{x}$ and covariance $\bar{P}$, and the 2nd sum running over all $\tilde{\chi}=\chi\Phi(\psi)$ with χ a $D(\phi)\times D(\phi)$ permutation matrix and $\psi\in\{0,1\}^{L_t}$ such that $D(\psi)=D(\phi)$.

PROOF IMM's basic derivation [38, App. A] yields:

$$p_{x_t,\theta_t|Y_{t-1}}(x,\theta) = \int_{\mathbb{R}^{Mn}}N_{Mn}\{x;A(\theta)x',B(\theta)B(\theta)^T\}\cdot\sum_{\eta\in\{1,\ldots,N\}^M}[\Pi_{\eta\theta}p_{x_{t-1},\theta_{t-1}|Y_{t-1}}(x',\eta)]dx'. \tag{20}$$

Substituting (17) and (20) in (18) and rearranging the summation over $\tilde{\chi}$ yields (19).

Equation (19) is a recursive equation for the exact Bayesian solution for tracking multiple targets from possibly false and missing measurements. From (19) it follows that if the initial density is a Gaussian mixture, then the exact conditional density solution of recursive equation (19) is a Gaussian mixture, the number of Gaussians increasing exponentially with time.

REMARK 1 For jump-linear systems such recursive filter equations have been characterized by [23], and for jump-non-linear systems by [16], [3]. In [14] we provide a version of Theorem 1 under the assumption that λ is homogeneous.

REMARK 2 Proposition 1 and Theorem 1 also apply when the initial densities are permutation symmetric over the targets, i.e. a situation studied by [32].

5. SIR JOINT PARTICLE FILTER

In this section a SIR joint particle filter of the exact filter characterization of Theorem 1 is developed. In this SIR joint PF a particle is defined as a triplet (μ_j,x_j,θ_j), $\mu_j\in[0,1]$, $x_j\in\mathbb{R}^{Mn}$, $\theta_j\in\mathbb{M}^M$, $j\in[1,S]$. One filter cycle consists of the following steps:

- *SIR joint particle filter Step 0:* Initiation.
  Each filter cycle starts with a set of S joint particles in $[0,1]\times\mathbb{R}^{Mn}\times\mathbb{M}^M$, i.e.:
  $$\{(\mu_{j,t-1}=1/S,x_{j,t-1},\theta_{j,t-1});\ j\in[1,S]\}$$
  with, for $t=0$, $\theta_{j,0}$ and $x_{j,0}$ independently drawn from $p_{\theta_0}(\cdot)$ and $p_{x_0|\theta_0}(\cdot\mid\theta_{j,0})$ respectively for each $j\in[1,S]$.
- *SIR joint particle filter Step 1:* Joint mode switching.
  Determine the new joint mode per joint particle ($\mu_{j,t-1}$ and $x_{j,t-1}$ are not changed)
  $$\{(\mu_{j,t-1},x_{j,t-1},\bar{\theta}_{j,t});\ j\in[1,S]\}$$
  by generating for each joint particle a new value $\bar{\theta}_{j,t}$ according to the transition probabilities:
  $$\mathrm{Prob}\{\bar{\theta}_{j,t}=\bar{\theta}\mid\theta_{j,t-1}=\theta\}=\Pi_{\theta,\bar{\theta}}. \tag{21}$$
- *SIR joint particle filter Step 2:* Prediction.
  Determine the new state per joint particle (the weights $\mu_{j,t-1}$ are not changed)
  $$\{(\mu_{j,t-1},\bar{x}_{j,t},\bar{\theta}_{j,t});\ j\in[1,S]\}$$
  by running for each particle a Monte Carlo simulation from $(t-1)$ to t according to the model
  $$\bar{x}_{j,t}=A(\bar{\theta}_{j,t})x_{j,t-1}+B(\bar{\theta}_{j,t})w_{j,t-1}. \tag{22}$$
- *SIR joint particle filter Step 3:* Measurement update.
  Determine new weight per joint particle, i.e.
  $$\{(\bar{\mu}_{j,t},\bar{x}_{j,t},\bar{\theta}_{j,t});\ j\in[1,S]\}$$

with, for the new weights, using (17) and (18):

$$\bar{\mu}_{j,t} = \mu_{j,t-1} \cdot \frac{1}{c_t} \sum_{\tilde{\chi},\phi} \left[\tilde{F}_t(\phi, \tilde{\chi}, \bar{x}_{j,t}, \bar{\theta}_{j,t}) \cdot \prod_{i=1}^{L_t - D(\phi)} \lambda([\Phi(1_{L_t} - \tilde{\chi}^T \tilde{\chi} 1_{L_t}) \mathrm{y}_t]_i) \cdot \prod_{i=1}^{M} [(1 - P_{\mathrm{d}}^i)^{(1-\phi_i)} (P_{\mathrm{d}}^i)^{\phi_i}] \right] \quad (23)$$

where

$$\tilde{F}_t(\phi, \tilde{\chi}, x, \theta) = [(2\pi)^{mD(\phi)} \mathrm{Det}\{\tilde{Q}_t(\phi, \theta)\}]^{-1/2} \cdot \exp\{-\tfrac{1}{2} \tilde{\nu}_t^T(\phi, \tilde{\chi}, x, \theta) \tilde{Q}_t(\phi, \theta)^{-1} \tilde{\nu}_t(\phi, \tilde{\chi}, x, \theta)\} \quad (24)$$

with

$$\tilde{\nu}_t(\phi, \tilde{\chi}, x, \theta) \stackrel{\Delta}{=} \tilde{\chi} \mathrm{y}_t - \mathbf{\Phi}(\phi) H(\theta) x$$

$$\tilde{Q}_t(\phi, \theta) \stackrel{\Delta}{=} \mathbf{\Phi}(\phi)(G(\theta) G(\theta)^T) \mathbf{\Phi}(\phi)^T$$

and c_t a normalizing constant such that

$$\sum_{j=1}^{S} \bar{\mu}_{j,t} = 1$$

- *SIR joint particle filter Step 4:* MMSE output equations:

$$\hat{\gamma}_t(\theta) = \sum_{j=1}^{S} \bar{\mu}_{j,t} 1_{\bar{\theta}_{j,t}}(\theta)$$

$$\hat{x}_t(\theta) = \sum_{j=1}^{S} \bar{\mu}_{j,t} \bar{x}_{j,t} 1_{\bar{\theta}_{j,t}}(\theta)$$

$$\hat{P}_t(\theta) = \sum_{j=1}^{S} \bar{\mu}_{j,t} [\bar{x}_{j,t} - \hat{x}_t(\theta)][\bar{x}_{j,t} - \hat{x}_t(\theta)]^T 1_{\bar{\theta}_{j,t}}(\theta)$$

$$\hat{x}_t = \sum_{\theta \in \mathbb{M}^M} \hat{\gamma}(\theta) \hat{x}_t(\theta)$$

$$\hat{P}_t = \sum_{\theta \in \mathbb{M}^M} \hat{\gamma}(\theta) [\hat{P}_t(\theta) + [\hat{x}_t(\theta) - \hat{x}_t][\hat{x}_t(\theta) - \hat{x}_t]^T]$$

- *SIR joint particle filter Step 5:* Resampling.
Generate the new set of joint particles

$$\{(\mu_{j,t} = 1/S, x_{j,t}, \theta_{j,t});\ j \in [1, S]\}$$

with $\theta_{j,t}$ and $x_{j,t}$ the jth of the S samples drawn independently from the joint particle spanned conditional densities for θ_t given y_t and for x_t given Y_t and $\theta_t = \theta_t^j$:

$$p_{\theta_t | Y_t}(\theta) \approx \hat{\gamma}_t(\theta)$$

$$p_{x_t | \theta_t, Y_t}(\cdot \mid \theta_{j,t}) \approx \sum_{l=1}^{S} \bar{\mu}_t^l 1_{\bar{\theta}_{l,t}}(\theta_{j,t}) \delta_{\bar{x}_{l,t}}(\cdot).$$

In the next section we modify the enumeration of the particles and adopt the particle resampling Step 5.

6. HYBRID SIR JOINT PARTICLE FILTER

In this section a hybrid SIR joint particle filter of the exact filter characterization of Theorem 1 is developed. The difference with the SIR joint particle filter is that we now resample a fixed number of joint particles per joint mode. A joint particle is defined as a triplet $(\mu^{\theta,j}, x^{\theta,j}, \theta)$, $\mu^{\theta,j} \in [0,1]$, $x^{\theta,j} \in \mathbb{R}^{Mn}$, $\theta \in \mathbb{M}^M$, $j \in [1, S']$. One cycle of this hybrid SIR joint particle filter consists of the following steps:

- *Hybrid SIR joint particle filter Step 0:* Initiation.
Each filter cycle starts with a set of $S = NS'$ joint particles in $[0,1] \times \mathbb{R}^{Mn} \times \mathbb{M}^M$, i.e.:

$$\{(\mu_{t-1}^{\theta,j}, x_{t-1}^{\theta,j}, \theta_{t-1}^{\theta,j} = \theta);\ j \in [1, S'],\ \theta \in \mathbb{M}^M\}$$

with, for $t = 0$, $\mu_0^{\theta,j} = p_{\theta_0}(\theta)/S'$, and $x_0^{\theta,j}$ independently drawn from $p_{x_0|\theta_0}(\cdot \mid \theta)$ for each $j \in 1, \ldots, S'$.

- *Hybrid SIR joint particle filter Step 1:* Mode switching.
Determine the new mode per particle ($\mu_{t-1}^{\theta,j}$ and $x_{t-1}^{\theta,j}$ are not changed)

$$\{(\mu_{t-1}^{\theta,j}, x_{t-1}^{\theta,j}, \bar{\theta}_t^{\theta,j});\ j \in [1, S'],\ \theta \in \mathbb{M}^M\}$$

by generating for each joint particle a new value $\bar{\theta}_t^{\theta,j}$ according to the model

$$\mathrm{Prob}\{\bar{\theta}_t^{\theta,j} = \bar{\theta} \mid \theta_{t-1}^{\theta,j} = \theta\} = \Pi_{\theta,\bar{\theta}}. \quad (25)$$

- *Hybrid SIR joint particle filter Step 2:* Prediction.
Determine the new state per joint particle (the weights $\mu_{t-1}^{\theta,j}$ are not changed)

$$\{(\mu_{t-1}^{\theta,j}, \bar{x}_t^{\theta,j}, \bar{\theta}_t^{\theta,j});\ j \in [1, S'],\ \theta \in \mathbb{M}^M\}$$

by running for each particle a Monte Carlo simulation from $(t-1)$ to t according to the model

$$\bar{x}_t^{\theta,j} = A(\bar{\theta}_t^{\theta,j}) x_{t-1}^{\theta,j} + B(\bar{\theta}_t^{\theta,j}) w_{t-1}. \quad (26)$$

- *Hybrid SIR joint particle filter Step 3:* Measurement update.
Determine new weight per joint particle, i.e.

$$\{(\bar{\mu}_t^{\theta,j}, \bar{x}_t^{\theta,j}, \bar{\theta}_t^{\theta,j});\ j \in [1, S'],\ \theta \in \mathbb{M}^M\}$$

with for the new weights, using (17) and (18):

$$\bar{\mu}_t^{\theta,j} = \mu_{t-1}^{\theta,j} \cdot \frac{1}{c_t} \sum_{\tilde{\chi},\phi} \left[\tilde{F}_t(\phi, \tilde{\chi}, \bar{x}_t^{\theta,j}, \bar{\theta}_t^{\theta,j}) \cdot \prod_{i=1}^{L_t - D(\phi)} \lambda([\Phi(1_{L_t} - \tilde{\chi}^T \tilde{\chi} 1_{L_t}) \mathrm{y}_t]_i) \cdot \prod_{i=1}^{M} [(1 - P_{\mathrm{d}}^i)^{(1-\phi_i)} (P_{\mathrm{d}}^i)^{\phi_i}] \right] \quad (27)$$

where

$$\tilde{F}_t(\phi,\tilde{\chi},x,\theta) = [(2\pi)^{mD(\phi)}\mathrm{Det}\{\tilde{Q}_t(\phi,\theta)\}]^{-1/2} \cdot \exp\{-\tfrac{1}{2}\tilde{\nu}_t^T(\phi,\tilde{\chi},x,\theta)\tilde{Q}_t(\phi,\theta)^{-1}\tilde{\nu}_t(\phi,\tilde{\chi},x,\theta)\} \tag{28}$$

with

$$\tilde{\nu}_t(\phi,\tilde{\chi},x,\theta) \triangleq \tilde{\chi}\mathrm{y}_t - \Phi(\phi)H(\theta)x$$

$$\tilde{Q}_t(\phi,\theta) \triangleq \Phi(\phi)(G(\theta)G(\theta)^T)\Phi(\phi)^T$$

and c_t a normalizing constant such that

$$\sum_{\theta\in\mathbb{M}^M}\sum_{j=1}^{S'}\bar{\mu}_t^{\theta,j} = 1.$$

- *Hybrid SIR joint particle filter Step 4:* MMSE output equations:

$$\hat{\gamma}_t(\theta) = \sum_{\eta\in\mathbb{M}^M}\sum_{j=1}^{S'}\bar{\mu}_t^{\eta,j}1_{\bar{\theta}_t^{\eta,j}}(\theta)$$

$$\hat{x}_t(\theta) = \sum_{\eta\in\mathbb{M}^M}\sum_{j=1}^{S'}\bar{\mu}_t^{\eta,j}\bar{x}_t^{\eta,j}1_{\bar{\theta}_t^{\eta,j}}(\theta)$$

$$\hat{P}_t(\theta) = \sum_{\eta\in\mathbb{M}^M}\sum_{j=1}^{S'}\bar{\mu}_t^{\eta,j}[\bar{x}_t^{\eta,j}-\hat{x}_t(\theta)][\bar{x}_t^{\eta,j}-\hat{x}_t(\theta)]^T1_{\bar{\theta}_t^{\eta,j}}(\theta)$$

$$\hat{x}_t = \sum_{\theta\in\mathbb{M}^M}\hat{\gamma}(\theta)\hat{x}_t(\theta)$$

$$\hat{P}_t = \sum_{\theta\in\mathbb{M}^M}\hat{\gamma}(\theta)[\hat{P}_t(\theta)+[\hat{x}_t(\theta)-\hat{x}_t][\hat{x}_t(\theta)-\hat{x}_t]^T].$$

- *Hybrid SIR joint particle filter Step 5:* Resampling per mode.
  Generate the new set of joint particles

$$\{(\mu_t^{\theta,j}=\hat{\gamma}_t(\theta)/S', x_t^{\theta,j}, \theta_t^{\theta,j}=\theta);\ j\in[1,S'],\ \theta\in\mathbb{M}^M\}$$

with $x_t^{\theta,j}$ the jth of the S' samples drawn independently from the particle spanned conditional density for x_t given Y_t and $\theta_t=\theta$:

$$p_{x_t|\theta_t,Y_t}(\cdot\mid\theta) \approx \sum_{\eta\in\mathbb{M}^M}\sum_{l=1}^{S'}\bar{\mu}_t^{\eta,l}1_{\bar{\theta}_t^{\eta,l}}(\theta)\delta_{\bar{x}_t^{\eta,l}}(x).$$

For homogeneous λ, this hybrid SIR joint particle filter has been introduced in [11] under the name Joint IMMPDA particle filter.

7. IMMJPDA ASSUMPTIONS

The assumptions that are underlying to the IMMJPDA of [18] are:

C1) $p_{\theta_t|Y_{t-1}}(\theta) = \prod_{i=1}^M p_{\theta_t^i|Y_{t-1}}(\theta^i)$;

C2) $p_{x_t|\theta_t,Y_{t-1}}(x\mid\theta) = \prod_{i=1}^M p_{x_t^i|\theta_t^i,Y_{t-1}}(x^i\mid\theta^i)$;

C3) $p_{x_t^i|\theta_t^i,Y_{t-1}}(x^i\mid\theta^i)$ is Gaussian with mean $\bar{x}_t^i(\theta^i)$ and covariance $\bar{P}_t^i(\theta^i)$.

Application of these assumptions, to the exact equations of Proposition 1 yields the following theorem.

THEOREM 2 *Assume* C1, C2 *and* C3 *are satisfied. Then* $\beta_t(\phi,\tilde{\chi},\theta)$ *of Proposition* 1 *satisfies:*

$$\beta_t(\phi,\tilde{\chi},\theta) = \left[\prod_{i=1}^{L_t-D(\phi)}\lambda([\Phi(1_{L_t}-\tilde{\chi}^T\tilde{\chi}1_{L_t})\mathrm{y}_t]_i)\right] \cdot\prod_{i=1}^M[f_t^i(\phi,\tilde{\chi},\theta^i)(1-P_{\mathrm{d}}^i)^{(1-\phi_i)}(P_{\mathrm{d}}^i)^{\phi_i}\cdot p_{\theta_t^i|Y_{t-1}}(\theta^i)]/c_t \tag{29}$$

with, for $\phi^i=0$*:* $f_t^i(\phi,\tilde{\chi},\theta^i)=1$*, and for* $\phi^i=1$*:*

$$f_t^i(\phi,\tilde{\chi},\theta^i) = [(2\pi)^m\mathrm{Det}\{\bar{Q}_t^i(\theta^i)\}]^{-\phi_i/2} \cdot\exp\left\{-\frac{1}{2}\sum_{k=1}^{L_t}([\Phi(\phi)^T]_{i*}\tilde{\chi}_{*k}\nu_t^{ik}(\theta^i)^T[\bar{Q}_t^i(\theta^i)]^{-1}\nu_t^{ik}(\theta^i))\right\} \tag{30a}$$

$$\nu_t^{ik}(\theta^i) = \mathrm{y}_t^k - h^i(\theta^i)\bar{x}_t^i(\theta^i) \tag{30b}$$

$$\bar{Q}_t^i(\theta^i) = h^i(\theta^i)\bar{P}_t^i(\theta^i)h^i(\theta^i)^T + g^i(\theta^i)g^i(\theta^i)^T \tag{30c}$$

where $[\Phi(\phi)^T]_{i*}$ *and* $\tilde{\chi}_{*k}$ *are the* i*th row and* k*th column of* $\Phi(\phi)^T$ *and* $\tilde{\chi}$*, respectively. Moreover,* $p_{x_t^i|\theta_t^i,Y_t}(x^i\mid\theta^i)$*,* $i\in[1,M]$*, is a Gaussian mixture, while its overall mean* $\hat{x}_t^i(\theta^i)$ *and its overall covariance* $\hat{P}_t^i(\theta^i)$ *satisfy:*

$$p_{\theta_t^i|Y_t}(\theta^i) = \sum_{\substack{\phi,\tilde{\chi},\eta\\ \eta^i=\theta^i}}\beta_t(\phi,\tilde{\chi},\eta) \tag{31a}$$

$$\hat{x}_t^i(\theta^i) = \bar{x}_t^i(\theta^i) + W_t^i(\theta^i)\left(\sum_{k=1}^{L_t}\beta_t^{ik}(\theta^i)\nu_t^{ik}(\theta^i)\right) \tag{31b}$$

$$\hat{P}_t^i(\theta^i) = \bar{P}_t^i(\theta^i) - W_t^i(\theta^i)h^i(\theta^i)\bar{P}_t^i(\theta^i)\left(\sum_{k=1}^{L_t}\beta_t^{ik}(\theta^i)\right) + W_t^i(\theta^i)\left(\sum_{k=1}^{L_t}\beta_t^{ik}(\theta^i)\nu_t^{ik}(\theta^i)\nu_t^{ik}(\theta^i)^T\right)W_t^i(\theta^i)^T - W_t^i(\theta^i)\left(\sum_{k=1}^{L_t}\beta_t^{ik}(\theta^i)\nu_t^{ik}(\theta^i)\right) \cdot\left(\sum_{k'=1}^{L_t}\beta_t^{ik'}(\theta^i)\nu_t^{ik'}(\theta^i)\right)^T W_t^i(\theta^i)^T \tag{31c}$$

with:

$$W_t^i(\theta^i) = \bar{P}_t^i(\theta^i)h^i(\theta^i)^T[\bar{Q}_t^i(\theta^i)]^{-1} \tag{31d}$$

$$\begin{aligned}\beta_t^{ik}(\theta^i) &\triangleq \text{Prob}\{[\Phi(\phi_t)^T]_{i*}[\tilde{\chi}_t]_{*k} = 1 \mid \theta_t^i = \theta^i, Y_t\} \\ &= \sum_{\substack{\phi,\tilde{\chi},\eta \\ \phi\neq 0 \\ \eta^i=\theta^i}} [\Phi(\phi)^T]_{i*}\tilde{\chi}_{*k}\beta_t(\phi,\tilde{\chi},\eta)/p_{\theta_t^i|Y_t}(\theta^i).\end{aligned} \tag{31e}$$

PROOF See Appendix B.

Equation (30a) replaces six nested equations of [18, eqs. (18) and (20)–(24)]. As a direct consequence, Theorem 2 leads to a more compact version of IMMJPDA, the detailed steps of which we give in the next section.

8. TRACK-COALESCENCE-AVOIDING IMMJPDA FILTER

Fitzgerald [24] has shown that less likely permutation hypotheses pruning provides an effective strategy towards reducing JPDA's sensitivity to track coalescence if $\lambda = 0$ and $P_d^i = 1$. In [8] we have shown that for $\lambda > 0$ or $P_d^i < 1$, the appropriate strategy is to prune per (ϕ_t, ψ_t)-hypothesis all but the most likely χ_t-hypothesis prior to measurement updating. This hypothesis pruning strategy is now extended as follows: evaluate all $(\phi_t, \psi_t, \theta_t)$ hypotheses and prune per $(\phi_t, \psi_t, \theta_t)$-hypothesis all but the most-likely χ_t-hypothesis. For every ϕ, ψ and θ, satisfying $D(\psi) = D(\phi) \leq \min\{M, L_t\}$, the most likely χ hypothesis satisfies the mapping $\hat{\chi}_t(\phi,\psi,\theta)$:

$$\hat{\chi}_t(\phi,\psi,\theta) \triangleq \arg\max_{\chi} \beta_t(\phi, \chi^T\Phi(\psi), \theta)$$

where the maximization is over all permutation matrices χ of size $D(\phi) \times D(\phi)$.

The pruning strategy of evaluating all (ϕ,ψ,θ)-hypotheses and only one χ-hypothesis per (ϕ,ψ,θ)-hypothesis implies that we adopt the following pruned hypothesis weights $\hat{\beta}_t(\phi,\psi,\theta)$:

$$\begin{aligned}\hat{\beta}_t(\phi,\psi,\theta) &= \beta_t(\phi, \hat{\chi}(\phi,\psi,\theta)^T\Phi(\psi), \theta)/\hat{c}_t \\ &\qquad \text{if} \quad 0 < D(\phi) \leq \min\{M, L_t\} \\ &= \beta_t(\{0\}^M, \{\}^{L_t}, \theta)/\hat{c}_t \qquad \text{if} \quad D(\phi) = 0 \\ &= 0 \qquad\qquad \text{else}\end{aligned}$$

with $\hat{c}_t$ a normalization constant for $\hat{\beta}_t$; i.e. such that

$$\sum_{\substack{\phi,\psi,\theta \\ D(\psi)=D(\phi)}} \hat{\beta}_t(\phi,\psi,\theta) = 1.$$

Through combining the equations of Theorem 2 with the above step, we arrive at the track-coalescence-avoiding IMMJPDA, for short IMMJPDA*:

IMMJPDA Step 1:* For each target this comes down to the interaction step of the IMM algorithm [7] for all $i \in [1, M]$: Starting with

$$\hat{\gamma}_{t-1}^i(\theta^i) \triangleq p_{\theta_{t-1}^i|Y_{t-1}}(\theta^i), \qquad \theta^i \in \mathbb{M}$$

$$\hat{x}_{t-1}^i(\theta^i) \triangleq E\{x_{t-1}^i \mid \theta_{t-1}^i = \theta^i, Y_{t-1}\}, \qquad \theta^i \in \mathbb{M}$$

$$\begin{aligned}\hat{P}_{t-1}^i(\theta^i) \triangleq E\{&[x_{t-1}^i - \hat{x}_{t-1}^i(\theta^i)] \\ &\cdot [x_{t-1}^i - \hat{x}_{t-1}^i(\theta^i)]^T \mid \theta_{t-1}^i = \theta^i, Y_{t-1}\}, \qquad \theta^i \in \mathbb{M}\end{aligned}$$

one evaluates the mixed initial condition for the filter matched to $\theta_t^i = \theta^i$ as follows (due to (4)):

$$\bar{\gamma}_t^i(\theta^i) = \sum_{\eta^i=1}^N \Pi_{\eta^i,\theta^i}^i \cdot \hat{\gamma}_{t-1}^i(\eta^i)$$

$$\hat{x}_{t-1|\theta_t^i}^i(\theta^i) = \sum_{\eta^i=1}^N \Pi_{\eta^i,\theta^i}^i \cdot \hat{\gamma}_{t-1}^i(\eta^i) \cdot \hat{x}_{t-1}^i(\eta^i)/\bar{\gamma}_t^i(\theta^i)$$

$$\begin{aligned}\hat{P}_{t-1|\theta_t^i}^i(\theta^i) = \sum_{\eta^i=1}^N \Pi_{\eta^i,\theta^i}^i \cdot \hat{\gamma}_{t-1}^i(\eta^i) \\ \cdot [\hat{P}_{t-1}^i(\eta^i) + [\hat{x}_{t-1}^i(\eta^i) - \hat{x}_{t-1|\theta_t^i}^i(\theta^i)] \\ \cdot [\hat{x}_{t-1}^i(\eta^i) - \hat{x}_{t-1|\theta_t^i}^i(\theta^i)]^T]/\bar{\gamma}_t^i(\theta^i).\end{aligned}$$

IMMJPDA Step 2:* Prediction for all $i \in [1, M]$, $\theta^i \in \mathbb{M}$:

$$\bar{x}_t^i(\theta^i) = a^i(\theta^i)\hat{x}_{t-1|\theta_t^i}^i(\theta^i) \tag{32a}$$

$$\bar{P}_t^i(\theta^i) = a^i(\theta^i)\hat{P}_{t-1|\theta_t^i}^i(\theta^i)a^i(\theta^i)^T + b^i(\theta^i)b^i(\theta^i)^T \tag{32b}$$

$$\bar{Q}_t^i(\theta^i) = h^i(\theta^i)\bar{P}_t^i(\theta^i)h^i(\theta^i)^T + g^i(\theta^i)g^i(\theta^i)^T. \tag{32c}$$

IMMJPDA Step 3:* Gating, which is based on [5].

Identify for each target the mode for which $\text{Det}\,\bar{Q}_t^i(\theta)$ is largest:

$$\theta_t^{*i} = \arg\max_{\theta}\{\text{Det}\bar{Q}_t^i(\theta)\}$$

and use this to define for each target i a gate $G_t^i \in \mathbb{R}^m$ as follows:

$$\begin{aligned}G_t^i \triangleq \{z^i \in IR^m; &[z^i - h^i(\theta_t^{*i})\bar{x}_t^i(\theta_t^{*i})]^T \\ &\cdot \bar{Q}_t^i(\theta_t^{*i})^{-1}[z^i - h^i(\theta_t^{*i})\bar{x}_t^i(\theta_t^{*i})] \leq \kappa\}\end{aligned}$$

with κ the gate size. Now we define L_t to denote the number of measurements y_t that are in one or more of the gates G_t^i.

IMMJPDA Step 4:* Evaluation of the detection/association/mode hypotheses is based on Theorem 2; for all $\phi \in \{0,1\}^M$, $\tilde{\chi} \in \{0,1\}^{D(\phi)\times L_t}$, $\theta \in \mathbb{M}^M$,

$$\begin{aligned}\beta_t(\phi,\tilde{\chi},\theta) \cong & \left[\prod_{i=1}^{L_t-D(\phi)} \lambda([\Phi(1_{L_t}-\tilde{\chi}^T\tilde{\chi}1_{L_t})\mathrm{y}_t]_i)\right] \\ & \cdot \prod_{i=1}^{M}[f_t^i(\phi,\tilde{\chi},\theta^i)\cdot\bar{\gamma}_t^i(\theta^i) \\ & \cdot (1-P_{\mathrm{d}}^i\chi_m^2(\kappa))^{(1-\phi_i)}(P_{\mathrm{d}}^i\chi_m^2(\kappa))^{\phi_i}]/c_t \\ & \quad \text{if} \quad \tilde{\chi}1_{L_t} = 1_{D(\phi)} \\ = & \ 0 \qquad \text{else} \end{aligned} \tag{33a}$$

with for $\phi^i = 0$: $f_t^i(\phi,\tilde{\chi},\theta^i) = 1$, and for $\phi^i = 1$:

$$\begin{aligned} & f_t^i(\phi,\tilde{\chi},\theta^i) \\ & \quad \cong [(2\pi)^m \mathrm{Det}\{\bar{Q}_t^i(\theta^i)\}]^{-\phi_i/2} \\ & \qquad \cdot \exp\left\{-\frac{1}{2}\sum_{k=1}^{L_t}[\Phi(\phi)^T]_{i*}\tilde{\chi}_{*k}\nu_t^{ik}(\theta^i)^T[\bar{Q}_t^i(\theta^i)]^{-1}\nu_t^{ik}(\theta^i)]\right\} \end{aligned} \tag{33b}$$

$$\nu_t^{ik}(\theta^i) = \mathrm{y}_t^k - h^i(\theta^i)\bar{x}_t^i(\theta^i). \tag{33c}$$

IMMJPDA Step 5:* Track-coalescence hypothesis pruning.

First, evaluate for every (ϕ,ψ,θ) such that $0 < D(\psi) = D(\phi) \le \min\{M, L_t\}$:

$$\hat{\chi}_t(\phi,\psi,\theta) \stackrel{\Delta}{=} \arg\max_{\chi} \beta_t(\phi,\chi^T\Phi(\psi),\theta).$$

Next, evaluate all $\hat{\chi}_t(\phi,\psi,\theta)$ hypothesis weights:

$$\begin{aligned}\hat{\beta}_t(\phi,\psi,\theta) &= \beta_t(\phi,\hat{\chi}_t(\phi,\psi,\theta)^T\Phi(\psi),\theta)/\hat{c}_t \\ &\qquad \text{if} \quad 0 < D(\psi) = D(\phi) \le \min\{M,L_t\} \\ &= \beta_t(\{0\}^M,\{\}^{L_t},\theta)/\hat{c}_t \\ &\qquad \text{if} \quad D(\psi) = D(\phi) = 0 \\ &= 0 \qquad \text{else}\end{aligned}$$

where $\hat{c}_t$ is a normalizing constant for $\hat{\beta}_t$.

IMMJPDA Step 6:* Measurement update equations (also based on Theorem 2); for all $i \in [1,M]$, $\theta^i \in \mathbb{M}$,

$$\hat{\gamma}_t^i(\theta^i) \cong \sum_{\substack{\phi,\psi,\eta \\ \eta^i=\theta^i}} \hat{\beta}_t(\phi,\psi,\eta) \tag{34a}$$

$$\hat{x}_t^i(\theta^i) \cong \bar{x}_t^i(\theta^i) + W_t^i(\theta^i)\left(\sum_{k=1}^{L_t}\hat{\beta}_t^{ik}(\theta^i)\nu_t^{ik}(\theta^i)\right) \tag{34b}$$

$$\begin{aligned}\hat{P}_t^i(\theta^i) \cong & \ \bar{P}_t^i(\theta^i) - W_t^i(\theta^i)h^i(\theta^i)\bar{P}_t^i(\theta^i)\left(\sum_{k=1}^{L_t}\hat{\beta}_t^{ik}(\theta^i)\right) \\ & + W_t^i(\theta^i)\left(\sum_{k=1}^{L_t}\hat{\beta}_t^{ik}(\theta^i)\nu_t^{ik}(\theta^i)\nu_t^{ik}(\theta^i)^T\right)W_t^i(\theta^i)^T \\ & - W_t^i(\theta^i)\left(\sum_{k=1}^{L_t}\hat{\beta}_t^{ik}(\theta^i)\nu_t^{ik}(\theta^i)\right) \\ & \cdot \left(\sum_{k'=1}^{L_t}\hat{\beta}_t^{ik'}(\theta^i)\nu_t^{ik'}(\theta^i)\right)^T W_t^i(\theta^i)^T \end{aligned} \tag{34c}$$

with

$$W_t^i(\theta^i) = \bar{P}_t^i(\theta^i)h^i(\theta^i)^T[\bar{Q}_t^i(\theta^i)]^{-1} \tag{34d}$$

$$\begin{aligned}\hat{\beta}_t^{ik}(\theta^i) = & \left(\sum_{\substack{\phi,\psi,\eta \\ \phi,\psi\neq 0 \\ \eta^i=\theta^i}}[\Phi(\phi)^T]_{i*}[\hat{\chi}_t(\phi,\psi,\eta)^T\Phi(\psi)]_{*k}\right. \\ & \left.\cdot\hat{\beta}_t(\phi,\psi,\eta)\right)\Big/\hat{\gamma}_t^i(\theta^i)\end{aligned} \tag{34e}$$

where $[.]_{*k}$ is the kth column of $[.]$ and $[.]_{i*}$ is the ith row of $[.]$.

IMMJPDA Step 7:* Output equations:

$$\hat{x}_t^i = \sum_{\theta^i=1}^{N}\hat{\gamma}_t^i(\theta^i)\cdot\hat{x}_t^i(\theta^i) \tag{35a}$$

$$\hat{P}_t^i = \sum_{\theta^i=1}^{N}\hat{\gamma}_t^i(\theta^i)(\hat{P}_t^i(\theta^i) + [\hat{x}_t^i(\theta^i) - \hat{x}_t^i]\cdot[\hat{x}_t^i(\theta^i) - \hat{x}_t^i]^T). \tag{35b}$$

REMARK 3 By deleting the track coalescence hypothesis pruning Step 5 from IMMJPDA*, and by replacing $\hat{\beta}(\phi,\psi,\eta)$ by $\beta(\phi,\psi,\eta)$ in Steps 6 and 7, we get the compact IMMJPDA filter. As already announced in Remark 2, the reason to refer to compact IMMJPDA is that (33b) replaces six nested equations in the IMMJPDA of [18, eqs. (18) and (20)–(24)].

9. MONTE CARLO SIMULATIONS

In this section some Monte Carlo simulation results are given for the two novel joint particle filters, for the (compact) IMMJPDA and IMMJPDA* filter algorithms, and for a multi-target tracker using an IMM-PDA for each track. The two particle filters ran on a total of $S = 10^4$ joint particles. The simulations aim at gaining insight into the behavior and performance of the filters regarding track maintenance when two targets move in and out of close approach situations, while giving the filters enough time to converge after

a maneuver has taken place. In the example scenarios there are two tracked targets, each modeled with two possible modes. The first mode represents a constant velocity model and the second mode represents a constant acceleration model. It is assumed that both targets are initially tracked well, that for their initial track estimates there is no uncertainty regarding which track belongs to which target. Both objects move towards each other, each with constant initial velocity V_{initial}. At a certain moment in time both objects start decelerating with -50 m/s^2 until they both have zero velocity. The moment at which the deceleration starts is such that when the objects both have zero velocity, the distance between the two objects equals d (see Fig. 1). After spending a significant number of scans with zero velocity, both objects start accelerating with 50 m/s^2 away from each other without crossing until their velocity equals the opposite of their initial velocity. From that moment on the velocity of both objects remains constant again (thus the final relative velocity $V_{\text{rel, final}} = V_{\text{rel, initial}}$). Note that $d < 0$ implies that the objects have crossed each other before they have reached zero velocity. In each simulation the filters start with perfect estimates and run for 40 scans. Examples of the trajectories for $d \geq 0$ and $d < 0$ are depicted in Figs. 1(a) and 1(b) respectively.

For each target, the underlying model of the potential target measurements is given by (1) and (2), i.e.:

$$x_{t+1}^i = a^i(\theta_{t+1}^i)x_t^i + b^i(\theta_{t+1}^i)w_t^i$$
$$z_t^i = h^i(\theta_t^i)x_t^i + g^i(\theta_t^i)v_t^i$$

with for $i \in \{1,2\}$ and $\theta_t^i \in \{1,2\}$:

$$a^i(1) = \begin{bmatrix} 1 & T_s & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad a^i(2) = \begin{bmatrix} 1 & T_s & \frac{1}{2}T_s^2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}$$

$$b^i(1) = \sigma_a^i \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \qquad b^i(2) = \sigma_a^i \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$h^i = [1\ 0\ 0], \qquad g^i = \sigma_m^i$$

$$\Pi = \begin{bmatrix} 1 - T_s/\tau_1 & T_s/\tau_1 \\ T_s/\tau_2 & 1 - T_s/\tau_2 \end{bmatrix}$$

where σ_a^i represents the standard deviation of acceleration noise and σ_m^i represents the standard deviation of the measurement error. For simplicity we consider the situation of similar targets only; i.e., $\sigma_a^i = \sigma_a$, $\sigma_m^i = \sigma_m$, $P_{\text{d}}^i = P_{\text{d}}$. With this, the scenario parameters are P_{d}, λ, d, V_{initial}, T_s, σ_m, σ_a, τ_1, τ_2, and the gate size γ. We used fixed parameters $\sigma_m = 30$, $\sigma_a = 50$, $\tau_1 = 50$, $\tau_2 = 5$, and $\gamma = 25$. Table I gives the other scenario parameter values that are being used for the Monte Carlo simulations.

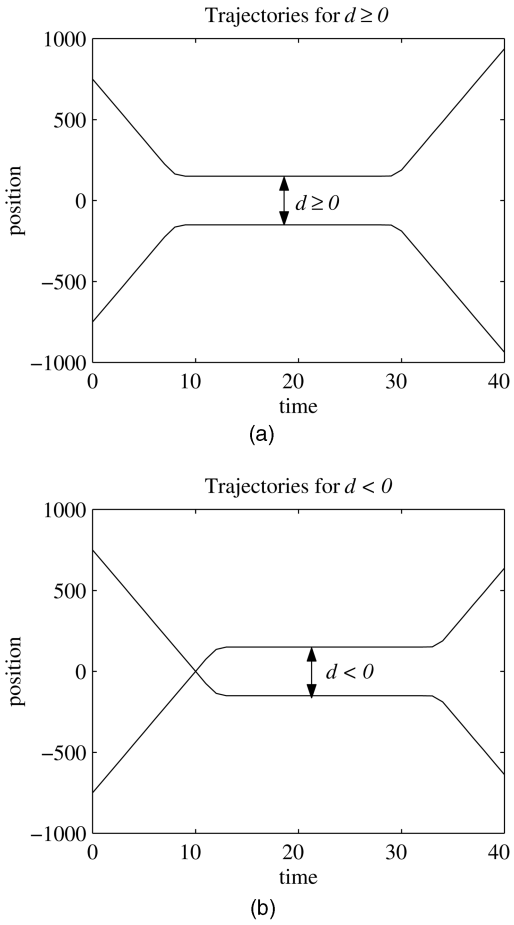


Fig. 1. Trajectories examples for $d \geq 0$ and for $d < 0$.

TABLE I
Scenario Parameter Values[1]

Scenario	P_{d}	λ	d	V_{initial}	T_s
1	1	0	Variable	75	1
2	1	0.001	Variable	75	1
3	0.9	0	Variable	75	1
4	0.9	0.001	Variable	75	1

[1]IMMPDA's $\lambda = 0.00001$ for scenarios 1 and 3.

During our simulations we counted track i "OK" if

$$|h^i\hat{x}_T^i - h^i x_T^i| \leq 9\sigma_m$$

and we counted track $i \neq j$ "Swapped" if

$$|h^i\hat{x}_T^i - h^j x_T^j| \leq 9\sigma_m.$$

Furthermore, two tracks $i \neq j$ are counted "Coalescing" at scan t, if

$$|h^i\hat{x}_t^i - h^j\hat{x}_t^j| \leq \sigma_m \wedge |h^i x_t^i - h^j x_t^j| > \sigma_m.$$

For each of the scenarios Monte Carlo simulations containing 100 runs have been performed for each of

the tracking filters. The initial track estimates are

$$\hat{x}_0^1(\theta) = \begin{bmatrix} -750 \\ 75 \\ 0 \end{bmatrix}$$

$$\hat{x}_0^2(\theta) = \begin{bmatrix} 750 \\ -75 \\ 0 \end{bmatrix}, \qquad \theta \in \{1,2\}$$

$$\hat{P}_0^i(1) = \begin{bmatrix} 100 & 0 & 0 \\ 0 & \frac{100}{9} & 0 \\ 0 & 0 & \frac{1}{9} \end{bmatrix}$$

$$\hat{P}_0^i(2) = \begin{bmatrix} 100 & 0 & 0 \\ 0 & \frac{100}{9} & 0 \\ 0 & 0 & \frac{1}{36} \end{bmatrix}$$

$$\hat{\gamma}_0^i(1) = 0.9, \qquad \hat{\gamma}_0^i(2) = 0.1 \qquad \text{for} \quad i = 1,2.$$

The results of the Monte Carlo simulations for the four scenarios are shown in tables and figures as follows:

- The percentage of Both tracks "OK," see Table II, and Figs. 2(a), 3(a), 4(a) and 5(a).
- The percentage of Both tracks "OK" or "Swapped," see Table III, and Figs. 2(b), 3(b), 4(b) and 5(b).
- The average number of "Coalescing" scans, see Table IV, and Figs. 2(c), 3(c), 4(c) and 5(c).
- The average CPU time per scan (in seconds), see Table V.

The results in Tables II–IV and Figs. 2–5 show that for targets that come close to each other, IMMJPDA, IMMJPDA* and the particle filters perform much better than IMMPDA. As expected, these simulation results show increased difficulty for $P_d = 0.9$ when compared to $P_d = 1$ and for $\lambda = 0.001$ when compared to $\lambda = 0$. Furthermore $\lambda = 0.001$ has more impact on the performance than $P_d = 0.9$. This can be explained by the fact that for $\lambda = 0.001$ a target track may diverge because of false measurements. The SIR-H joint particle filter suffers the least from this.

Measured in terms of "both tracks OK" (Table II and Figs. 2(a)–5(a)) the SIR-H joint particle filter performed best, the IMMJPDA* second best, the SIR-H joint particle filter third and the IMMJPDA fourth. The both tracks "OK" Figs. 2(a)–5(a) show a slight difference for $d < 0$ and $d > 0$. This is because for $d < 0$ the target trajectories cross each other before they have reached zero velocity, while for $d > 0$ they do not cross (see Fig. 1).

Figs. 2(a)–5(a) show that IMMJPDA and IMMJPDA* filters have oscillating variation in performance which is lacking for SIR-H joint particle filter. This phenomenon can be explained by the observation that the effect of "overshoot" during a maneuver is for IMMJPDA and IMMJPDA* more profound than for the SIR-H joint particle filter, because the latter filters perform time extrapolation from only one state estimate per mode, whereas the SIR-H joint particle filter performs time extrapolation for many particles per mode. The effect is that for some d values IMMJPDA and IMMJPDA* actually benefit from overshoot in the sense that it keeps the tracks separated, while for other d values the overshoot actually moves the tracks closer to each other. This effect is less profound for the SIR-H joint particle filters due to time extrapolation for many particles per mode; hence oscillating variation in performance does not occur.

Rather surprisingly, IMMJPDA* outperforms Hybrid SIR joint particle filter regarding the both tracks "OK" or "Swapped" criterion (Table III and Figs. 2(b)–5(b)) on the "easy" scenarios 1–3. Scenario 4 shows that IMMJPDA* is outperformed on this criterion by the SIR-H joint particle filter when missing and false measurement conditions become more challenging.

Table IV and Figs. 2(c)–5(c) show that IMMJPDA* performs best on track coalescence avoidance. Next best

TABLE II
Average % Both Tracks "OK"

Scen.	IMMPDA	IMMJPDA	IMMJPDA*	SIR Joint	SIR-H Joint
1	19	66	73	70	75
2	10	56	68	65	70
3	6	63	69	70	72
4	4	41	50	43	57

TABLE III
Average % Both Tracks "OK" or "Swapped"

Scen.	IMMPDA	IMMJPDA	IMMJPDA*	SIR Joint	SIR-H Joint
1	28.3	99.96	100	97.8	96.2
2	18.9	92.5	96.8	91.6	94.6
3	8.5	99.8	100	97.6	95.8
4	5.6	76.6	80.96	66.0	82.3

TABLE IV
Average Number of Coalescing Scans

Scen.	IMMPDA	IMMJPDA	IMMJPDA*	SIR Joint	SIR-H Joint
1	9.7	1.5	0.4	1.2	1.3
2	11.0	2.1	0.3	1.2	1.4
3	18.9	1.7	0.5	1.3	1.3
4	14.5	2.6	0.5	1.3	1.5

TABLE V
Average CPU Time Per Scan (in milliseconds)

Scen.	IMMPDA	IMMJPDA	IMMJPDA*	SIR Joint	SIR-H Joint
1	16	22	23	385	439
2	38	54	48	7245	7959
3	14	20	20	377	438
4	38	61	56	7170	7810

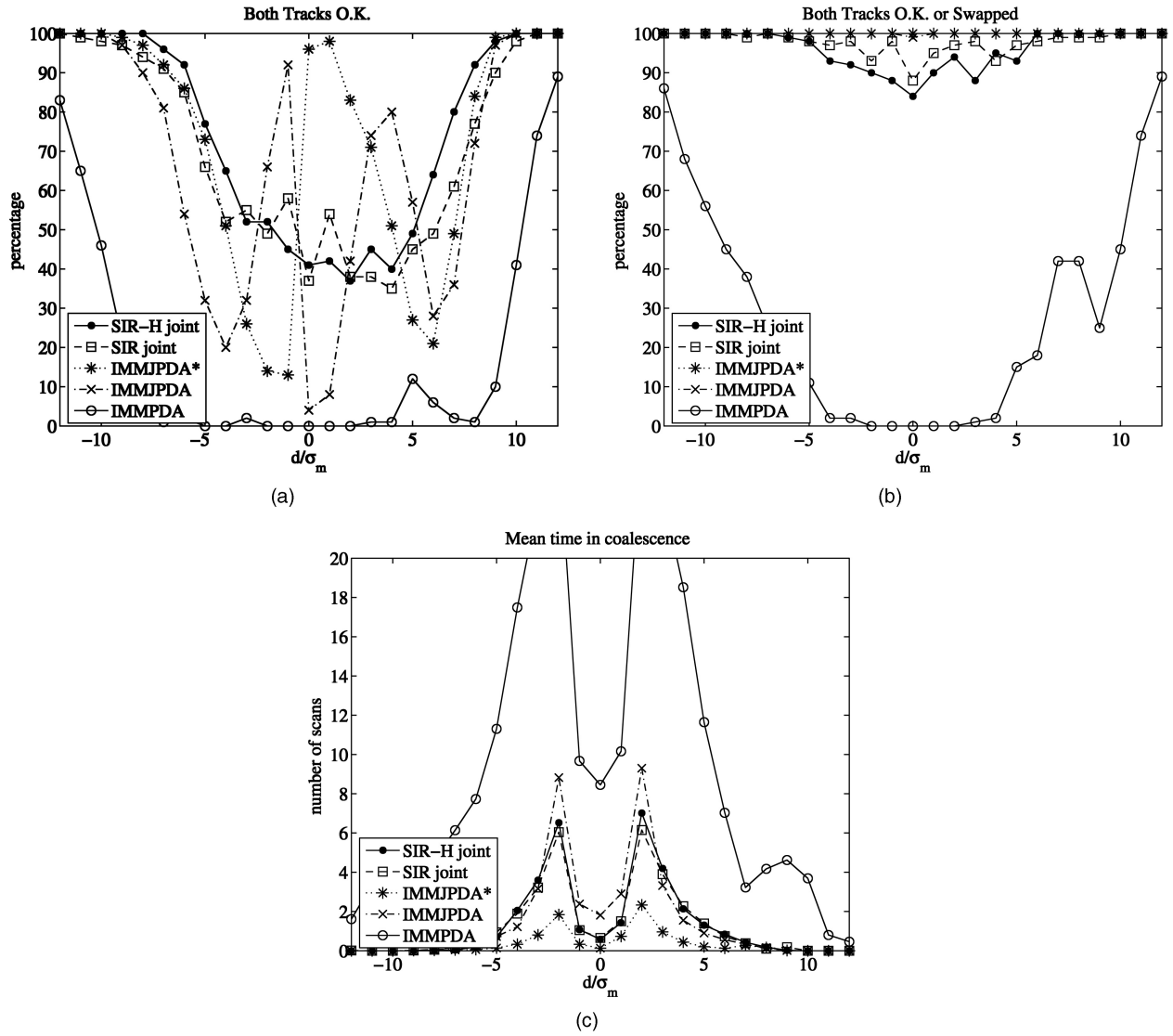


Fig. 2. Simulation results for scenario 1. (a) Both tracks "OK" percentage. (b) Both tracks "OK" or "Swapped" percentage. (c) Average number of "Coalescing" scans.

are the two particle filters, and fourth is IMMJPDA. The "dip" in "mean time in coalescence" around zero is due to the definition of "coalescing tracks." That is, when the targets are actually moving very close to each other, which is the case for small d values, there are no coalescing scans counted. Scans are only counted coalescing when the targets are separated from each other far enough.

Table V indicates a significant CPU-time increase for joint particle filters relative to the others. The increase is one order of magnitude for scenarios without clutter and two orders of magnitude for scenarios with clutter.

It should be noticed that there are various complementary methods available that allow to reduce the number of particles and/or CPU time significantly without reducing performance (e.g. [1], [38]). Hence when reading Table V one should be aware that these methods have not been investigated in this paper.

10. CONCLUDING REMARKS

In this paper we studied the problem of maneuvering target tracking from possibly missing and false measurements. The density of the false measurements was assumed to be non-homogeneous. For this problem we studied particle filtering as an alternative to multi-target track maintenance versions of IMM in combination with PDA or JPDA. The approach taken is to first characterize the problem in terms of filtering for a jump linear descriptor system with both Markovian and i.i.d. coefficients, and next to use this for the derivation of the exact recursive equation for the Bayesian filter (Theorem 1). This result has been used to develop two SIR type particle filters, one which resamples a fixed number of joint particles (SIR joint particle filter) and one which resamples a fixed number of joint particles per joint mode (SIR-H joint particle filter). We have also shown that application of the approximating assumptions of [18] to

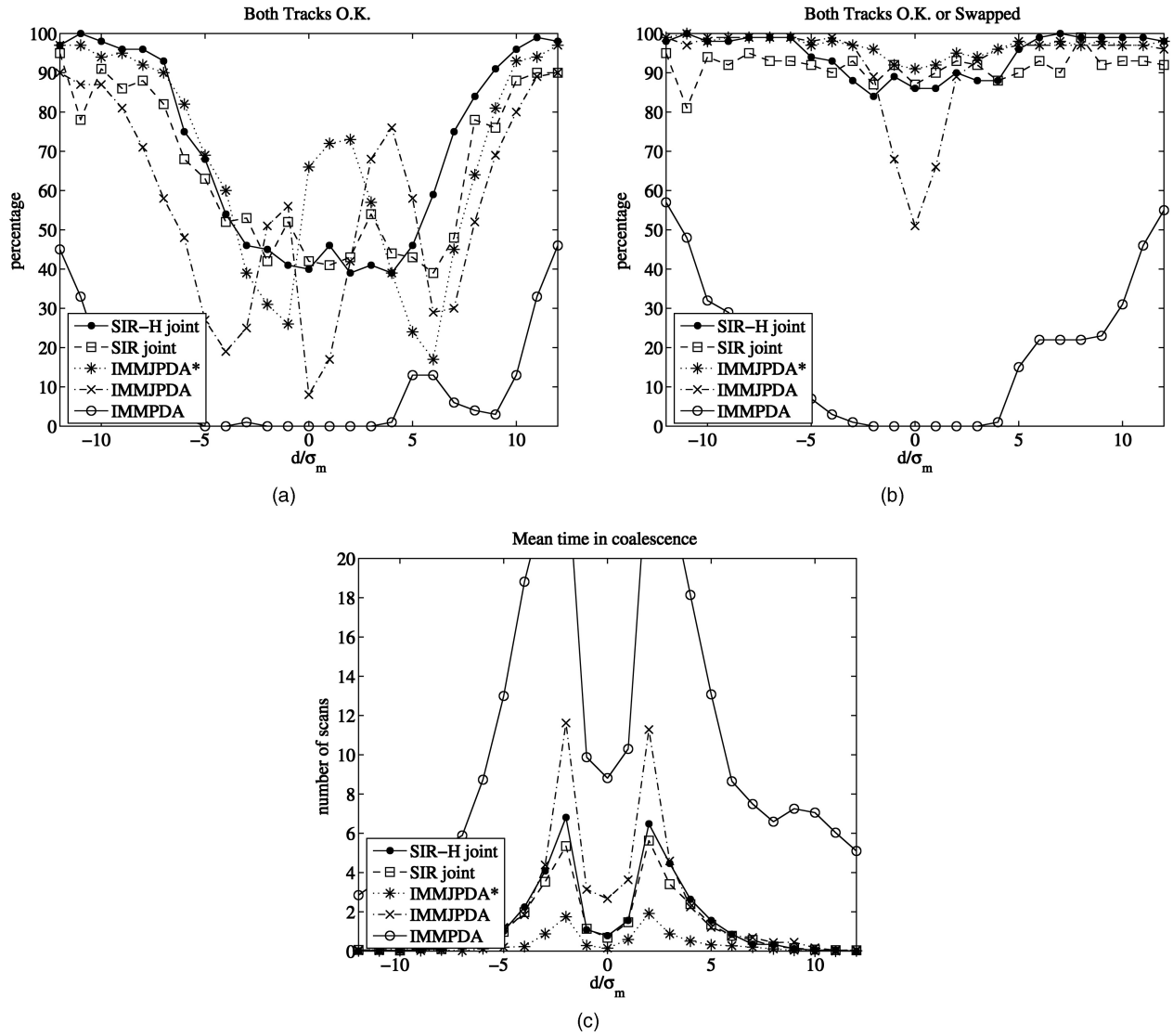


Fig. 3. Simulation results for scenario 2. (a) Both tracks "OK" percentage. (b) Both tracks "OK" or "Swapped" percentage. (c) Average number of "Coalescing" scans.

the exact Bayesian filter equations (Theorem 2) leads to a compact version of their IMMJPDA filter equations. For this (compact) IMMJPDA filter we also developed a track-coalescence-avoiding version (IMMJPDA*) by introduction of a particular pruning of permutation hypotheses. All our four novel filter algorithms cover the situation of non-homogeneous density of false measurements.

Through Monte Carlo simulations for a series of simple scenarios with two targets and two associated tracks these four novel filters have been compared to each other and to a filter which runs a single target IMMPDA (per track). All four clearly outperformed IMMPDA. The particle filters used 10^4 joint particles; with this the SIR-H joint particle filter appears to approximate the Bayesian filter well, whereas the SIR joint particle filter did not. On all scenarios, IMMJPDA* performs significantly better than IMMJPDA and sometimes even remarkably close to the performance of the SIR-H joint particle filter. Apparently, the performance reduction by the IMMJPDA approximation of the exact Bayesian filter can be partly compensated by introducing the additional IMMJPDA* approximation. IMMJPDA and IMMJPDA* both perform less well than the SIR-H joint PF on the following two points:

- The performance of both IMMJPDA and IMMJPDA* varies heavily with changes in the geometry of encountering target paths; this varying kind of behavior is not shown by the SIR-H joint particle filter;
- The SIR-H joint particle filter is least sensitive to divergence of track because of switching to running on false measurements; this advantage shows both when targets are clearly separated from each other and when target paths come close to each other.

Recently both [12] and [47] explored the potential effect on performance of extending IMMJPDA and

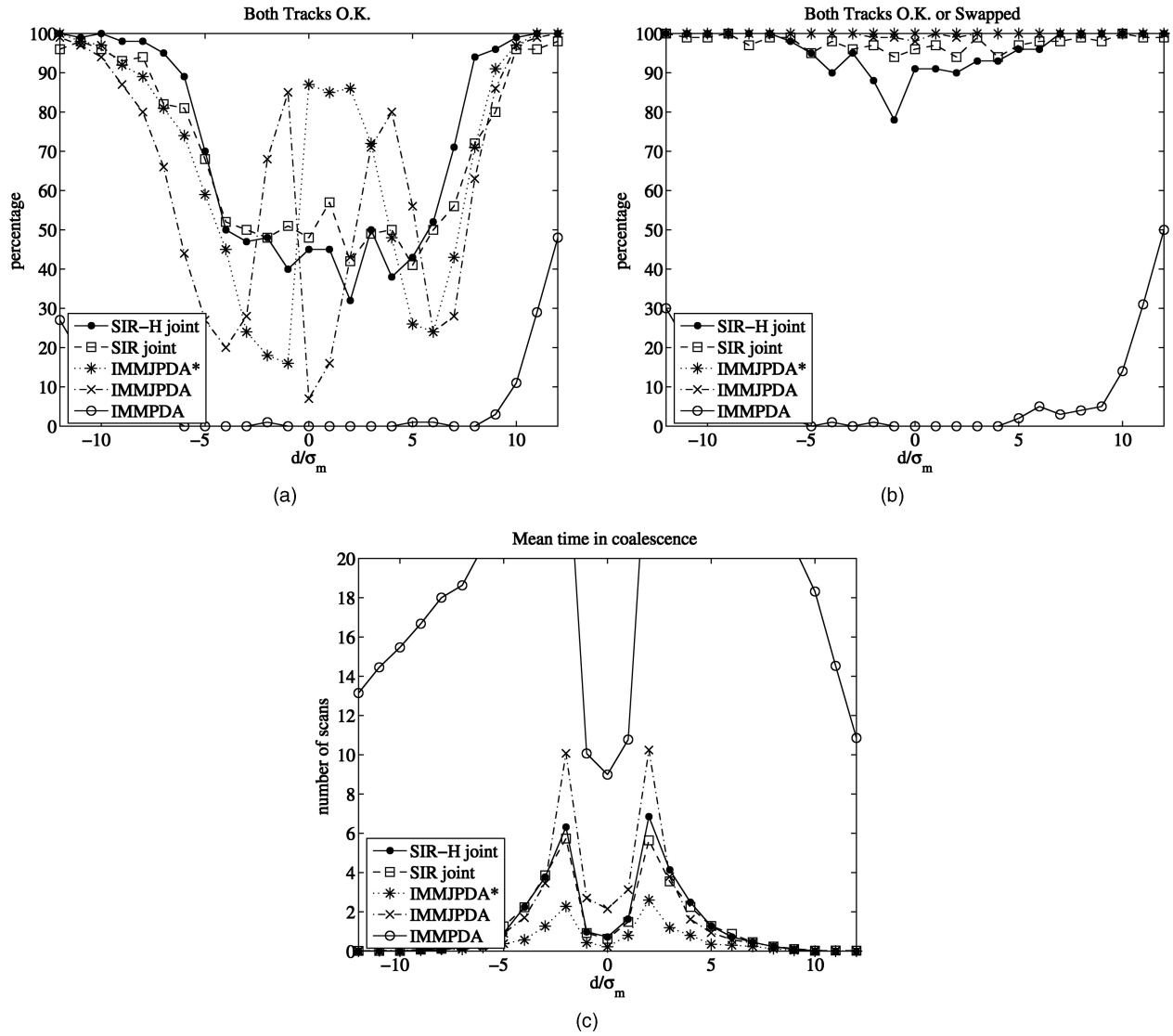


Fig. 4. Simulation results for scenario 3. (a) Both tracks "OK" percentage. (b) Both tracks "OK" or "Swapped" percentage. (c) Average number of "Coalescing" scans.

IMMJPDA* to joint tracking versions, i.e., to versions where the multi-target states/modes are jointly estimated. Tugnait [47] showed slightly improved simulation results for a particular example. In [12] we showed examples where the joint tracking versions performed better and examples where they performed worse. On average, the joint tracking versions even performed worse. In [14], [13] we showed that an appropriate pruning of permutation hypothesis also yields a track-coalescence-avoiding joint tracking version. The two weak points listed above for IMMJPDA and IMMJPDA* also apply to these joint versions.

Because the computational load of IMMJPDA* is one to two orders of magnitude lower than the computational load of the SIR-H joint particle filter is, this may be a fair reason to prefer IMMJPDA* over the SIR-H joint particle filter for particular applications. One should also be aware that the efficiency of the SIR-H joint particle filter can be significantly improved by incorporating various methods from literature (e.g. [1, 38, 42]).

In addition to the option of improving the efficiency of the SIR-H joint particle filtering, it is an option to improve the adaptation of the output equations. In this paper we considered the mean and covariance of target states only, and thus averaged over the states of all particles. One alternative approach might be trying to incorporate the permutation hypothesis pruning strategy of IMMJPDA* within the output equations of the SIR-H joint particle filter. Another direction [32] is to apply clustering of particles prior to averaging.

There are several other interesting extensions possible for the jump-linear descriptor framework and the novel exact and approximate filters. For example, to incorporate the target initiation and termination approach of [39], or to incorporate unresolved measurements (e.g. [31]).

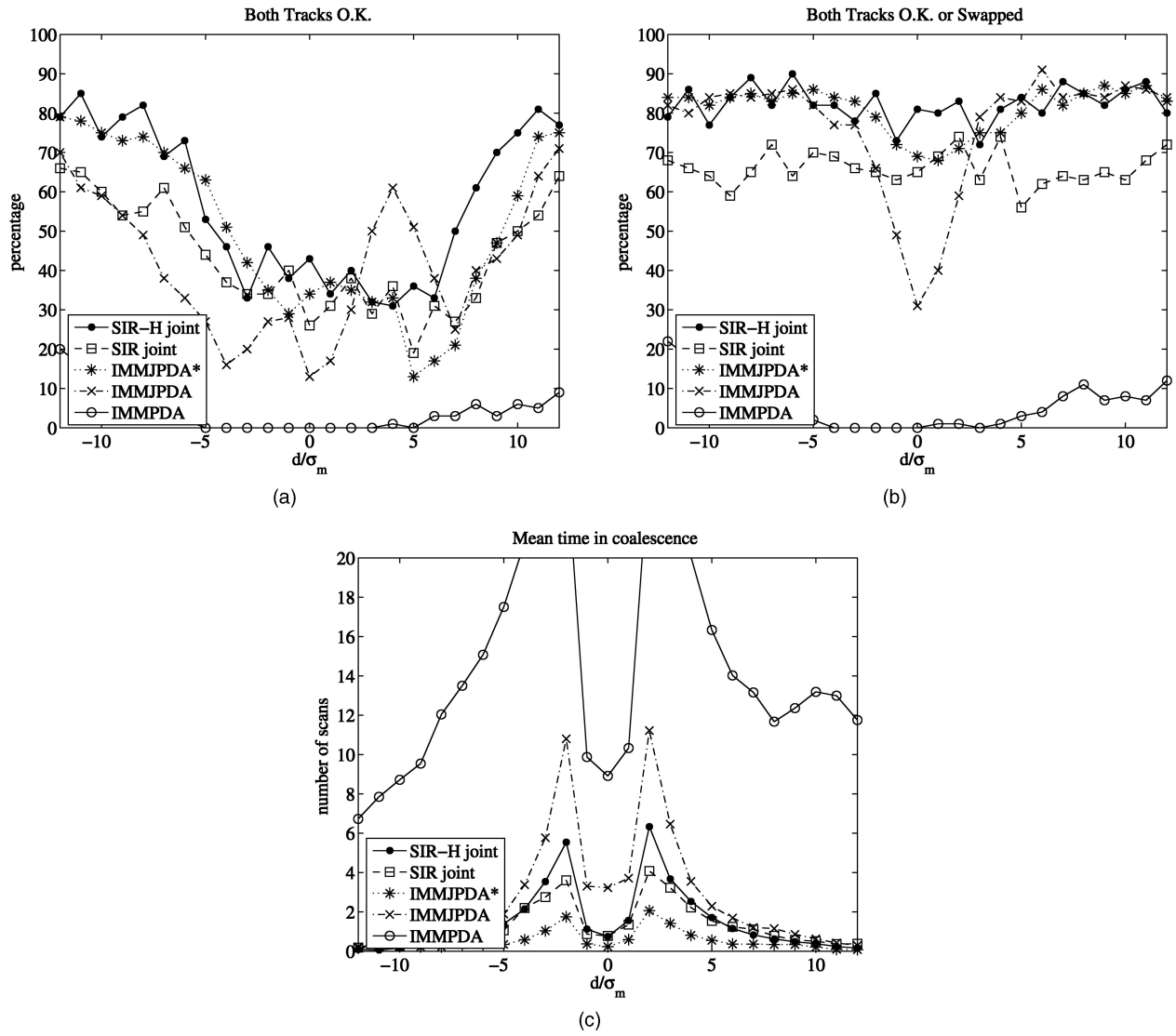


Fig. 5. Simulation results for scenario 4. (a) Both tracks "OK" percentage. (b) Both tracks "OK" or "Swapped" percentage. (c) Average number of "Coalescing" scans.

Appendix A

PROOF If $\phi = 0$ we get

$$p_{x_t|\theta_t,\phi_t,\tilde{\chi}_t,Y_t}(x \mid \theta, 0, \tilde{\chi}) = p_{x_t|\theta_t,Y_{t-1}}(x \mid \theta). \tag{A1}$$

Else, i.e., $\phi \neq 0$:

$$\begin{aligned} &p_{x_t|\theta_t,\phi_t,\tilde{\chi}_t,Y_t}(x \mid \theta, \phi, \tilde{\chi}) \\ &\quad = p_{x_t|\theta_t,\phi_t,\tilde{\chi}_t,\mathrm{y}_t,L_t,Y_{t-1}}(x \mid \theta, \phi, \tilde{\chi}, \mathrm{y}_t, L_t) \\ &\quad = p_{x_t|\theta_t,\phi_t,\tilde{\chi}_t,\mathrm{y}_t,L_t,\tilde{\mathrm{y}}_t,Y_{t-1}}(x \mid \theta, \phi, \tilde{\chi}, \mathrm{y}_t, L_t, \tilde{\boldsymbol{\chi}}\mathrm{y}_t) \\ &\quad = p_{x_t|\theta_t,\phi_t,\tilde{\mathrm{y}}_t,Y_{t-1}}(x \mid \theta, \phi, \tilde{\boldsymbol{\chi}}\mathrm{y}_t) \\ &\quad = p_{\tilde{z}_t|x_t,\theta_t,\phi_t}(\tilde{\boldsymbol{\chi}}\mathrm{y}_t \mid x, \theta, \phi) \cdot p_{x_t|\theta_t,Y_{t-1}}(x \mid \theta)/F_t(\phi, \tilde{\chi}, \theta) \end{aligned} \tag{A2}$$

with

$$F_t(\phi, \tilde{\chi}, \theta) \triangleq p_{\tilde{z}_t|\theta_t,\phi_t,Y_{t-1}}(\tilde{\boldsymbol{\chi}}\mathrm{y}_t \mid \theta, \phi). \tag{A3}$$

Subsequently

$$\begin{aligned} \beta_t(\phi, \tilde{\chi}, \theta) &\triangleq \mathrm{Prob}\{\phi_t = \phi, \tilde{\chi}_t = \tilde{\chi}, \theta_t = \theta \mid Y_t\} \\ &= p_{\phi_t,\tilde{\chi}_t,\theta_t|Y_t}(\phi, \tilde{\chi}, \theta) \\ &= p_{\phi_t,\tilde{\chi}_t\theta_t|\mathrm{y}_t,L_t,Y_{t-1}}(\phi, \tilde{\chi}, \theta \mid \mathrm{y}_t, L_t) \\ &= p_{\mathrm{y}_t,\tilde{\chi}_t,\theta_t|\phi_t,L_t,Y_{t-1}}(\mathrm{y}_t, \tilde{\chi}, \theta \mid \phi, L_t) \\ &\quad \cdot p_{\phi_t|L_t,Y_{t-1}}(\phi \mid L_t)/c'_t \\ &= p_{\mathrm{y}_t,\tilde{\chi}_t|\theta_t,\phi_t,L_t,Y_{t-1}}(\mathrm{y}_t, \tilde{\chi} \mid \theta, \phi, L_t) \\ &\quad \cdot p_{\phi_t|L_t,Y_{t-1}}(\phi \mid L_t) p_{\theta_t|Y_{t-1}}(\theta)/c'_t. \end{aligned} \tag{A4}$$

If $\phi \neq 0$, we have $D_t > 0$ and

$$\tilde{\chi}_t^T \tilde{\chi}_t = \Phi(\psi_t)^T \chi_t \chi_t^T \Phi(\psi_t) = \Phi(\psi_t)^T \Phi(\psi_t) = \mathrm{Diag}\{\psi_t\}. \tag{A5}$$

Hence

$$\psi_t = \mathrm{Diag}\{\psi_t\} 1_{L_t} = \tilde{\chi}_t^T \tilde{\chi} 1_{L_t}$$

with 1_{L_t} an L_t column vector with L_t 1-valued components.

Moreover, because

$$\tilde{\chi}_t \Phi(\psi_t)^T = \chi_t^T \Phi(\psi_t)\Phi(\psi_t)^T = \chi_t^T \tag{A6}$$

this shows that the transformation from (ψ_t, χ_t) into $\tilde{\chi}_t$ has an inverse. For the first term on the right hand side of (A.4) this implies:

$$\begin{aligned} &p_{\mathrm{y}_t,\tilde{\chi}_t|\theta_t,\phi_t,L_t,Y_{t-1}}(\mathrm{y}_t, \chi^T\Phi(\psi) \mid \theta,\phi,L_t) \\ &\quad = p_{\mathrm{y}_t,\psi_t,\chi_t|\theta_t,\phi_t,L_t,Y_{t-1}}(\mathrm{y}_t,\psi,\chi \mid \theta,\phi,L_t). \end{aligned} \tag{A7}$$

Furthermore, because the transformation from $(\mathrm{y}_t, \psi_t, \chi_t)$ into $(\tilde{z}_t, f_t, \psi_t, \chi_t)$ is a permutation, we get for $L_t > D(\phi) > 0$

$$\begin{aligned} &p_{\mathrm{y}_t,\psi_t,\chi_t|\theta_t,\phi_t,L_t,Y_{t-1}}(\mathrm{y}_t,\psi,\chi \mid \theta,\phi,L_t) \\ &\quad = p_{\tilde{z}_t,f_t,\psi_t,\chi_t|\theta_t,\phi_t,L_t,Y_{t-1}}(\boldsymbol{\chi}^T\boldsymbol{\Phi}(\psi)\mathrm{y}_t, \boldsymbol{\Phi}(1_{L_t}-\psi)\mathrm{y}_t,\psi,\chi \mid \theta,\phi,L_t). \end{aligned} \tag{A8}$$

Substituting (A8) in (A7) and this into (A4) yields:

$$\begin{aligned} &\beta_t(\phi,\chi^T\Phi(\psi),\theta) \\ &\quad = p_{\tilde{z}_t,f_t,\psi_t,\chi_t|\theta_t,\phi_t,L_t,Y_{t-1}}(\boldsymbol{\chi}^T\boldsymbol{\Phi}(\psi)\mathrm{y}_t, \boldsymbol{\Phi}(1_{L_t}-\psi)\mathrm{y}_t,\psi,\chi \mid \theta,\phi,L_t) \\ &\qquad \cdot p_{\phi_t|L_t,Y_{t-1}}(\phi \mid L_t) p_{\theta_t|Y_{t-1}}(\theta)/c_t'. \end{aligned} \tag{A9}$$

Hence, for $L_t > D(\phi) > 0$, this yields:

$$\begin{aligned} &\beta_t(\phi,\chi^T\Phi(\psi),\theta) \\ &\quad = p_{\tilde{z}_t|\theta_t,\phi_t,Y_{t-1}}(\boldsymbol{\chi}^T\boldsymbol{\Phi}(\psi)\mathrm{y}_t \mid \theta,\phi) \\ &\qquad \cdot p_{f_t|\phi_t,\psi_t,L_t}(\boldsymbol{\Phi}(1_{L_t}-\psi)\mathrm{y}_t \mid \phi,\psi) p_{\psi_t|\phi_t,L_t}(\psi \mid \phi) \\ &\qquad \cdot p_{\chi_t|\phi_t}(\chi \mid \phi) p_{L_t|\phi_t}(L_t \mid \phi) p_{\phi_t}(\phi) p_{\theta_t|Y_{t-1}}(\theta)/c''. \end{aligned} \tag{A10}$$

Evaluation of the terms in (A10) yields:

$$\begin{aligned} &p_{f_t|\phi_t,\psi_t,L_t}(\boldsymbol{\Phi}(1_{L_t}-\psi)\mathrm{y}_t \mid \phi,\psi) \\ &\quad = p_{f_t|F_t,\psi_t}(\boldsymbol{\Phi}(1_{L_t}-\psi)\mathrm{y}_t \mid L_t - D(\phi),\psi) \\ &\quad \overset{(6b)}{=} \prod_{i=1}^{L_t-D(\phi)} p_f([\Phi(1_{L_t}-\psi)\mathrm{y}_t]_i) \\ &\quad = \prod_{i=1}^{L_t-D(\phi)} p_f([\Phi(1_{L_t}-\tilde{\chi}^T\tilde{\chi}1_{L_t})\mathrm{y}_t]_i) \end{aligned} \tag{A11}$$

$$p_{\psi_t|\phi_t,L_t}(\psi \mid \phi,L_t) = D(\phi)!(L_t - D(\phi))!/L_t! \tag{A12}$$

$$p_{\chi_t|\phi_t}(\chi \mid \phi) = 1/D(\phi)! \tag{A13}$$

$$\begin{aligned} p_{L_t|\phi_t}(L_t \mid \phi) &= p_{F_t}(L_t - D(\phi)) \\ &= (\hat{F}_t)^{(L_t-D(\phi))} \exp\{-\hat{F}_t\}/(L_t - D(\phi))! \\ &\qquad \text{if} \quad L_t \geq D(\phi) \\ &= 0 \qquad \text{if} \quad L_t < D(\phi) \end{aligned} \tag{A14}$$

$$p_{\phi_t}(\phi) = \prod_{i=1}^{M} [(P_{\mathrm{d}}^i)^{\phi_i}(1-P_{\mathrm{d}}^i)^{1-\phi_i}]. \tag{A15}$$

Substituting (A3) and (A11) through (A15) into (A10) and subsequent evaluation yields for $L_t > D(\phi) > 0$:

$$\begin{aligned} \beta_t(\phi,\chi^T\Phi(\psi),\theta) &= F_t(\phi,\chi^T\Phi(\psi),\theta) \\ &\quad \cdot \hat{F}_t^{(L_t-D(\phi))} \cdot \prod_{j=1}^{L_t-D(\phi)} p_f([\Phi(1_{L_t}-\tilde{\chi}^T\tilde{\chi}1_{L_t})\mathrm{y}_t]_j) \\ &\quad \cdot \prod_{i=1}^{M} [(P_d^i)^{\phi_i}(1-P_d^i)^{(1-\phi_i)}] \cdot p_{\theta_t|Y_{t-1}}(\theta)/c_t \end{aligned}$$

with c_t a normalizing constant. It can be easily verified that the last equation also holds true if $L_t = D(\phi)$ or if $D(\phi) = 0$. Together with (6c) this yields (14).

Appendix B

PROOF From the proof of Proposition 1 we have

$$\begin{aligned} F_t(\phi,\tilde{\chi},\theta) &= p_{\tilde{z}_t|\theta_t,\phi_t}(\tilde{\boldsymbol{\chi}}\mathrm{y}_t \mid \theta,\phi) \\ &= \int_{\mathbb{R}^{Mn}} p_{\tilde{z}_t|x_t,\theta_t,\phi_t,Y_{t-1}}(\tilde{\boldsymbol{\chi}}\mathrm{y}_t \mid x,\theta,\phi) \\ &\qquad \cdot p_{x_t|\theta_t,\phi_t,Y_{t-1}}(x,\theta)dx \end{aligned} \tag{B1}$$

$$\begin{aligned} &p_{\tilde{z}_t|x_t,\theta_t,\phi_t}(\tilde{\boldsymbol{\chi}}\mathrm{y}_t \mid x,\theta,\phi) \\ &\quad = \prod_{\substack{i=1 \\ \phi^i=1}}^{M} p_{\tilde{z}_t^i|x_t^i,\theta_t^i}([\Phi(\phi)\tilde{\chi}]_{ik}\mathrm{y}_t^k \mid x^i,\theta^i). \end{aligned} \tag{B2}$$

This together with C2) yields:

$$F_t(\phi,\tilde{\chi},\theta) = \prod_{i=1}^{M} f_t^i(\phi,\tilde{\chi},\theta) \tag{B3}$$

with

$$\begin{aligned} f_t^i(\phi,\tilde{\chi},\theta) &= \int_{\mathbb{R}^n} p_{\tilde{z}_t^i|x_t^i,\theta_t^i}([\Phi(\phi)\tilde{\chi}]_{ik}\mathrm{y}_t^k \mid x^i,\theta^i) \\ &\qquad \cdot p_{x_t^i|\theta_t^i,Y_{t-1}}(x^i \mid \theta^i)dx^i \qquad \text{if} \quad \phi^i = 1 \\ &= 1 \qquad \text{if } \phi^i = 0. \end{aligned} \tag{B4}$$

Together with C3) the last two equations yield (29) and (30a,b,c).

Substitution of (B2) and C2) into (13) yields

$$\begin{aligned} &p_{x_t^i|\theta_t^i,\phi_t,\tilde{\chi}_t,Y_t}(x^i \mid \theta^i,\phi,\tilde{\chi}) \\ &\quad = \frac{p_{\tilde{z}_t^i|x_t^i,\theta_t^i}([\Phi(\phi)\tilde{\chi}]_{ik}\mathrm{y}_t^k \mid x^i,\theta^i) \cdot p_{x_t^i|\theta_t^i,Y_{t-1}}(x^i \mid \theta^i)}{f_t^i(\phi,\tilde{\chi},\theta)}. \end{aligned} \tag{B5}$$

If $p_{x_t^i|\theta_t^i,Y_{t-1}}(x^i \mid \theta^i)$ is Gaussian with mean $\bar{x}_t^i(\theta^i)$ and covariance $\bar{P}_t^i(\theta^i)$, then the density $p_{x_t^i|\phi_t,\tilde{\chi}_t,\theta_t^i,Y_t}(x^i \mid \phi,\tilde{\chi},\theta^i)$ is Gaussian with mean $\hat{x}_t^i(\phi,\tilde{\chi},\theta^i)$ and covariance $\hat{P}_t^i(\phi,\theta^i)$ satisfying for $\phi^i \neq 0$,

$$\begin{aligned} \hat{x}_t^i(\phi,\tilde{\chi},\theta^i) &= \bar{x}_t^i(\theta^i) + K_t^i(\phi,\theta^i)[[\tilde{\boldsymbol{\chi}}\mathrm{y}_t]_i - h^i(\theta^i)\bar{x}_t^i(\theta^i)] \\ \hat{P}_t^i(\phi,\theta^i) &= \bar{P}_t^i(\theta^i) - K_t^i(\phi,\theta^i)h^i(\theta^i)\bar{P}_t^i(\theta^i) \end{aligned}$$

and for $\phi^i = 0$:

$$\hat{x}_t^i(\phi, \tilde{\chi}, \theta^i) = \bar{x}_t^i(\theta^i)$$
$$\hat{P}_t^i(\phi, \theta^i) = \bar{P}_t^i(\theta^i)$$

Hence, $p_{x_t^i|\theta_t^i, Y_t}(. \mid \theta^i)$ is a Gaussian mixture, and all equations in Theorem 2 follow from a lengthy but straightforward evaluation of this mixture.

Acknowledgment

The authors would like to thank the anonymous reviewers for valuable suggestions in improving the paper.

REFERENCES

[1] M. S. Arulampam, S. Maskell, N. Gordon and J. Clapp
A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian trackers.
IEEE Transactions on Signal Procesing, **50** (2002), 174–188.

[2] D. Avitzour
Stochastic simulation Bayesian approach to multitarget tracking.
IEE Proceedings—Radar, Sonar and Navigation, **142** (1995), 41–44.

[3] Y. Bar-Shalom, S. Challa and H. A. P. Blom
IMM estimator versus optimal estimator for hybrid systems.
IEEE Transactions on Aerospace and Electronic Systems, **41** (2005), 986–991.

[4] Y. Bar-Shalom, K. C. Chang and H. A. P. Blom
Tracking splitting targets in clutter using an interacting multiple model joint probabilistic data association filter.
Multitarget Multisensor Tracking, Vol. II, Artech House, 1992, 93–110.

[5] Y. Bar-Shalom and R. Li
Multitarget-multisensor tracking: Principles and techniques. 1995.

[6] H. A. P. Blom
An efficient filter for abruptly changing systems.
In *Proceedings of 23th IEEE Conference on Decision and Control*, 1984, 656–658.

[7] H. A. P. Blom and Y. Bar-Shalom
The interacting multiple model algorithm for systems with Markovian switching coefficients.
IEEE Transactions on Automatic Control, **33** (1988), 780–783.

[8] H. A. P. Blom and E. A. Bloem
Probabilistic data association avoiding track coalescence.
IEEE Transactions on Automatic Control, **45** (2000), 247–259.

[9] H. A. P. Blom and E. A. Bloem
Combining IMM and JPDA for tracking multiple maneuvering targets in clutter.
In *Proceedings of 5th International Conference on Information Fusion*, Annapolis, MD, July 8–11, 2002, Vol. 1, 705–712.

[10] H. A. P. Blom and E. A. Bloem
Interacting multiple model joint probabilistic data association avoiding track coalescence.
In *Proceedings of IEEE Conference on Decision and Control*, Dec, 2002, 3408–3415.

[11] H. A. P. Blom and E. A. Bloem
Joint IMMPDA particle filter.
In *Proceedings of Fusion 2003*, Cairns, Australia, July 2003.

[12] H. A. P. Blom and E. A. Bloem
Tracking multiple maneuvering targets by joint combinations of IMM and PDA.
In *Proceedings of 42nd IEEE Conference on Decision and Control*, Maui, 2003.

[13] H. A. P. Blom and E. A. Bloem
Joint IMM and coupled PDA to track closely spaced targets and to avoid track coalescence.
In *Proceedings of Fusion 2004*, Stockholm, Sweden.

[14] H. A. P. Blom and E. A. Bloem
Exact Bayesian filter and joint IMM coupled PDA tracking of maneuvering targets from possibly missing and false measurements.
Automatica, **42** (2006), 127–135.

[15] Y. Boers and J. N. Driessen
Particle filter based detection for tracking.
In *Proceedings of American Control Conference*, June 2001, 4393–4397.

[16] Y. Boers and H. Driessen
Hybrid state estimation: A target tracking application.
Automatica, **38** (2002), 2153–2158.

[17] Y. Boers, J. N. Driessen, F. Verschure, W. P. M. H. Heemels and A. Juloski
A multi target track before detect application.
In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, June 2003.

[18] B. Chen and J. K. Tugnait
Tracking of multiple maneuvering targets in clutter using IMM/JPDA filtering and fixed-lag smoothing.
Automatica, **37** (2001), 239–249.

[19] L. Dai
Singular control systems.
Lecture notes in Control and Information Sciences, Vol. 118, Springer, 1989.

[20] M. De Feo, A. Graziano, R. Miglioli and A. Farina
IMMJPDA versus MHT and Kalman filter with NN correlation: Performance comparison.
IEE Proceedings—Radar, Sonar and Navigation, **144** (1997), 49–56.

[21] A. Doucet
On sequential simulation-based methods for Bayesian filtering.
Technical Report CUED/F-INFENG/TR.310, Department of Engineering, University of Cambridge, 1998.

[22] A. Doucet, N. J. Gordon and V. Krishnamurthy
Particle filters for state estimation of jump Markov linear systems.
IEEE Transactions on Signal Processing, **49** (2001), 613–624.

[23] R. J. Elliott, F. Dufour and D. D. Sworder
Exact hybrid filters in discrete time.
IEEE Transactions Automatic Control, **41** (1996), 1807–1810.

[24] R. J. Fitzgerald
Development of practical PDA logic for multitarget tracking by micro processor," In Y. Bar-Shalom (Ed.), *Multitarget-Multisensor Tracking*, Artech House, 1990, 1–23.

[25] M. Gauvrit
Bayesian adaptive filter for tracking with measurements of uncertain origin.
Automatica, **20** (1984), 217–224.

[26] N. J. Gordon
A hybrid bootstrap filter for target tracking in clutter.
IEEE Transactions on Aerospace and Electronic Systems, **33** (1997), 353–358.

[27] N. Gordon, D. Salmon and D. Fisher
Bayesian target tracking after group pattern distortion.
In *Proceedings of SPIE*, 1997, 238–248.

[28] N. J. Gordon, D. J. Salmond and A. F. M. Smith
Novel approach to nonlinear/non-Gaussian Bayesian state estimation.
IEE Proceedings, Part F, **140** (1993), 107–113.

[29] C. Hue, J. P. Le Cadre and P. Perez
Tracking multiple objects with particle filtering.
IEEE Transactions on Aerospace and Electronic Systems, **38** (2002), 791–811.

[30] R. Karlsson and F. Gustafsson
Monte Carlo data association for multiple target tracking.
In *Proceedings of IEE Seminar Target Tracking: Algorithms and Applications*, Oct. 2001, 13/1–13/5.

[31] W. Koch and G. Van Keuk
MHT maintenance with possibly unresolved measurements.
IEEE Transactions on Aerospace and Electronic Systems, **33** (1997), 883–892.

[32] C. Kreucher, K. Kastella and A. O. Hero
Tracking multiple targets using a particle filter representation of the joint multitarget probability density.
In *Proceedings of SPIE International Symposium on Optical Science and Technology*, San Diego, CA, Aug. 2003.

[33] J. MacCormick and A. Blake
A probabilistic exclusion principle for tracking multiple objects.
In *Proceedings of 7th International Conference on Computer Vision*, Greece, 1999, 572–580.

[34] J. MacCormick and M. Isard
Bramble: A Bayesian multiple-blob tracker.
In *Procedings of 8th International Conference on Computer Vision*, Vancouver, July 2001, 34–41.

[35] S. Maskell, M. Rollason, N. Gordon and D. Salmond
Efficient particle filtering for multiple target tracking with application to tracking in structured images.
In *Proceedings on Signal and Data Processing of Small Targets*, Orlando, FL, 2002, *SPIE*, Vol. 4728.

[36] S. McGinnity and G. W. Irwin
Multiple model bootstrap filter for maneuvering target tracking.
IEEE Transactions on Aerospace and Electronic Systems, **36** (2000), 1006–1012.

[37] S. McGinnity and G. W. Irwin
Maneuvering target tracking using a multiple-model bootstrap filter.
In A. Doucet, N. de Freitas and N. Gordon (Eds.), *Sequential Monte Carlo Methods in Practice*, Springer, 2001, 479–497.

[38] M. Morelande and S. Challa
Maneuvering target tracking in clutter using particle filters.
IEEE Transactions on Aerospace and Electronic Systems, **41** (2005), 252–270.

[39] D. Mušicki and R. Evans
Clutter map information for data association and track initiation.
IEEE Transactions on Aerospace and Electronic Systems, **40** (2004), 387–398

[40] M. Orton and W. Fitzgerald
A Bayesian approach to tracking multiple targets using sensor arrays and particle filters.
IEEE Transactions on Signal Processing, **50** (2002), 216–223.

[41] M. Orton and A. Marrs
A Bayesian approach to multi-target tracking and data fusion with out-of-sequence measurements.
In *Proceedings of IEE Seminar Target Tracking: Algorithms and Applications*, Oct. 2001, 15/1–15/5.

[42] B. Ristic, S. Arulampalam and N. Gordon
Beyond the Kalman Filter—Particle Filters for Tracking Applications.
Artech House, 2004.

[43] D. B. Rubin
Using the SIR Algorithm to Simulate Posterior Distributions.
Bayesian Statistics, vol. 3, Oxford University Press, 1988, 395–402.

[44] D. J. Salmon and H. Birch
A particle filter for track-before-detect.
In *Proceedings of American Control Conference*, June 2001, 3755–3760.

[45] D. J. Salmond, D. Fisher and N. J. Gordon
Tracking and identification for closely spaced objects in clutter.
In *Proceedings of European Control Conference*, 1997.

[46] D. Schulz, W. Burgard, D. Fox and A. B. Cremers
People tracking with mobile robots using sample-based joint probabilistic data association filters.
The International Journal of Robotics Research, **22** (2003), 99–116.

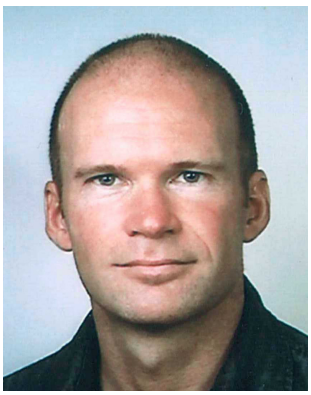
[47] J. K. Tugnait
Tracking of multiple maneuvering targets in clutter using multiple sensors, IMM and JPDA coupled filtering.
IEEE Transactions on Aerospace and Electronic Systems, **40** (2004), 320–330.

[48] J. Vermaak, S. J. Godsill and P. Perez
Monte Carlo filtering for multi-target tracking and data association.
IEEE Transactions on Aerospace and Electronic Systems, **41** (2005), 309–331.

Henk Blom was born in The Netherlands in 1953. In 1978, he received the M.Sc. degree in electrical engineering from Twente University.

In 1979, he performed research in forward-looking infra-red picture processing at TNO Physics Laboratory, The Hague. In 1980, he joined National Aerospace Laboratory NLR, Amsterdam. In 1988, he was visiting researcher at the University of Connecticut, Storrs, In 1990, he received the Ph.D. degree from Delft University of Technology on a dissertation entitled: "Bayesian Estimation for Decision-Directed Stochastic Control." His research focussed on stochastic hybrid systems with potential applications to air traffic management (ATM), and he gave scientific lead to the development of stochastic analysis based innovative systems in ATM, the most notable of which are ARTAS and TOPAZ. The former is EUROCONTROL's Bayesian multi-sensor multi-target tracking system, which incorporates automated data registration and contingency management. Since 1998, ARTAS has become operational by a steadily increasing number of air traffic control centres within Europe. TOPAZ is NLR's stochastic analysis based accident risk modelling system to provide safety/capacity feedback to advanced ATM designs. Since 2002, TOPAZ has proven its great value in providing safety feedback to the design and introduction of novel ATM operations at Amsterdam Airport. For these innovative system developments he received from NLR the Dr. Ir. B. M. Spee Award in 2004.

Dr. Blom is author or coauthor of over 50 conference papers, 10 journal papers and 10 book chapters, and is coeditor of a Lecture Note in Control and Information Science entitled: "Stochastic Hybrid Systems: Theory and Safety Critical Applications." He is member of joint FAA/Eurocontrol working groups to coordinate future ATM R&D on Separation Minima and on Safety respectively.



Edwin Bloem received the M.Sc. degree in systems and control theory from the University of Twente, Netherlands, in 1993.

Since 1993, he is research engineer at the National Aerospace Laboratory NLR. His work focuses on mathematical modelling for air transport safety and surveillance. His surveillance related work includes the development of advanced tracking algorithms and performance analysis. In the area of air transport safety he has developed extensive models of ATC technical systems and aircraft as well as models for controller and pilot behaviour and their interactions with technical systems. His research interests are in the areas of information fusion, estimation theory, control theory and safety.

Algorithms Fusion for Face Localization

R. BELAROUSSI

L. PREVOST

M. MILGRAM

Institute of Intelligent Systems and Robotics—PRC
University Pierre and Marie Curie

Face localization is a face detection problem where the number of people is known. We present a comparison between different algorithms fusion methods dedicated to the localization of faces in color images. Data to combine result from an appearance model supported by an auto-associative network, an ellipse model based on Generalized Hough Transform, and a skin color model. We introduce and compare several fusion methods like the Bayesian classifier with parametric or non-parametric technique, a fuzzy inference system, and a weighted average. Given an input image, we compute a kind of probability map on it using a sliding window. The face position is then determined as the location of the absolute maximum over this map. Improvement of basic detectors localization rates is clearly shown and prevalence of the weighted average is reported.

Manuscript submitted December 29, 2004; revised April 2, 2006.

Refereeing of this contribution was handled by Associate Editor Alexander Toet.

Authors' address: University Pierre and Marie Curie–Paris VI, LISIF-PARC, BC252, 4 place Jussieu 75252 Paris cedex 05 France, E-mail: (maurice.milgram {lionel.prevost}@upmc.fr).

1. INTRODUCTION

Face detection in an image has become a very important issue for many applications such as biometric, presence detection, video-conferencing, visiophony, indexation, car driver monitoring, virtual reality, lips reading, gaze tracking. Because of the high variability of the pattern to be detected, face detection without any hypothesis is a tough task [38]. Fixed camera and known background, use of motion information [6], strong hypothesis on the face location [20], scale or pose [33], special background for an easy extraction of the silhouette [24] or special lighting conditions (reflected infrared [9] or thermal infrared [11]): face detection applications often start with making assumptions. The face localization issue [4, 17, 20, 33] can be regarded as a face detection problem knowing the number of faces in the image. The location of the faces in the image—position and extent—is searched. The face localization issue is addressed in the present paper. It is not simpler without additional assumption.

A wide variety of works have been reported in face detection, much more than for face localization. Structural and holistic approaches, common in Pattern Recognition, are applied. Structural approaches try to detect facial landmarks (eyes, mouth, nose, head contour) and combines the results using models [3] or constellation analysis [2]. [3] built a generic model of the face through a joint distribution of parts (features models) positions. [12] brings a matching algorithm for pictorial structures (models of parts and connection between parts) applied to representation of an articulated human body. In [37] a hierarchical knowledge-based method finds face candidates at a low resolution and verifies presence of eyes and mouth at a high resolution. [39] uses deformable templates using a radiometrical model of eyes. For each facial feature, a statistical (GMM) model of Gabor filters responses is built in [17]. Features are detected over the whole image, then similarity with a constellation model is computed on a scanning-window, resulting in a coarse face localization. Then a cascade of two boosted SVM gives the accurate location of the face. In [2] component classifiers (SVM) are trained over selected parts of the face (bridge of the nose, nose, eyebrows, eyes, cheek, mouth): a constellations analysis performs face/non-face classification on a scanning window at several scales. [4] implements a similar approach at three scales with a skin/mouth color segmentation pre-processing. SVM are also used to model eyes and mouth in [33] at only one scale (corresponding to face's size).

Holistic approaches of face detection process a subimage of the input image into a feature vector (momentum, projection, gray level, wavelet…). These approaches estimate the classifier parameters on a training set, usually using a boosting procedure. Parameters can be weights of neural networks [16], [29], of weak classifiers [36] as described in Section 6.3 or terms of

a covariance matrix (statistical classifier) [34]. As in many detection issues, it is almost impossible to define the opposite class, the non-face patterns, which drives researchers to choose the model-based approach. A model does not require counter examples [13], which may seems an advantage but actually decreases classifier efficiency: generalization in a high dimension space (221 for 13×17 sub-images) is tough without knowing where are the patterns that might be confused. Another way is to design a combination of several detectors (classifiers). [13] and [14] did it to perform face detection. In [13] uses a conditional mixture of constraint generative models (Diabolo see Section 4) trained on different ranges of face orientation. Product and sum rules are used in [14] to combine two detectors based on edge orientation (edge orientation matching and Generalized Hough Transform) and one based on gray levels (Sparse Network of Winnows). Classifier combination has also been used in character [27, 28] and face recognition [7].

Our approach makes co-operate holistic and structural approaches: it is quite different but related to [14]. Generalization capability of a single classifier is limited, especially in a high dimension space. A more reliable decision can be obtained by combining output of several experts [27]: the face localization issue is divided in sub-problems easier to deal with. Various information is extracted from the same image using different kind of detectors. Some try to model global features while the others concentrate on structural features. Each face cues are searched by a relevant expert: elliptical shape, global appearance and skin color. Cooperation between experts exploits their complementarities and can also handle conflicts between sources.

An auto-associator network appearance based model and an ellipse detector are based on the image gradient's direction. A luminance-free skin color model is also implemented. The combination of these three detectors is done via various methods that are compared: Bayesian classifier, fuzzy inference system and neural networks.

Section 2 describes the skin color model, Section 3 details the ellipse model, and Section 4 deals with the appearance based model. Several combination strategies are presented in Section 5. Comparison of the combinations is detailed in Section 6 along with our experimental results and the contribution of the combination to the face localization problem. The last section is devoted to conclusions and prospects.

2. SKIN COLOR MODEL

Skin color classification aims at determining whether a color pixel has the color of flesh or not. Such a classification should overcome difficulties like different skin tones (white, pink, yellow, brown, black…) and scene illuminations, and the fact that background pixels can have the same color as a flesh type.

2.1. Color Spaces Definition

Two color spaces are investigated for skin color classification: HSV and YCbCr. These spaces are commonly used [26] in image processing for they are expected to be more robust to lighting condition by separating chrominance (color information) and luminance (grayscale levels) information. In a video signal, color images encoding separates the luminance and chrominance information: this way television standards (NSTC, Pal, Secam) ensured backward compatibility with black and white television. Chrominance is the color information to be added to the grayscale information to obtain a color image in red, green and blue primary colors. Chrominance information is widely used for skin color classification as it is expected to be a common cue between different skin tones contrarly to the luminance. Skin color classifiers based on chrominance tend to be more robust to different lighting conditions.

RGB conversion to YCbCr is linear (see (1))

$$\begin{aligned} \mathrm{Y} &= 0.299\mathrm{R} + 0.587\mathrm{G} + 0.114\mathrm{B} \\ \mathrm{Cb} &= 0.564(\mathrm{B} - \mathrm{Y}) + 128 \\ \mathrm{Cr} &= 0.713(\mathrm{R} - \mathrm{Y}) + 128. \end{aligned} \tag{1}$$

Y channel is the luminance, Cb and Cr channels represent chrominance. We used the definition of [19], it uses an RGB model that fits the phosphor emission characteristics of older cathode ray tubes. Y, Cb and Cr values range from 0 to 255. Variants of this definition that fit the phosphor emission characteristics of newer tubes and other modern display equipment can be found. YPbPr, YUV, YIQ are same or similar color spaces.

HSV space is a non-linear transformation of RGB space (see (2)): colors are defined by hue (H channel), saturation (S channel) and luminance (V channel)

$$\mathrm{V} = \max(\mathrm{R},\mathrm{G},\mathrm{B}), \qquad \mathrm{S} = 255\frac{\mathrm{V} - \min(\mathrm{R},\mathrm{G},\mathrm{B})}{V}$$

$$\mathrm{H} = \begin{cases} 30\dfrac{\mathrm{G}-\mathrm{B}}{\mathrm{S}} & \text{if} \quad \mathrm{V} = \mathrm{R} \\ 30\dfrac{\mathrm{B}-\mathrm{R}}{\mathrm{S}} + 90 & \text{if} \quad \mathrm{V} = \mathrm{G} \\ 30\dfrac{\mathrm{R}-\mathrm{G}}{\mathrm{S}} + 120 & \text{if} \quad \mathrm{V} = \mathrm{B} \end{cases}. \tag{2}$$

S ranges from 0 to 255, and represents the grayness of the color: the lower the saturation of a color is the more faded it appears (a monochrome color corresponds to S = 0). H values are defined modulo 180 from red (H = 0) through yellow, green, cyan, blue, and magenta, and returns to red (H = 180). Similar color spaces include HSB, HLS, and HIS.

2.2. Skin Color Pixels Classification

A recent comparison of different skin color classification algorithms can be found in [26]. Linear [1, 5,

TABLE I
Training and Validation Sets for Skin Segmentation

Dataset	No. Images	Skin Pixels	Non-Skin Pixels
Training	500	18.2 million	120.9 million
Test	550	23 million	136.6 million

TABLE II
Confusion Matrix of the CbCr Fixed Range Skin Classifier

Classification Class	Skin	Non-Skin
Skin	77%	23%
Non-skin	17%	83%

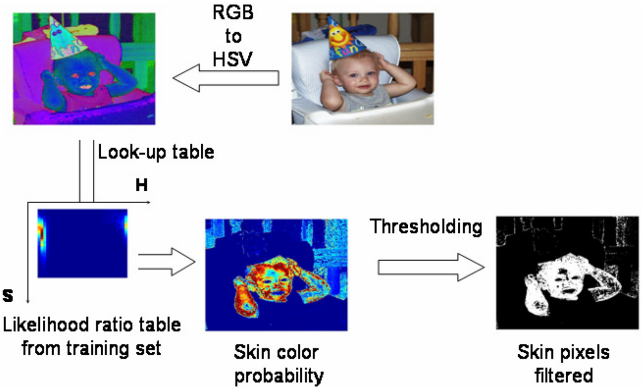


Fig. 1. Back project of the histogram ratio.

6] and Bayesian classifiers [21, 26] are proposed and compared in the present paper.

1050 images of the ECU database described in Section 6.1 are used for training and assessment of skin segmentation methods presented in the following subsection.

Repartition of the two sets is summarized in Table I. These images are not used for face localization tests (Section 6.2 and 6.3).

2.2.1. Rectangular Boundary in CbCr Plane

Linear classification uses a piecewise linear decision boundary in the Cb-Cr plane. The following fixed-range in Cb and in Cr is used to define skin color pixels:

$$\mathrm{Cb} \in [100\ \ 130] \qquad \text{and} \qquad \mathrm{Cr} \in [135\ \ 165].$$

These thresholds were experimentally tuned using images with people. Skin being characterized by specific chrominance information, the filter can be applied to any ethnic skin color but our threshold is not universal because the chrominance component is actually related to the luminance value Y [18]. In poor or bright illumination condition the filtered components are spurious and in some cases no skin at all is filtered: this skin detector is coarse but simple and we use it as a reference for comparison with other skin classifiers.

This classifier results in a one point ROC curve (see Fig. 3): Table II is the confusion matrix obtained over the validation set.

2.2.2. Statistical Classifiers

The Bayesian decision rule is a popular method in statistical pattern classification [10]. A color pixel $\vec{X}$ is classified as a skin pixel if its likelihood ratio is higher than a threshold:

$$\frac{P(\vec{X} \mid \text{skin})}{P(\vec{X} \mid \text{non-skin})} \geq \tau \tag{3}$$

$P(\vec{X} \mid \text{skin})$ and $P(\vec{X} \mid \text{non-skin})$ are the conditional probability density functions (denoted pdf in this paper) of respectively skin and non-skin color. τ is the decision threshold. A given τ value results in a confusion matrix: ROC curve of the classifier is obtained by varying the threshold τ.

The computation of the pdfs is done using the histogram technique. In [31] face color is tracked using this technique. Statistical repartition of skin pixels in HS plane (or CbCr plane) is calculated in a 2D histogram. Scaling the histogram results in $P(\vec{X} \mid \text{skin})$. Same operation is done with non-skin pixels to evaluate $P(\vec{X} \mid \text{non-skin})$.

Ratio of the two histogram results in a likelihood ratio table [31]: skin probability of a color pixel featured by (H, S) values is then computed by look-up table. Back projecting the histogram ratio onto the HSV (or YCbCr) image results in a skin color probability image as shown in Fig. 1.

H and S channels (respectively Cb and Cr channels) feed the 2D skin and non skin histograms. 32 bins per channel are allocated. [21] found that 32 bins are optimal whereas [26] concludes that larger histogram leads to finer pdfs estimation and better performances when training samples are sufficient. As explained in [26], when training set is not large enough, a larger histogram results in a noisier pdf compared to a smaller histogram size. Subsampling their original training set, they found that the 256-bin histogram is more sensitive to the number of training samples compared to the 32-bin histogram. And even with a huge sample number the larger histogram is just a few percent more efficient than a 32-bin histogram, justifying our choice.

In Fig. 2 it appears that skin and non-skin pixels are pretty well-separated in the HS plane. On the opposite, skin and non-skin pixel distributions in CbCr plane are clearly overlapping. Therefore model the non-skin distribution brings poor improvement in HS plane and dramatically increases classification performance in CbCr plane as shown by the ROC curves plotted in Fig. 3.

Assuming the non-skin pixel distribution is uniform, the decision rule in (3) is simplified. A color pixel is classified as skin color if

$$P(\vec{X} \mid \text{skin}) \geq \tau. \tag{4}$$

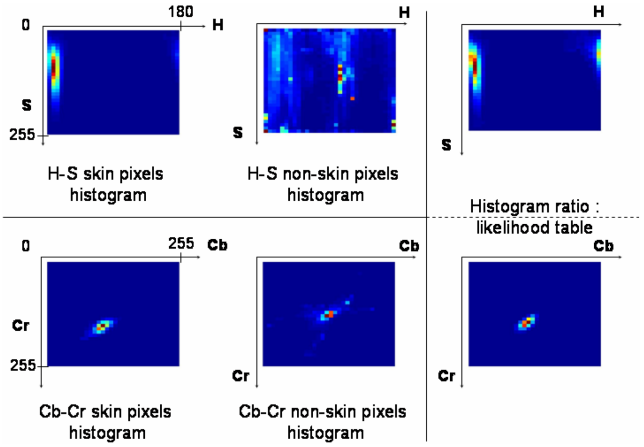


Fig. 2. Skin, non-skin and ratio histogram in HS plane (first line) and CbCr plane (last line): hot colors correspond to high values.

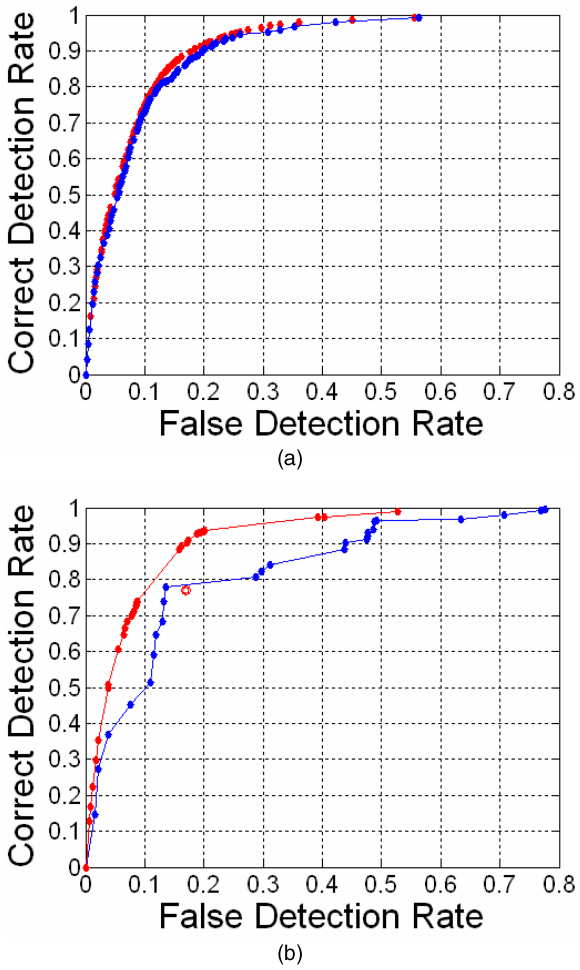


Fig. 3. ROC of skin color classifier in HS space (a) and CbCr space (b).

A classifier based on (4) uses the statistical repartition of skin pixels regardless of non-skin pixels distribution.

Classification performance is represented with the Receiver Operating Characteristic (ROC) curve: skin segmentation performance for a given decision threshold τ is measured in terms of correct detection rate and false detection rate. Correct detection rate is the proportion of skin pixels correctly classified whereas the false detection rate is the proportion of non-skin pixels classified as skin pixels. The ROC curve is obtained by calculating these rates for all coherent τ values.

ROC curve of the classification based on Cb-Cr statistical models of skin and non skin (decision rule (3)) color pixels is plotted in red in Fig. 3(b). Skin classifier based on statistical repartition of skin color pixels (decision rule (4)) in the Cb-Cr plane is plotted in blue in the same figure. The CbCr fixed range classifier ROC point is plotted in red.

ROC curve of CbCr skin model is highly irregular whereas classification that use the likelihood ratio is quite satisfactory compared to state of the art reported by [26]. In [26] the best classification performance is obtained by a Bayesian classifier (decision rule (3)) in the RGB space with the histogram technique: for a false detection rate of 10% a correct detection rate of 82% is reached whereas our classifier correct detection rate is 75% for the same false detection rate. Moreover it appears that modeling the non-skin distribution in the CbCr space is crucial: a classifier only based on the statistical repartition of skin pixels is not really efficient with a correct decision rate of 50% for 10% of false detection.

In Fig. 3(a), ROC curve of the classifier that models both skin and non-skin distributions in the H-S plane is plotted in red. ROC curve of the classifier modeling only the skin distribution in the H-S plane is plotted in blue.

Modeling non-skin distribution in the HS plane only brings a slight improvement of skin color classification performance compared to a classification based on the skin distribution alone. Moreover, modeling the non-skin distribution is not a satisfying approach as non-skin color cannot be defined: such a distribution completely depends on the non-skin training database. ROC of the likelihood ratio classifier in Cb-Cr plane is a bit better than ROC of the classifier based on skin color repartition in H-S plane but it is also more irregular and requires to compute the non-skin pixel distribution in the Cb-Cr plane.

Therefore, we selected the Bayesian classifier in HS space based on the skin color repartition as our skin detector for the multi-scale segmentation of the face in Section 6.3.

The CbCr fixed range classifier is also used for combiners comparison presented in Section 6.2 for its simplicity.

2.3. Skin Detector

For combination purpose (Section 5) each sub-window of the original image is featured by a single value. A retinal approach is implemented after the skin color pixels classification stage. A sliding window of fixed size (13×17) scans the skin filtered image and

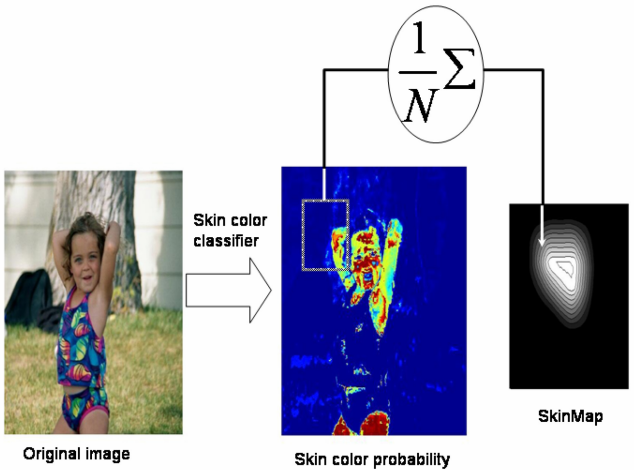


Fig. 4. SkinMap: proportion of skin pixels array.

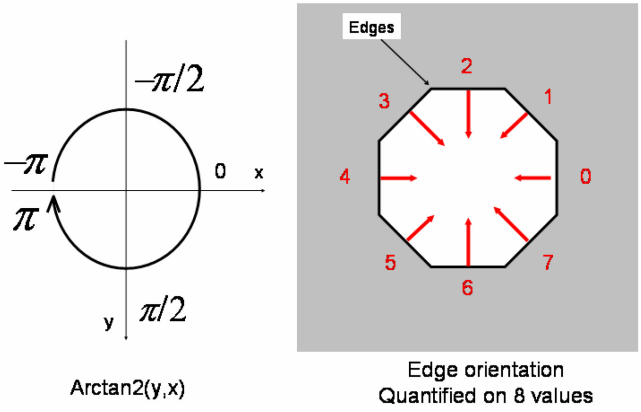


Fig. 5. Four-quadrant inverse tangent and quantification of edge orientation.

calculates the mean skin pixels probability at every position [32] as shown in Fig. 4.

The resulting array is named "SkinMap" and represents the face sub-image probability.

3. ELLIPSE DETECTOR BASED ON GENERALIZED HOUGH TRANSFORM

3.1. Edge Orientation Field

Edge orientation information is processed by an appearance-based model (so called Diabolo see Section 4) and an ellipse detector (Generalized Hough Transform).

Evaluation of the orientation of the gradient on the edges requires a low pass filtering of the image: see Fig. 5. Gradient field is estimated using Roberts masks (2×2), so that horizontal gradient is calculated by $I_x = I_{\text{filtered}} \otimes [1 \ \ -1]$ and vertical gradient with $I_y = I_{\text{filtered}} \otimes \begin{bmatrix} 1 \\ -1 \end{bmatrix}$.

Then the gradient magnitude $= \sqrt{I_x^2 + I_y^2}$ is threshold to define edge pixels. For the generalized Hough transform, a global threshold is applied over the whole input image. Orientations of these edge pixels are then

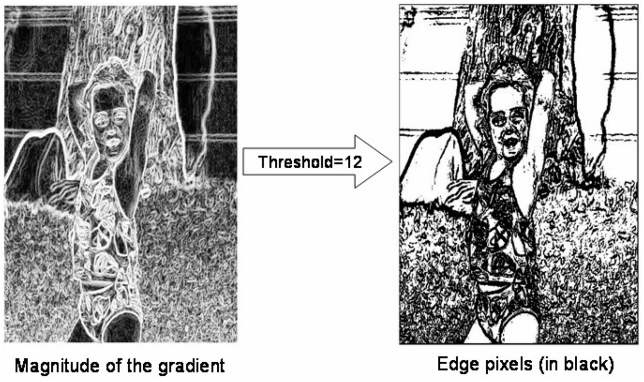


Fig. 6. Threshold of magnitude field defines edge pixels.

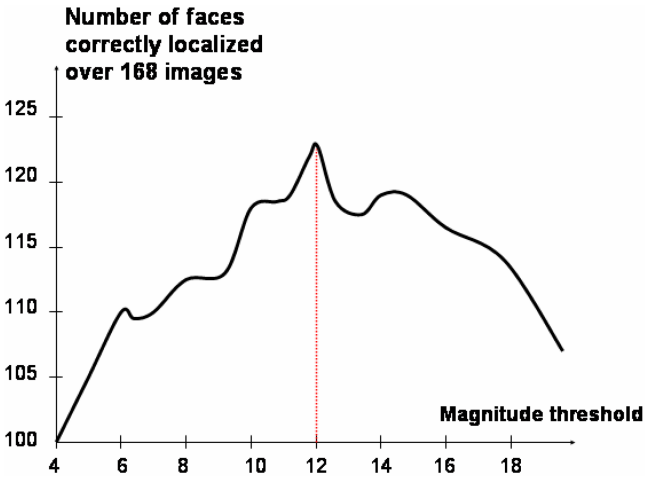


Fig. 7. Magnitude threshold to define edge is tuned to maximize GHT performance over 168 images.

quantized on $N = 36$ values:

$$\text{orien} = \text{round}\left(\frac{N}{2\pi}\arctan 2(I_y, -I_x)\right) \mod N \quad (5)$$

where arctan 2 is the four-quadrant inverse tangent. This function is depicted in Fig. 5 with an ellipse's edge orientation quantified on $N = 8$ values (Freeman chain code) using (5).

For the Generalized Hough Transform edges are defined as pixels with a magnitude greater than a threshold equal to 12: see Fig. 6.

This threshold was tuned over 168 training images containing only one person. This training corpus is not overlapping with the test set used in Section 6. For a given image, a Generalized Hough Transform is computed (see next section) and the maximum of the accumulator is defined as the location of the face. Using ground truth we evaluate the number of faces correctly localized versus threshold, see Fig. 7.

3.2. Ellipse Detector Based on Generalized Hough Transform

The elliptical shape of the face is searched using a Generalized Hough Transform: faces are modeled as vertical ellipses with a specific eccentricity.

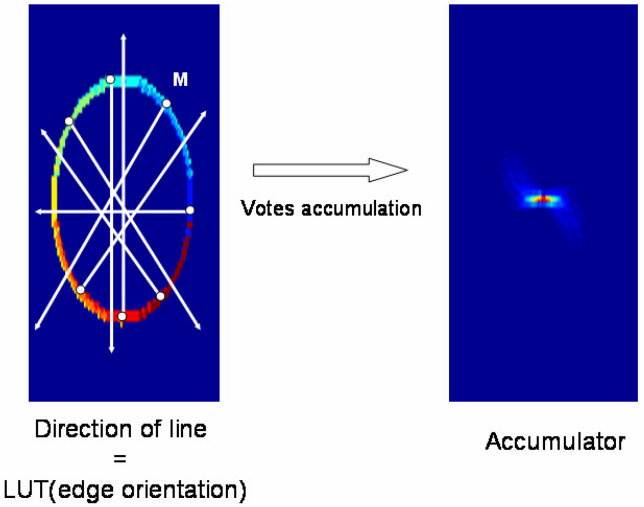


Fig. 8. Generalized Hough Transform in case of an ellipse: half-line votes accumulation.

The gray level dynamic of the input image is first linearly adjusted between 0 and 255. This operation proved to be better than performing histogram equalization. Orientation of the gradient over the whole gray level image is then determined. Then a Generalized Hough Transform (GHT) is performed on the resulting orientation map: the HT constitutes a popular method for extracting geometrical properties [10, 32]. When the edge orientation is used and when it is applied to non parametric curves, the HT becomes the Generalized HT. Each edge pixel votes for all possible location of the shape (actually for the location of the barycentre). For ellipse detection, there is a simplified structure for the GHT based on the geometrical properties of ellipses.

The method consists in casting votes for a half-line starting at each boundary pixel M with an orientation determined by the edge one. The method consists in casting votes for a line through each boundary pixel with an orientation indexed in a look-up table by the edge orientation. We suppose that we know the orientation of the ellipse. So for each point M, a simple look-up table specifies the angle between the tangent Mt (to the boundary) and the radius MO (O is the centre of an ellipse passing through M). Faces are modeled as vertical ellipses with a specific eccentricity so we can build up our look-up table to cast votes from each edge pixel, knowing its gradient orientation. Fig. 8 illustrates an ellipse case: some half-lines are drawn. Each pixel of a line increment a vote array which is the accumulation of all lines votes.

Accumulator maximum corresponds in the image to the position most likely to be the center of an upright ellipse with a horizontal minor axis $a = 8$, and a vertical major axis $b = 10$. Fig. 9 illustrates an example of such an accumulation by Generalized Hough Transform in case of a cluttered scene. Finally, the accumulator is scanned with a 13×17 sliding window and at each position a weighted average of the number of vote is

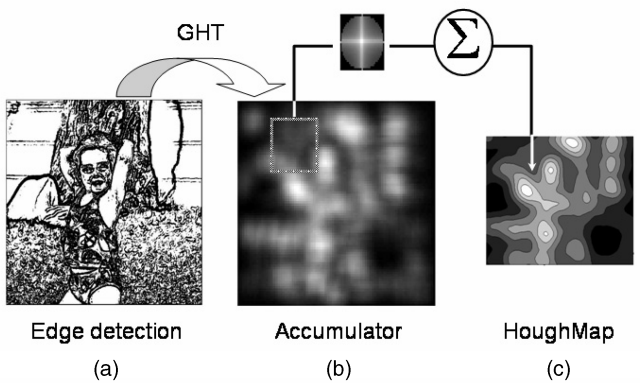


Fig. 9. Edge detection (a), Generalized Hough Transform accumulator (b) computed over gradient orientation of the edge and resulting HoughMap (c).

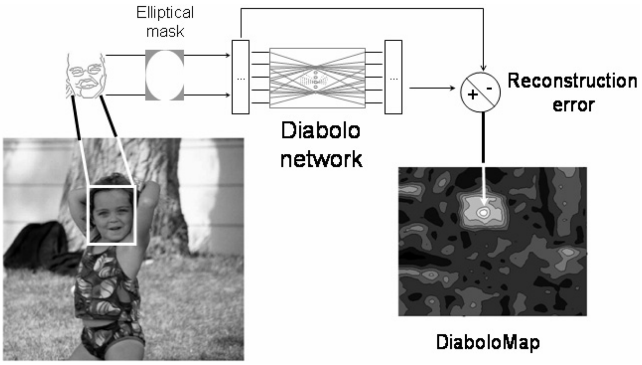


Fig. 10. DiaboloMap: array of reconstruction errors calculated at all positions of the image.

calculated as shown in Fig. 9: the resulting array is named "HoughMap."

4. APPEARANCE-BASED MODEL OF THE FACE

The Diabolo is an auto-associator network: its number of output equals its number of input. It is trained to reconstruct an output identical to its input, and only face examples constitute the training database. It implements a specialized compression for its hidden layer has much less units than input or output does. So a non-face image should be badly compressed and the reconstruction error (square root of the mean square error between the input and the calculated output) would be higher than for a face image. The Diabolo was successfully used for handwritten character recognition [30], face detection [13] and compression [8].

As represented in Fig. 10, reconstruction error is computed on a fixed size window sliding over the entire image. The resulting array is called "DiaboloMap": clear color correspond to small reconstruction error.

Diabolos we implemented have one hidden layer. Hidden neurons have sigmoid activation function, and output neurons have linear activation function, see Fig. 11. The training set is made of face images (see Table III). Faces are various in terms of pose, lighting conditions and skin tones.

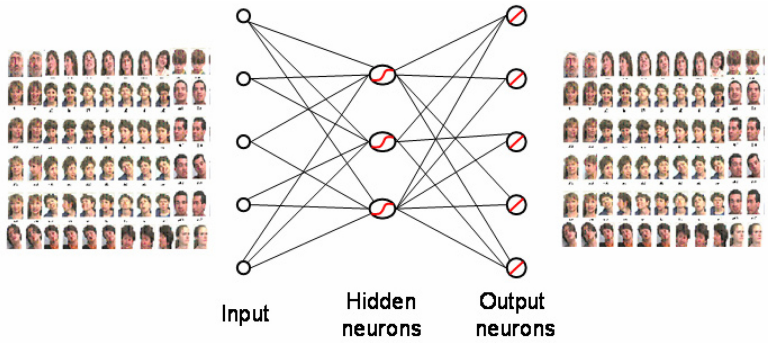


Fig. 11. Architecture of a Diabolo: target is equal to input, training set is made of faces examples.

TABLE III
Training and Cross-Validation Sets of Diabolos

Set	Training	Cross-Validation
No face images	1602	178

The training database is divided into two sets: one for neural networks training, one for the cross-validation and assessment of the best architecture. The cross-validation face samples are extracted from 167 images: amongst these images, 126 images contain only one person and are used as an assessment set to optimize the inputs coding.

Face examples are used to learn parameters (weights) of the neural networks. Training is done using a gradient descent with adaptative learning rate stopped by cross-validation. Gradient descent algorithm is a standard backpropagation in which the network weights are moved along the negative of the gradient of the cost function. The cost function implemented here is the sum over training examples of the square reconstruction error between target and simulation (output calculated by the MLP).

Networks are trained for pattern model: target is equal to the input. Before training the MLP weights must be initialized: a different initialization leads to different weights, therefore to different networks. For a given neural net architecture, several initializations must be tested in order to avoid the network to fall in a local minimum of the performance function different to its global minima.

The Diabolo is fed with a specific coding of edges orientation. Gradient field orientation is quantized on $N = 36$ values as defined in Section 3.1 by (5). Edges are defined by a local magnitude threshold depending on the search sub-window. The threshold is defined over each 13×17 sub-windows of the input image, so that 20% of the pixels are then regarded as edge: an example is given in Fig. 12.

A global threshold over the whole image (face + background) would result in a strong smoothing of the face. A local threshold keeps facial features visible when the sear window is over the face, but it also emphasizes edges over non-face subwindow, which results in lot of false alarms if the face location is defined

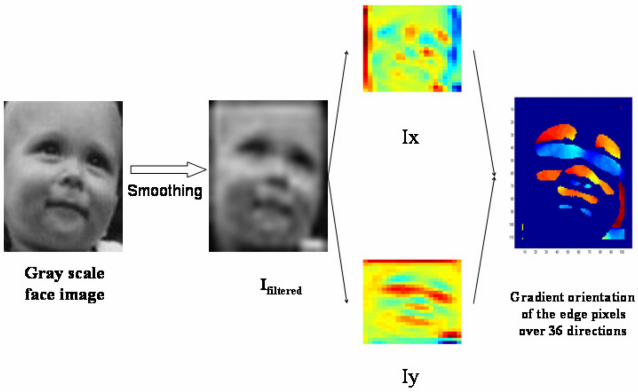


Fig. 12. Estimation of gradient field and edges orientation.

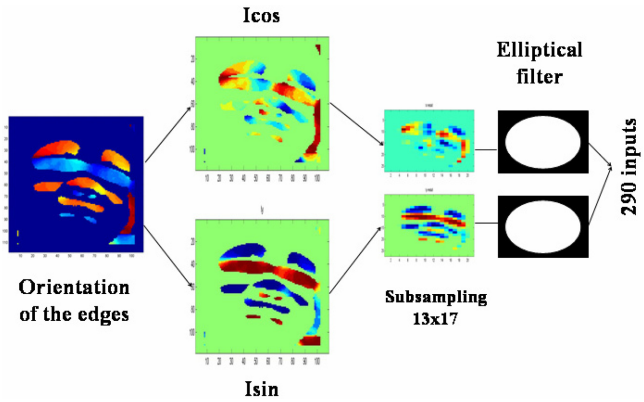


Fig. 13. Training example pre-processing.

as location of the smallest reconstruction error of the whole image (minimum of DiaboloMap).

Each pixel is described by two features $(I_{\cos}, I_{\sin})$: $I_{\cos}(i,j) = \cos((2\pi/N) \cdot \text{orien}(i,j))$ and $I_{\sin}(i,j) = \sin((2\pi/N) \cdot \text{orien}(i,j))$ for the edge pixels, where orien refer to (5); (0,0) is allocated to the non-edge pixels. An elliptical mask filters the interior part of the face as shown in Fig. 13.

We compared that coding with two others: feeding the Diabolo with grayscale face image or with the gradient field as illustrated in Fig. 14.

For the three selected input coding the best Diabolo architecture (i.e. optimal number of hidden neuron) correspond to the best face localization rate. Face localiza-

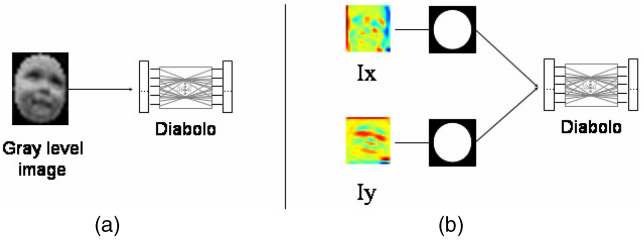


Fig. 14. Other input coding: grayscale image (a) and gradient field (b).

TABLE IV
Performances Versus Input Coding

Input Coding	Gray Levels	Gradient Field	Orientations Coding
Localization rate	27%	18%	40%

tion rate is evaluated over an assessment set: 126 images from which the cross-validation set was extracted that contain only one face. For each image a DiaboloMap is built as in Fig. 10 and face location is defined as the position of the minimum (smaller reconstruction error over the whole image). Face localization performance of the Diabolo fed with the three kind of inputs are given in Table IV: gradient orientation coding reaches the higher localization rate (40%), followed by the gray level coding of inputs (27%) and the gradient field (18%).

This comparison was done using a 21×27 retina to build the DiaboloMaps. We also investigated 17×22 and 13×17 retina: for the selected pre-processing of the training examples (see Fig. 13) the optimal retina size is 13×17. It is the best size for face localization purpose and also for computational effort. Finally the optimal Diabolo architecture is made of 290 inputs and outputs, and 18 hidden neurons. Note the dimension reduction from 442 ($2 \times 13 \times 17$ elements in $I_{\cos}$ and $I_{\sin}$) to 290 due to the elliptical filtering of inputs.

Interior part of faces is used to train the network using an elliptical mask to reduce border effects and in order not to model the elliptical shape of the face. The Diabolo is trained to model facial features: mouth and eyes, mainly. This approach is different from a face detector based on neural network which takes the face contour into account: this enhances the face detection rate. Our approach aims at compute face contour and facial features separately. This way, redundant information between the appearance-based model and the ellipse model are reduced.

5. COMBINATION OF THE SOURCES FOR FACE LOCALIZATION PURPOSE

5.1. Overview of the Combination Approach

We have implemented three holistic detectors for a color image, which result in three maps: DiaboloMap, HoughMap, and SkinMap. When each detector alone failed to model facial features, the combination of the three sources can achieve this task very well. The combination can also handle conflicts between sources.

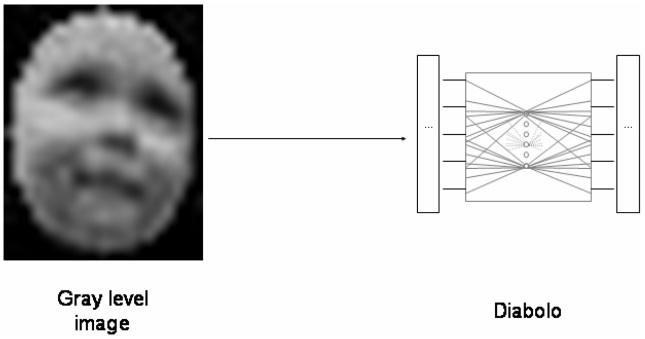


Fig. 15. Overview of the face localization system.

TABLE V
Training Set of Combiners

Class	Face	Non face
No. Samples	19.579	482.783

For that purpose, each detector map is linearly adjusted onto $[-1\ 1]$. Using the three detectors, a search window at position (i, j) in the original image is then featured by $I_{i,j}$ = [H D S].

Several architectures exist for data fusion, we can divide them into three kinds: serial (or sequential), parallel and hybrid (mixing sequential and parallel, with feed-back or interaction…). Our face localization system has a parallel architecture (see Fig. 15).

Combination rules are various, depending on the application: mean, weighted sum, product or maximum of experts outputs, majority vote, fuzzy rules, neural networks, or neuro-fuzzy inference for example.

Several algorithms have been proposed for combining our three detectors: parametric and non-parametric combination strategies are described in this sub-section. The next section is dedicated to their comparison.

Table V summarizes the number of face and non-face samples used for training combiners: these data were extracted from the cross-validation images used to stop Diabolo training.

5.2. Bayesian Classifier: Parametric and Non-Parametric Approaches

For combination purpose the input data of the Bayesian classifier are the normalized response of our three detectors. A sub-image featured by a 3D vector $\vec{X}$ = [H D S] is classified as a face if

$$\frac{P(\vec{X} \mid \text{face})}{P(\vec{X} \mid \text{non-face})} \geq \tau \tag{6}$$

where $P(\vec{X} \mid \text{face})$ and $P(\vec{X} \mid \text{non-face})$ are respectively the conditional probability density function (pdf) of the

face and non-face class. τ is the decision threshold usually estimated over a training set. As the application presented in this paper is face localization and not face detection, no estimation of τ was done. The face location shall be the one that maximize the value of the likelihood ratio (left hand side of (6)):

$$\text{Face location} \leftrightarrow \max\left(\frac{P(\vec{X} \mid \text{face})}{P(\vec{X} \mid \text{non-face})}\right) \qquad (7)$$

Parametric and non-parametric estimations of the class-conditional pdf are implemented.

The histogram technique is a non-parametric method. For each class a 3D histogram is computed using the training examples. Due to the small amount of face examples, it has only five bins per dimension: 5^3 being equal to 125, a mean of about 160 examples per bin is available. We combine the two histograms obtained into one histogram which bins values are the ratio of the bins frequency of the two preceding histograms (face/non-face). Resulting histogram values are then scaled into [0 255]. When a test image is processed three maps are calculated corresponding to our three face models over a sliding search window at each position of the image (see Fig. 15). For each position of the test image a 3D vector is computed and a back-projection of the histogram is done by a look-up table operation. This back-project is the FusionMap illustrated in Fig. 15, face location should correspond to the position of its maximum value.

A parametric approach models both skin and non skin class-conditional pdf by a unimodal Gaussians. The face location is then defined as the position of the maximum of the logarithm of the likelihood ratio:

$$(\vec{X} - \vec{\mathrm{M}}_{\text{face}})^T \Sigma_{\text{face}}^{-1} (\vec{X} - \vec{\mathrm{M}}_{\text{face}}) - (\vec{X} - \vec{\mathrm{M}}_{\text{non-face}})^T \Sigma_{\text{non-face}}^{-1} (\vec{X} - \vec{\mathrm{M}}_{\text{non-face}})$$

where the parameters of the Gaussian (Σ, M) are the mean and covariance matrix of each class computed over the training set. If $\vec{X}^i = (\mathrm{H}^i \ \mathrm{D}^i \ \mathrm{S}^i)^T$ is the ith example out of N_{faces} of the face training set:

$$\vec{\mathrm{M}}_{\text{face}} = \frac{1}{N_{\text{faces}}} \sum_{i=1}^{N_{\text{faces}}} \vec{X}^i$$

is the mean faces vector and

$$\sum_{\text{face}} = \frac{1}{N_{\text{faces}}} \sum_{i=1}^{N_{\text{faces}}} (\vec{X}^i - \vec{\mathrm{M}}_{\text{face}}) \cdot (\vec{X}^i - \vec{\mathrm{M}}_{\text{face}})^T$$

is the covariance matrix of the face class.

Other parametric functional forms of the pdf were investigated. The simplest is a unimodal Gaussian of the face class: this assumes that the non-face class is uniformly distributed. In this case the face location is defined as the maximum of the square Mahalanobis distance to the mean center of face training examples. Mixture of Gaussians were also tested but led to very poor results. Due to the small amount of training data available, this method is out of scope in this paper.

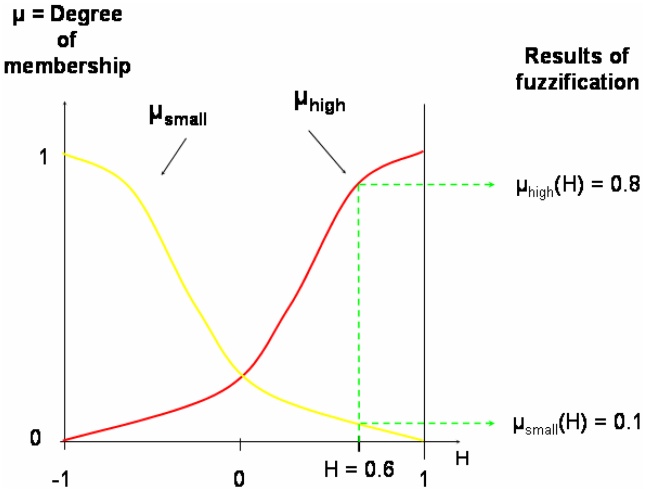


Fig. 16. Membership functions of the class "H high" and "H small".

5.3. Fuzzy Inference System

A face sub-image should be featured by a small Diabolo reconstruction error D, a high number of GHT votes H and a high proportion of skin pixels S. A classical set approach would define a threshold on each face model values. $\mathrm{H}_{\text{high}} = \{\mathrm{H} \mid \mathrm{H} > \text{thresh}\}$ the set of high H (for instance) values and $\mathrm{H}_{\text{small}} = \{\mathrm{H} \mid \mathrm{H} < \text{thresh}\}$ the set of small H values would be separated by this sharp boundary: a H value slightly under that threshold is then considered as small which make little sense. The fuzzy logic approach is more flexible by admitting partial membership to a class [40]. It is also coherent with natural language by introducing the degree of membership of H value in the class "high" and "small":

$$\mathrm{H}_{\text{high}} = \{\mathrm{H}, \mu_{\text{high}}(\mathrm{H})\} \quad \text{and} \quad \mathrm{H}_{\text{small}} = \{\mathrm{H}, \mu_{\text{small}}(\mathrm{H})\}.$$

In Fig. 16 a value of $\mathrm{H} = 0.6$ belongs the "high" class at 80% and the "small" class at 10%.

S value is the normalized proportion of skin in the sub-image: as H, high values of S correspond to high probability of the sub-image to contain a face. D value is the normalized Diabolo reconstruction error: the smaller it is, the higher is the probability of the sub-image to be a face one. For these three sources, two class are defined with respect to their value: high and small. As shown in Fig. 16, the membership functions for these classes are Gaussian functions centered respectively in $+1$ and -1.

To combine our three sources, a fuzzy inference system of Mamdani type [22] was built. A fuzzy inference system requires fuzzifying inputs, to formulate a set of linguistic rules and logical operators, and to aggregate results of the fuzzy rules. Three output class are defined as fuzzy sets: non-face, unknown, and face patterns.

Each output set is defined by a Gaussian membership function centered in 0 (non-face), $+0.5$ (unknown) or $+1$ (face), as shown in Fig. 17.

Considering only the ellipse model (H value), a simple statement can be formulated: if H is high then

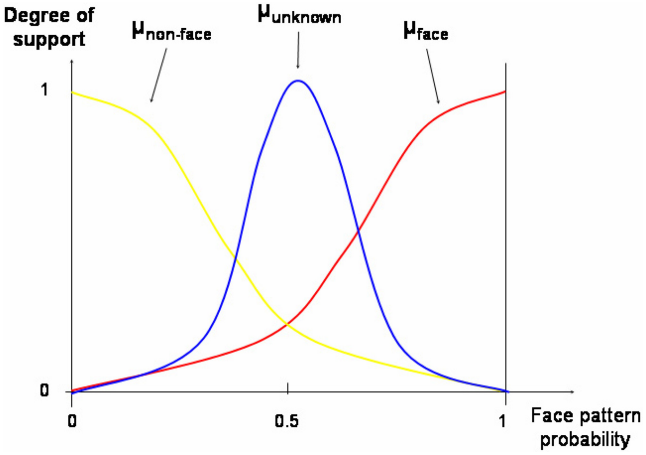


Fig. 17. Output fuzzy sets membership functions.

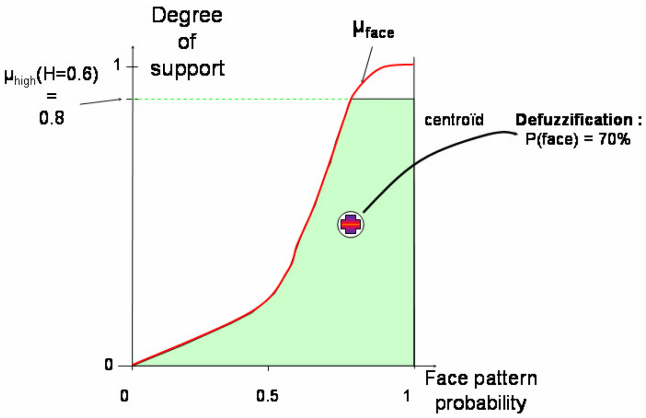


Fig. 18. Implication method: "then" operation.

sub-image is a face. Consequent of this fuzzy rule assigns a fuzzy set to the output which membership function is a truncation of the "face" set depending on the degree of support and according to the implication method (i.e. the mathematical definition of "then"). Degree of support in this particular statement only involving H value is the degree of membership in the "H high" class. The "then" operator results in a membership function equal to the minimum between the degree of support and the output fuzzy set membership function (the green area in Fig. 18 showing the case of H = 0.6).

Finally a decision can be made out of the resulting function by resolving a single value representing the probability of the sub-image to be a face pattern. A typical defuzzification method is the calculation of the center of the area under the curve (centroïd).

Now consider a statement with multi-part antecedent: if H is small or D is high or S is small then sub-image is unknown. The "or" fuzzy operation is mathematically defined as maximum of the three calculated degree of membership: this minimum is the degree of support for the output "unknown" set. In Fig. 19 a sub-image is featured by [H D S] = [0.6 0.8 −0.2]: for each source, a degree of membership is calculated. The "or" operation resolves them to a single number: the higher

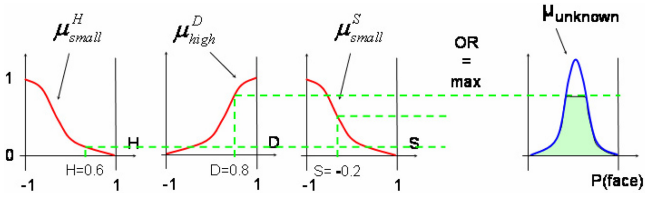


Fig. 19. Application of fuzzy operator "or".

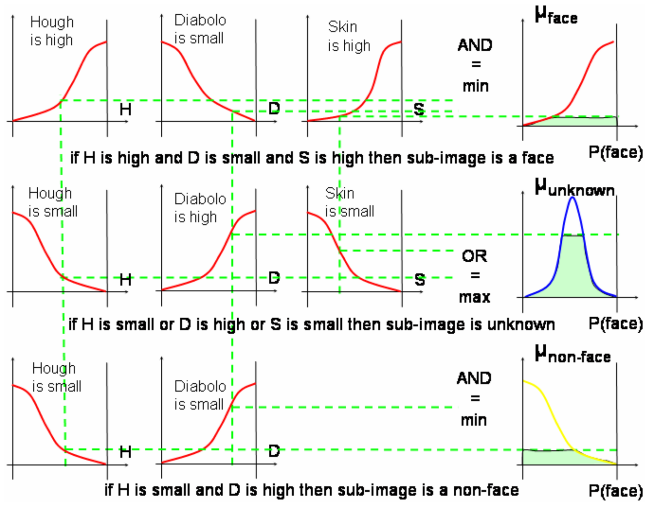


Fig. 20. Fuzzy inference diagram representing the rules.

value is kept as degree of support for the rule shaping the "unknown" fuzzy set.

One rule by itself leads to a very poor localization rate. We found experimentally that the three following fuzzy rules are optimal for face localization purpose:

—if H is high and D is small and S is high then sub-image is a face,

—if H is small or D is high or S is small then sub-image is unknown,

—if H is small and D is high then sub-image is a non-face.

The "and" operator is defined as the minimum of the degrees of membership. The rules are given the same weight, and order of the rules is unimportant as they are evaluated in parallel as shown in Fig. 20.

One can notice that the skin detector is not taken into account in the last rule: our skin color model is not elaborated enough and this is also noted with a weighted average combination (see next section).

Aggregation of the output fuzzy sets consists in calculating a membership function as the maximum of the three consequent membership functions calculated before (see Fig. 21).

This membership function is finally defuzzified by calculating the centroid of it, which provide a single number: the probability that the input sub-image is a face one.

This process is applied at all position of the original image to construct the "fuzzy" FusionMap used to define face location.

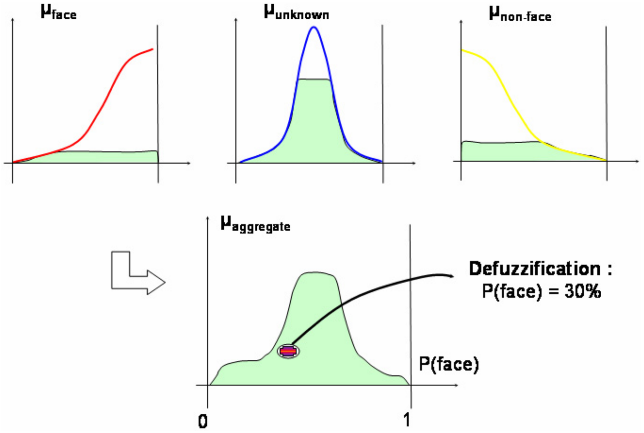


Fig. 21. The aggregate output fuzzy set.

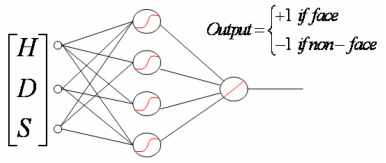


Fig. 22. Combining the three detectors with a multi-layer perceptron (MLP).

5.4. Weighted Average and Multilayer Perceptron

We investigate neural combination of the three face models: the three sources are the inputs of the multilayer perceptron (MLP). The MLP hidden neurons have sigmoid activation function, and the output neuron has a linear activation function as described in Fig. 22.

The training database is divided in two set: one for neural networks training, the other for assessment of the best architecture. 12713 face examples and 341316 non-face examples are used as training examples to learn parameters (weights) of the MLPs. Training is done using a gradient descent with adaptative learning rate stopped by cross-validation. The cost function implemented here is the sum over training examples of the square difference between target and network output.

The network is trained for pattern classification: target is $+1$ when the input [H D S] corresponds to a face and -1 else. Before training the MLP weights must be initialized: for a given neural net architecture (i.e. number of hidden neurons), several initializations are tested.

During the test phase the MLP output is a value of the interval $[-1 \ +1]$. Network output is calculated at all location of the image, which produces the "neural" FusionMap: face location is the position of the maximum of this map. The optimal MLP architectures is searched over 50 images (not used during training) containing only one face. MLPs with different number of hidden neurons, and different initialization of the weights are trained then assessed over this set. This exhaustive search leads us to the conclusion that the best architecture correspond to one output neuron. Actually, a growing number of hidden cells do not dramatically decreases the localization rate: for numbers of hidden neurons less than 3 the rates are quite the same order. The natural approach is to choose the simplest architecture for the MLP. That is to say the best neural combination is a weighted average of the inputs:

$$\text{FusionMap}_{i,j} = \text{a.H}_{i,j} + \text{b.D}_{i,j} + \text{c.S}_{i,j}$$

where a = 0.2280, b = −0.2620, and c = 0.1229.

One can notice the weight of the S input: as in the preceding section, it is half the weight of Hough or Diabolo response. This is due to the fact that the skin color model is pretty coarse.

This weighted average is compared to a simple average (same weight for the inputs: a = b = c = 1) in the next section.

6. EXPERIMENTAL RESULTS

In order to compare the combination strategies we used the ECU face database [26]: we compare the face localization rate of the algorithms on a test set of color images not used during training. Each of these image contains only one person, and the rectangle bounding the face is the same size over the whole set. A face is considered as correctly localized or not using the face ground truth and verification of a human operator. A correct localization of the face contains the eyes, the mouth, and is well-centered on the face.

6.1. ECU Face Detection Database

The ECU face and skin detection database was created in Edith Cowan University [26]. It has three sets of images particularly useful in our study (see Fig. 23). The first set is made of original color images. The second set is the corresponding ground-truth location of the faces. The third set is the ground-truth of skin pixels.

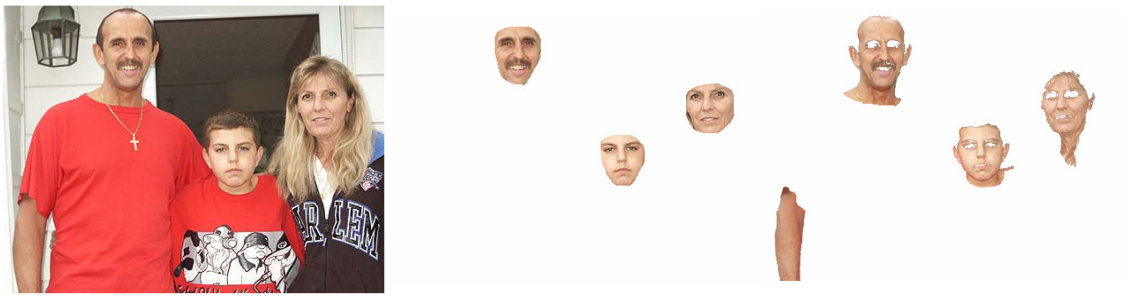
Almost all the images are taken from the Web, and were selected to have a wide variety of illumination conditions, background (mostly complex), face poses (upright, pan, tilted) and skin tones. It is widely depicted in [26].

Our test uses a set of 1353 images non overlapping with the training and cross-validation corpus. Each test image contains only one person.

6.2. Combiners Comparison

In the preceding section, different combination algorithms have been proposed. They include Bayesian classifier with parametric (unimodal Gaussian model of face and non-face) and non parametric techniques (histogram), fuzzy inference system, neural combination and weighted average.

It is important to outline the contribution of combination, and a reference for face localization rates.



(a) (b) (c)

Fig. 23. Sample from the ECU database: (a) original image, (b) face ground truth, (c) skin ground truth.

If we only consider the appearance-based model alone, the face location is defined as the minimum of Diabolo reconstruction error over the whole image, as a face image should be better reconstructed than a non-face image. Under this consideration, 656 faces out of 1353 are correctly localized: the localization rate is **48.5%**. Such a poor rate is explained in Section 4: the Diabolo is trained on examples of the interior part of the face, so we can see it as an eyes and mouth model. As an eye or a mouth detector it results in many false alarms in a cluttered scene [17]: non-face pattern is not compiled in the Diabolo. Moreover, edges are defined over each sub-window which makes appear patterns in a non-flat sub-image. And even if the Diabolo response shows a local minimum over the face area, lower minima can be found in unexpected area of the image.

Using the ellipse model alone, the face location is defined as the maximum number of vote given by the Generalized Hough Transform: 903 faces are correctly localized. In this case the face localization rate is higher: **67%**. The GHT is a cumulative approach more efficient than the appearance-based model. Missed faces of the test set correspond to an ellipse localized in a complex background with a lot of edge pixels from which a lot of vote were forecast to the accumulator.

The Bayesian classifier with the histogram technique reaches a rate of only **22%**. That means that the face and non-face distribution are strongly interleaved in the "H-D-S" space (see Fig. 24). The fuzzy approach is more efficient with a face localization rate of **72%**; it brings an improvement of 5% compared to the ellipse detector alone.

A classification based on the modeling of the face by a unimodal 3D Gaussian gives a poor **5%** of success. It means that the unimodal Gaussian center of the face class is not far enough from the non-face examples (see Fig. 25). When the three detectors respond strongly over the face region, it results in a feature vector HDS close to $A = [1 \ -1 \ 1]$ (area outlined by the red ellipse in Fig. 24(a). On the opposite, a lot of non-face sub-image are featured by a point close to point $B = [-1 \ 1 \ -1]$ in HDS (blue ellipse in Fig. 24(b) which corresponds to a non-face pattern for the three basics detectors. These two points should be correctly classify with a high confidence. But the face Gaussian center (Fig. 25) is at an equal Euclidian distance to these points: even with the covariance matrix of the face model it is not possible to discriminate samples from the two class. Results dramatically change if we use discriminant classification with both Gaussian distribution of face and non face. Indeed unimodal 3D-Gaussian of face and non-face Bayesian classification achieves **84%**. The non-face class Gaussian center is close to the non-face HDS point clouds as we can see on Fig. 25.

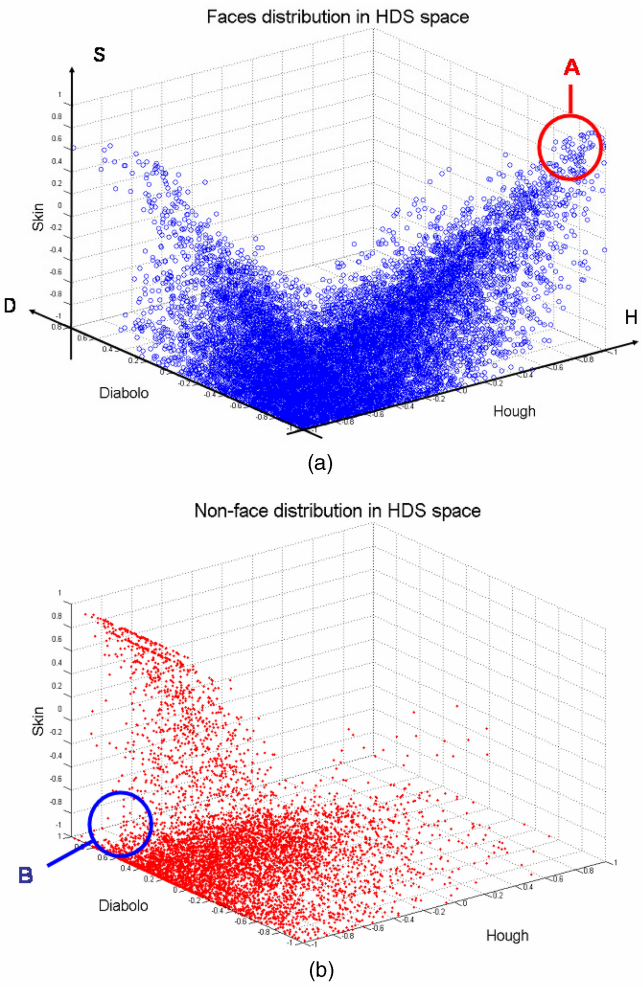


Fig. 24. Faces (a) and non-faces (b) distributions in the HDS space.

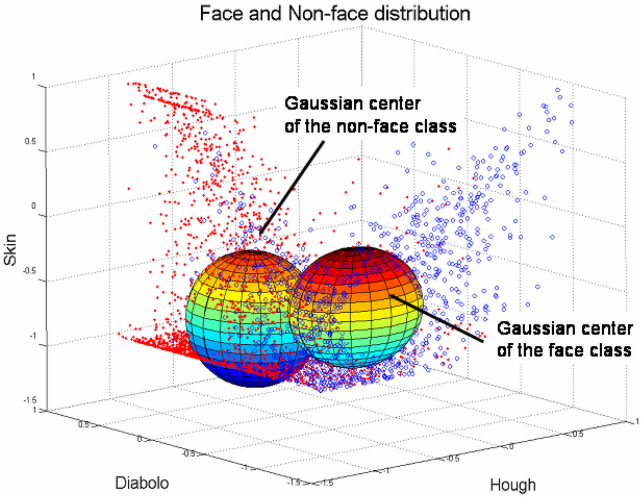


Fig. 25. Gaussian centers of the face and non-face distribution.

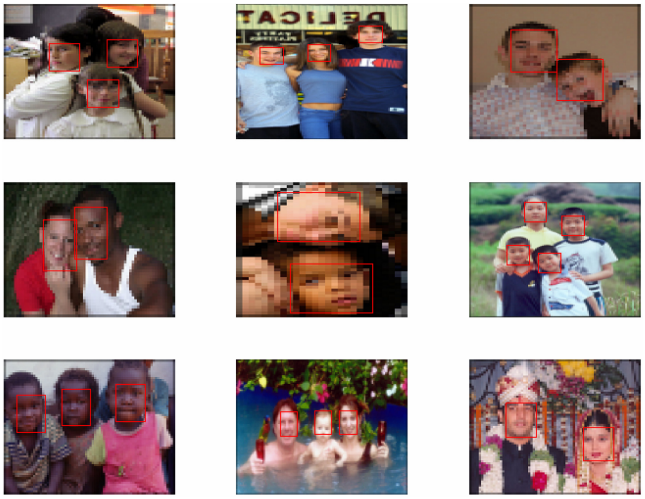


Fig. 27. Multiple faces localization: the number of faces is supposed to be known.

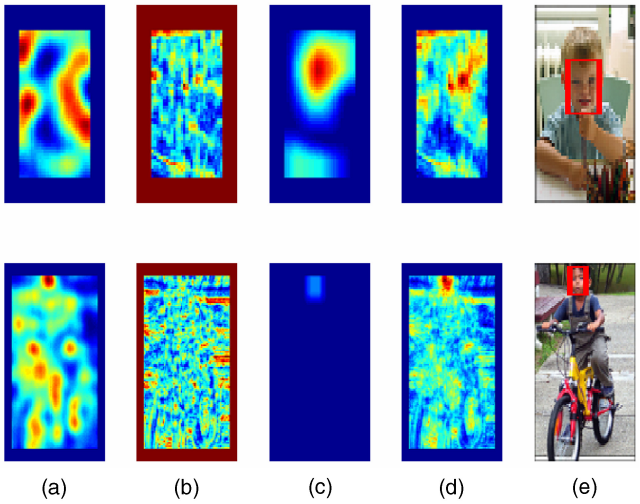


Fig. 26. (a) HoughMap (b) DiaboloMap (c) SkinMap (d) FusionMap and (e) the corresponding face localization on the original image.

Anyway, compare to all these methods, the weighted average performs the best. With a localization rate of **86%** it outperforms all the other approach. In order to measure the effect of the weights on the detection result, a simple average (i.e. all weights equal 1) is performed. With a rate of **80%** it performs well too, but less than the weighted average with the weights learned by gradient descent.

Amongst the multiple classifier systems, linear combiners are the most frequently used: a recent study can be found in [15] with a theoretical analysis based on the framework of [35]: the analysis of linear combiners is still a promising path of research.

Fig. 26 shows detectors response and their combination using the weighted average.

In the first example of Fig. 26, the ellipse detector failed to locate correctly the face, while the combination system did. In the second example, the SkinMap maximum is very low (0.23), but the combination (weighted average) brings a correct face location.

To validate the face localization rate of the weighted average combination, a second test was performed on 205 multiple faces images (non overlapping with the training and cross-validation corpus) containing a total of 482 faces. Number of people in each image is known in a face localization approach.

In single face images, face location is defined as the position of the maximum of FusionMap. In a N faces image (N is supposed to be known in a localization problem) the N highest maxima (with a sufficient distance to avoid overlapping detections) of FusionMap correspond to the location of the faces. 396 faces are correctly localized (**82**%). Some examples of correct localization are shown in Fig. 27.

For all tests of this sub-section the face size is also supposed to be known: this information can be retrieved if we know the distance between the person and the camera. Videos available at [25] were particularly interesting for this approach, showing people in front of their computer: face size does not vary widely along the image sequence.

The performance of the weighted average approach at a known scale on video sequences was tested on three videos sequences. In each image of the sequences only one person is present. Fig. 28 gives examples of the sequences, with the localization rates on each sequence.

6.3. Face Localization: Multiple Scale Approach

In the previous tests, face size is supposed to be known: it is the case when distance between the person and the camera is given. When this information is unavailable, a multi-scale approach of the weighted average combination is implemented. To localize faces of various sizes a pyramid of images is produced: the image is repeatedly subsampled with a classical [16, 36]

Fig. 28. Examples of face localization by our system on videos with the number of faces correctly localized. (a) Jamie sequence: 40 correct localization/43 images. (b) Ilkay sequence: 72 correct localization/80 images. (c) Geoff sequence: 24 correct localization/24 images.

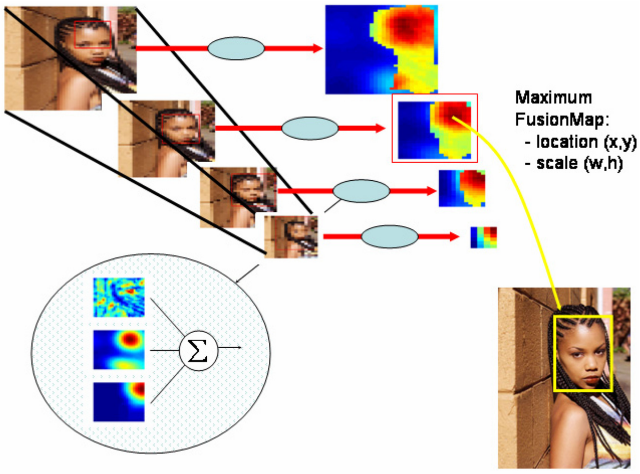


Fig. 29. Images Pyramid to deal with face size.

scale factor of 1.2. For each scale a FusionMap is built using a sliding window of a fixed size: face location probabilities are then compared across the different scales. Fig. 29 illustrates the images pyramid principle.

That multiscale approach is tested over 923 images containing one person with a face width superior to 100 pixels, so that the number of scale to scan is less than twelve. 537 faces are correctly localized: the face localization rate is **58%**. This rate is small compare to state of the art face detector [36]. We used the Haar face detector publicly available in [19]. A statistical model of the face, made of a cascade of boosted tree classifiers, is trained. The cascade is trained on face and non-face examples of fixed size 24×24. A 24×24

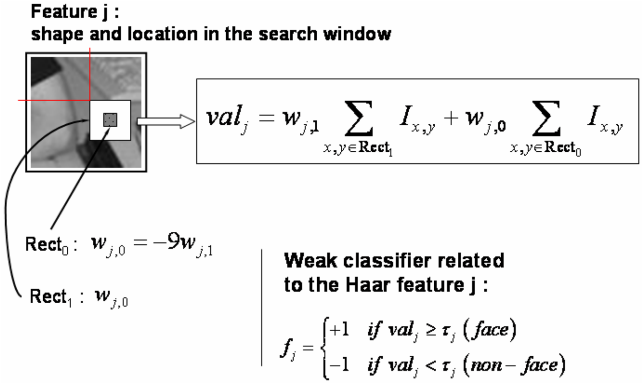


Fig. 30. A Haar-like feature is defined by its shape and its location relative to the 24×24 sliding window.

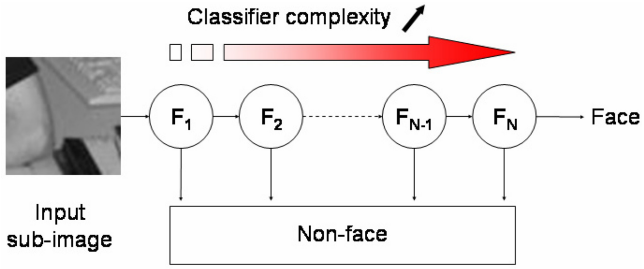


Fig. 31. Cascade of boosted classifiers.

sliding window scans the image and each sub-image is classified as face or non-face. To deal with face size the cascade is scaled with a factor of 1.2 by scaling the coordinates of all rectangles of Haar-like features. Hundreds of features are used as these shapes are applied at different position in the 24×24 retina: a feature is defined by its shape (including its size depending on a scale factor that defines the expected face size) and its location (see Fig. 30).

A simple decision tree classifier, referred to as "weak" classifier, processes the feature value. A complex classifier $F_k = \text{sign}(\sum_{i=1}^{n} c_i f_i)$ is iteratively computed as a weighted sum of weak classifiers using a boosting procedure. At each iteration a weak classifier parameters are trained and a weight c_j is assigned to the weak classifier relatively to its error on the training set. The trained weak classifier is then added to the sum and the training samples weights are updated in order to emphasize the misclassified ones to train the next weak classifier. Finally an attentional cascade is implemented: it is a cascade of boosted classifiers with increasing complexity. As shown in Fig. 31, the simplest classifiers comes first and is intended to reject majority of sub-window before calling more complex classifiers.

This face detector is robust to illumination condition but hardly work when face is too slanted. Fig. 32 illustrates the limitation of the detector: in the first row the face is correctly detected. In the second row the face moved slightly from the previous position and is not detected.

Localization rate measures a face localizer performance: a false positive also correspond to a missed

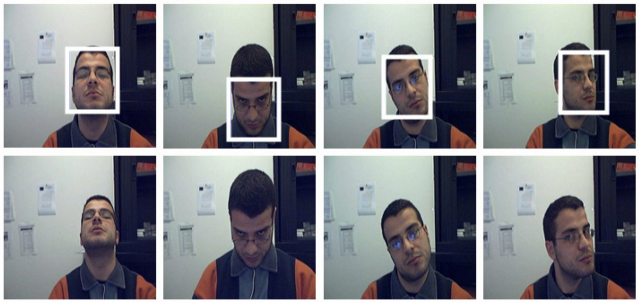


Fig. 32. Limits of the face detector.

face as only one location is searched in the image. A face detector is evaluated by its ROC performance: at least two scores are required, the detection rate (complementary of the missed rate) and the false positive rate. The cascaded face detector is more efficient than the weighted average combination. It detects 713 faces out of 923 (**77%**) with 78 false detections; 210 faces are missed. It is, with [16] the state-of-the-art in face detection. Its multi-scale approach is more efficient that the usual pyramid of images produced by down-sampling the original image: it scales the Haar filters, so that the search window contains a "high" resolution sub-image whatever the scale considered. In 210 images (23%) of the 923 test images, the face is missed. They correspond to faces highly rotated (pan, tilt or roll rotation) or occluded. On these particularly difficult images the weighted average localizer performs quite well with 90 faces correctly localized out of 210 (43%). It appears that our approach could be used as an alternative to the Haar detector when it fails to detect anyone in the scene. It potentially could decrease the missing rate by 43%.

7. CONCLUSION AND PROSPECTS

This paper aimed to present a significant contribution to the image fusion task with application to face localization. We have presented three different detectors: skin color, auto-associative multi-layer perceptron, and ellipse Hough Transform. We proposed three various combination schemes and compare them: Bayesian classifier, fuzzy logic and connexionist. An awesome improvement of localization rate is brought by the two last methods.

For the face detection/localization issue, several improvements are in progress: more sophisticated skin color models like ellipsoidal threshold, Gaussian density functions or mixture of Gaussians [38]. A more efficient appearance-based model is also elaborated, based on the Viola&Jones face detector [36]. For the combination part, it is not clear when and why a combination method outperforms the others: quantitative and qualitative investigations of classifiers output correlation effect on combiners performance are under study.

REFERENCES

[1] R. Belaroussi, L. Prevost and M. Milgram
Combining model-based classifiers for face localization.
In *Proceedings of the IAPR Conference on Machine Vision Application*, 2005, 290–293.

[2] S. M. Bileschi and B. Heisele
Advances in component based face detection.
In *Proceedings of the IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, 2003, 149–156.

[3] M. C. Burl, M. Weber and P. Perona
A probabilistic approach to object recognition using local photometry and global geometry.
In *Proceedings of the European Conference on Computer Vision*, Vol. 2, 1998, 628–642.

[4] P. Campadelli, R. Lanzarotti, G. Lipori, and E. Salvi
Face and facial feature localization.
In *Proceedings of the International Conference on Image Analysis and Processing*, Vol. 3617 of Lecture Notes in Computer Science, 2005, 1002–1009.

[5] D. Chai and K. N. Nang
Locating facial region of a head-and-shoulders color image.
In *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition*, 1998, 124–129.

[6] R. Choudhury Verma, C. Schmid and K. Mikolajczyk
Face detection and tracking in a video by propagating detection probabilities.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **25**, 10 (Oct. 2003), 1215–1228.

[7] J. Czyz, J. Kittler and L. Vandendorpe
Combining face verification experts.
In *Proceedings of the IAPR International Conference on Pattern Recognition*, Vol. 2, 2002, 28–31.

[8] D. DeMers and G. Cottrell
Non-linear dimensionality reduction.
In *Proceedings of the Conference on Neural Information Processing Systems*, Vol. 5, 1993, 580–587.

[9] J. Dowdall, I. Pavlidis and G. N. Bebis
Face detection in the near-IR spectrum.
Image and Vision Computing, **21**, 7 (July 2003), 565–578.

[10] R. O. Duda, P. E. Hart, and D. G. Stork
Pattern Classification.
New York: Wiley, 2001.

[11] C. K. Eveland, D. A. Socolinsky and L. B. Wolff
Tracking human faces in infrared video.
Image and Vision Computing, **21**, 7 (July 2003), 579–590.

[12] P. Felzenschwab and D. Huttenlocher
Efficient matching of pictorial structures.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2000, 66–73.

[13] R. Féraud, O. Bernier, J. Viallet, and M. Collobert
A fast and accurate face detector based on neural networks.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **23**, 1 (Jan. 2001), 42–53.

[14] B. Froba and W. Zink
On the combination of different template strategies for fast face detection.
In *Proceedings of the International Workshop on Multiple Classifier Systems*, 2001, 418–428.

[15] G. Fumera and F. Roli
A Theoretical and experimental analysis of linear combiners for multiple classifier systems.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **27**, 6 (June 2005), 942–956.

[16] C. Garcia and M. Delakis
Convolutional face finder: A neural architecture for fast and robust face detection.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **26**, 11 (Nov. 2004), 1408–1423.

[17] M. Hamouz, J. Kittler, J-K. Kamarainen, P. Paalanen, H. Kalviainen and J. Matas
Feature-based affine-invariant localization of faces.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **27**, 9 (Sept. 2005), 1490–1495.

[18] M. Hu, S. Worrall, A. H. Sadka and A. M. Kondoz
Automatic scalable face model design for 2D model-based video coding.
Signal Processing: Image Communication, **19** (2004), 421–436.

[19] Intel Corporation
Open Source Computer Vision Library—Reference Manual, 2001, homepage: http://developer.intel.com.

[20] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz
Robust face detection using the Hausdorff distance.
In *Proceedings of the IEEE International Conference on Audio- and Video-based Biometric Person Authentication*, 2001, 90–95.

[21] M. J. Jones and J. M. Rehg
Statistical color models with application to skin detection.
International Journal of Computer Vision, **46**, 1 (Jan. 2002), 81–96.

[22] E. H. Mamdani and S. Assilian
An experiment in linguistic synthesis with a fuzzy logic controller.
International Journal of Man-Machine Studies, **7**, (1) 1975, 1–13.

[23] S. J. McKenna, S. Gong, and Y. Raja
Modeling facial color and identity with gaussian mixtures.
Pattern Recognition, **31**, 12 (Dec. 1998), 1883–1892.

[24] K. Messer, J. Matas, J. Kittler, J. Luettin and G. Maitre
XM2VTSDB: The extended M2VTS database.
In *Proceedings of the International Conference on Audio- and Video-based Biometric Person Authentication*, 1999, 72–77.

[25] Microsoft Research Cambridge
Homepage: http://research.microsoft.com/vision/cambridge/i2i/DSWeb.htm.

[26] S. L. Phung, A. Bouzerdoum and D. Chai
Skin segmentation using color pixel classification: Analysis and comparison.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **27**, 1 (Jan. 2005), 148–154.

[27] L. Prevost and M. Milgram
Automatic allograph selection and multiple expert classification for totally unconstrained handwritten character recognition.
In *Proceedings of the IAPR International Conference on Pattern Recognition*, Vol. 1, 1998, 381–383.

[28] A. F. Rahman and M. C. Fairhust
Multiple classifier decision combination strategies for character recognition: A review.
International Journal on Document Analysis and Recognition, **5**, 4 (July 2003), 166–194.

[29] H. Rowley, S. Baluja and T. Kanade
Neural network-based face detection.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **20**, 1 (Jan. 1998), 23–38.

[30] H. Schwenk and M. Milgram
Transformation invariant auto-association with application to handwritten character recognition.
In *Proceedings of the Conference on Neural Information Processing Systems*, Vol. 7, 1995, 991–998.

[31] K. Schwerdt and J. L. Crowley
Robust face tracking using color.
In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2000, 90–95.

[32] R. Séguier
A very fast adaptive face detection system.
In *Proceedings of the International Conference on Visualization, Imaging, and Image Processing*, 2004.

[33] F. Smeraldi and J. Bigun
Retinal vision applied to facial features detection and face authentication.
Pattern Recognition Letters, **23**, 4 (Feb. 2002), 463–475.

[34] K. K. Sung and T. Poggio
Example-based learning for view-based human face detection.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **20**, 1 (Jan. 1998), 39–51.

[35] K. Tumer and J. Ghosh
Analysis of decision boundaries in linearly combined neural classifiers.
Pattern Recognition **29**, 2 (Feb. 1996), 341–348.

[36] P. A. Viola and M. J. Jones
Rapid object detection using a boosted cascade of simple features.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, 2001, 511–518

[37] G. Yang and T. S. Huang
Human face detection in complex background.
Pattern Recognition, **27**, 1 (Jan. 1994), 53–63.

[38] M-H. Yang, D. Kriegman, and N. Ahuja
Detecting faces in images: A survey.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **24**, 1 (Jan. 2002), 34–58.

[39] A. Yuille, P. Hallinan and D. Cohen
Feature extraction from faces using deformable templates.
International Journal of Computer Vision, **8**, 2 (Aug. 1992), 99–111.

[40] L. A. Zadeh
Outline of a new approach to the analysis of complex systems and decision processes.
IEEE Transactions on Systems, Man, and Cybernetics, **3**, 1 (Jan. 1973), 28–44.

Rachid Belaroussi received the M.S. degree in electronics from the University Pierre and Marie Curie and the engineer diploma of Ecole Supérieure de Physique et de Chimie Industrielle in 2002. He is pursuing a Ph.D. degree in computer vision at the LISIF laboratory. His research activities are focused on face detection and tracking in videos or still images, and include image processing, machine learning and neural networks.

Lionel Prevost received the Ph.D. degree in pattern recognition from the University Pierre and Marie Curie–Paris 6 in 1998. His research interests included machine learning and information fusion for handwriting recognition. He is currently associate professor at this university, in the Institute of Intelligent Systems and Robotics (Perception team). His current activities are in the fields of neural networks, pattern recognition, image and video analysis with application to face and facial feature localization and vehicle classification. He serves on the program committee of various conferences.

Maurice Milgram was born in 1948 in Paris and obtained the Agrégation of Mathematics in 1971 and a Ph.D. from UTC in probabilistic automata networks in 1981. He started as an assistant professor at Technology University of Compiègne (1973) then he became full professor (1983) at ENSEA and joined the Robotic Laboratory of the Pierre and Marie Curie University in Paris in 1986. He founded the PARC Laboratory, which is now part of the LISIF, in 1992. He has supervised about 30 Ph.D. thesis and 20 industrial contracts. His research interests concern pattern recognition and image processing.

His current research activities are focus on face localization/detection, gaze tracking, person tracking in image sequences, gesture recognition. His collaboration with public and private research laboratories includes recent contracts such as person localization for a smart airbag (Faurecia), automatic directional control of the vehicle's headlamp beams (Valeo), vehicle type recognition (LPR), biometry (Sagem), on-vehicle obstacle detection (PSA), baby gaze tracking (Necker Hospital).

Fusion of Redundant Information in Brake-By-Wire Systems Using a Fuzzy Voter

REZA HOSEINNEZHAD
ALIREZA BAB-HADIASHAR
Swinburne University of Technology

In safety critical systems such as brake-by-wire, fault tolerance is usually provided by virtue of having redundant sensors and processing hardware. The redundant information provided by such components should be properly fused to achieve a reliable estimate of the safety critical variable that is sensed or processed by the redundant sensors or hardware. Voting methods are well-known solutions for this category of fusion problems. In this paper, a new voting method, using a fuzzy system for decision-making, is presented. The voted output of the proposed scheme is a weighted average of the sensors signals where the weights are calculated based on the antecedents and consequences of some fuzzy rules in a rule-base. In a case study, we have tested the fuzzy voter along with the well-known majority voting method for a by-wire brake pedal that is equipped with a displacement sensor and two force sensors. Our experimental results show that the performance of the proposed voting method is desirable in the presence of short circuits to ground or supply, excessive noise and sensor drifts. Voting error (in terms of mean square error) is reduced by 82% by the proposed fuzzy voting method, compared to majority voting.

Manuscript received December 28, 2004; revised September 30, 2005 and April 25, 2006.

Refereeing of this contribution was handled by Professor David Hall.

This research work was supported by Research Centre for Advanced By-Wire Technologies (RABiT) and Pacifica Group Technologies Pty. Ltd.

Authors' address: Faculty of Engineering and Industrial Sciences, Swinburne University of Technology, Hawthorn, Victoria 3122, Australia, E-mail: (rhoseinnezhad@swin.edu.au, abab-hadiashar@swin.edu.au).

1. INTRODUCTION

Brake-by-wire is a frontier technology that will allow many braking functions to switch to electronic actuation. Its deployment will lead to more effective and safe braking systems, elimination of hydraulic technology, release of space and reduction of maintenance. Design and implementation of brake-by-wire systems has recently attracted interest from researchers in automotive and control engineering [9–12, 17]. The general architecture of a brake-by-wire system is shown (in schematic form) in Fig. 1. The figure shows that a large variety of sensors are utilised in a brake-by-wire system and therefore their consistent operation is vital for the functionality of such a system. To achieve a high level of coherency amongst such a large collection of sensors (mandated by the safety requirement of a brake system), the use of sophisticated data fusion techniques is unavoidable.

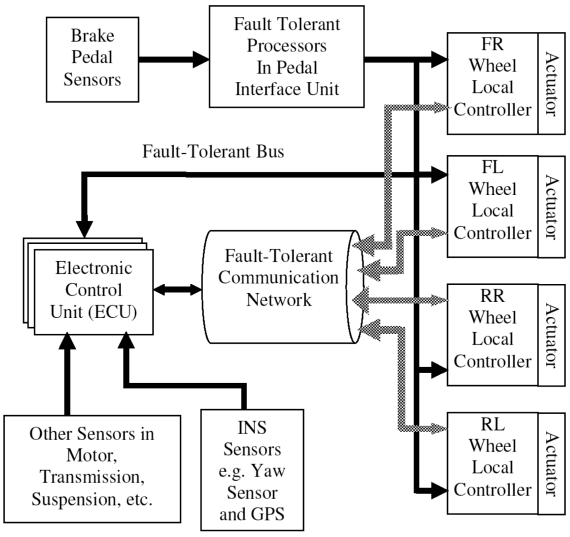


Fig. 1. A schematic architecture of a brake-by-wire system.

A brake-by-wire system, by nature, is a safety critical system and therefore fault tolerance is a vitally important characteristic of this system. As a result, a brake-by-wire system is designed in such way that many of its essential information would be derived from a variety of sources (sensors) and be handled by more than the bare necessity hardware. Three main types of redundancy usually exist in a brake-by-wire system:

1) Redundant sensors in safety critical components such as the brake pedal in Fig. 1.
2) Redundant copies of some signals that are of particular safety importance such as displacement and force measurements of the brake pedal copied by multiple processors in the pedal interface unit in Fig. 1.
3) Redundant hardware to perform important processing tasks such as multiple processors for the electronic controller unit (ECU) in Fig. 1.

Reliability, fault tolerance and accuracy are the main targeted outcomes of the fusion techniques that should be developed especially for redundancy resolution inside a brake-by-wire system. In order to utilise the existing redundancy, voting algorithms need to be evaluated, modified and adopted to meet the stringent requirements of a brake-by-wire system.

Several well known voting algorithms have been widely used in fault tolerant systems such as avionics and railway systems [6–8, 13] and fault tolerant VLSI circuits [4–6, 8, 13]. The n-input majority voter [1] produces a correct result if at least $[(n + 1)/2]$ voter inputs match each other. In cases of no majority, the voter generates an exception flag, which can be detected by the system supervisor to move the system toward a safe state. As an extended version of majority voter, plurality voter [2] implements "m out of n" voting, where m is less than a strict majority. Median voter is a mid-value selection algorithm. Assuming an odd number of redundant inputs, this algorithm successively eliminates pairs of outlying values until a single result remains. The weighted average voter [21], on the other hand, calculates the weighted mean of its redundant input values. Parhami [16] examined the performance of different voting techniques, in terms of their execution time, and proposed efficient implementations of a variety of algorithms.

There is no agreement checking in weighted average and median voters [15]. Hence, they are not appropriate for safety critical applications such as braking. In the case of lack of majority agreement, majority voters give no result in the output and instead a flag is set. In a brake-by-wire system, however, "no result" is not acceptable as the output of fusion. Instead, a status bit is generated for each sensor.[1] If the sensors do not agree, invalidity of the voter output will be deduced from the status bits. Another problem with a majority voter is its considerable output discontinuity in the event of long-time disagreements [14, 18]. Latif-Shabgahi and his colleagues tried to solve this problem by introducing a smoothing voter in which an agreement-checking threshold is adaptively set when the voter produces no result. While their proposed method results in a lower number of no result events in the output of the voter, such events are not completely eliminated.

As an alternative solution for the problem, we propose to use the mean of agreeing sensors as the output of a majority voter and use their median value if there is no agreement. In this method that we call hard voting, a status bit is set if the sensors agree, and reset if they don't. The main issue in this voting method is how to set the geometric distance threshold [18] value by which sensor agreement is checked. Due to sensor conversion errors, there is almost always a distance between two agreeing sensors of different types. Therefore, distance threshold should be large enough to prevent incorrect decisions about sensor agreements in the presence of sensor conversion errors. A large value for the distance threshold in the hard voting method will, however, give rise to late fault detection if the fault causes a gradual change in the sensory signal. Such faulty gradual changes in sensory signals usually happen because of drifts, short circuits,[2] and sensor noise that gradually increases with temperature.

Genetic algorithms have also been applied for voting [19]. This approach, however, is only efficient when used with off-line calculations and in particular, for cases when the population of redundant components is large.

In this paper we propose a new voting method, called soft voting (in contrast to its alternative, hard voting), using a fuzzy logic paradigm. By using fuzzy logic rule-base inference, a faulty sensor is gradually removed from the output of our proposed soft voter. Instead of status bits, a faultiness measure is defined for each sensor that gradually increases in the event of faults. Although fuzzy inference and fuzzy systems have been utilised for sensor fusion in drive-by-wire applications, they have been employed merely to generate control commands or signal estimates for control and estimation applications in drive-by-wire technology [3, 20].

The fuzzy voter introduced in this paper is novel in the sense that it actually realises an adaptive weighted averaging mechanism for voting in which the weights are intelligently determined by the fuzzy inference engine. This inference engine is designed in such a way that faulty sensors are automatically detected based on the geometric distance between their outputs and other sensory measurements. As such distances grow, the weights corresponding to faulty sensors gradually decrease toward zero. To our knowledge, fuzzy systems have not been applied for voting in such a scheme.

For voting applications in systems with redundant sensors (or information sources), our proposed soft voter has the following advantages compared to other existing methods: Firstly, it does not output "no result." Secondly, it is capable of early detection and rejection of faulty sensors. Thirdly, its noise tolerance is higher than existing methods (due to the automatic fault detection and noise rejection phenomenon realised by the fuzzy inference machine). In addition, the output of our proposed voter does not suddenly jump or fall in case of signal short-circuits, and finally its computational complexity is comparable with simple voting methods like majority voters (particularly for a small number of sensors). These advantages all together make the proposed voter significantly efficient for real-time voting in redundant multi-sensor systems. We emphasize that most of the many voting techniques in the current lit-

[1] Henceforward, by sensor, a source of information is intended. It can be a redundant sensor, a redundant signal or a redundant processor.

[2] The RC filters that are connected to the inputs of ADCs (analog to digital converters) cause a gradual change in sensory signals when a short circuit happens.

erature have been designed for voting on multiple decisions (equivalent to fusion in decision or symbol level) while the method proposed in this paper and the methods reviewed in this section are applicable to voting on redundant signals i.e., the cases involving signal-level fusion.

We introduce our soft voting method in Section 2. Implementation of a soft voter for fusion of the redundant information provided by three sensors of a brake pedal is presented in Section 3. Then comparative experimental results of hard and soft voting methods on real sensory data will also be given in this section. Among the voting methods reviewed in this paper, hard voting is the closest to the proposed fuzzy voter in a sense that it is also a weighted-averaging voter but the weights have binary values and jump to zero in the case of a faulty sensor. Our soft voter is capable of early detection of faulty sensors and makes the weights gradually decrease toward zero in case of such faults. Due to their similarity and their meaningful difference, the fuzzy soft voter and the hard voter have been compared in Section 3 as a fair comparison. Section 4 concludes this paper. Although our method has been implemented and experimented for fusion of redundant safety critical components in a brake-by-wire system, the general scheme of our proposed fuzzy voter, explained in Section 2, can be applied to fuse redundant information in any application with safety critical issues and fault tolerance requirements.

2. PROPOSED SOFT VOTING METHOD

The block diagram of the proposed fuzzy voter for fusion of redundant information is shown in Fig. 2. In this diagram, n sources of information (redundant sensors, signals or hardware) are called $S_1, S_2, \ldots, S_n$. Initially, low-pass filtering (to reduce the noise) and missing data handling (by using a multi-step ahead predictive filter [10, 11]) are performed on the raw sensory data. Then the signals are converted to an internal representation, which is a common format for the multi-source information. This conversion is required because different types of information (e.g. position data in millimetres and force information in Newton) should be converted to their equivalent values in a common format (internal representation) so that they have the same physical dimension before being compared and fused by a voter. The converted signals denoted by $x_1, x_2, \ldots, x_n$ are processed by an agreement evaluation block, resulting in $n(n-1)/2$ metrics denoted by $\{\alpha_{i,j} \mid i = 1, \ldots, n-1;\ j = i+1, \ldots, n\}$. In this block, the agreement of each pair of signals is quantified by an Euclidian distance measure. For example the agreement of the two sensors S_i and S_j is evaluated by the following equation:

$$\alpha_{i,j} = |x_i - x_j| \tag{1}$$

where x_i and x_j are the converted signals corresponding with the sensors S_i and S_j. In the final step, the sensory

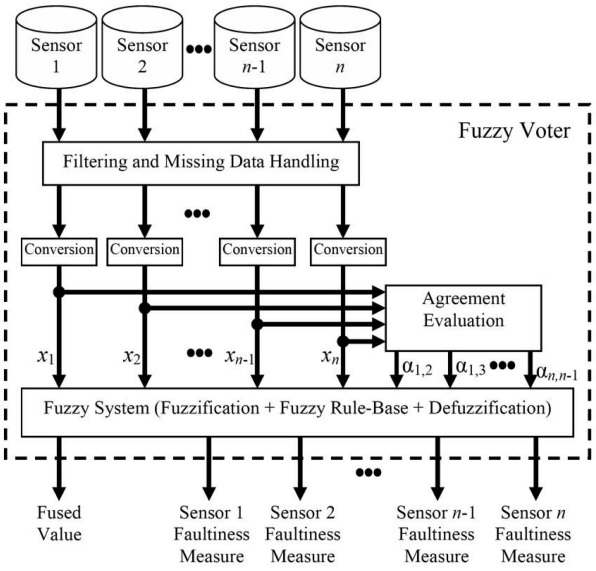


Fig. 2. Block diagram of the proposed fuzzy voter for fusion of redundant sensory information.

data $x_1, x_2, \ldots, x_n$ and their agreement evaluations $\{\alpha_{i,j}\}$ are passed on as inputs to a box that is responsible for fusion by voting. This box is a fuzzy system, comprising the common three subsystems i.e., fuzzification, a fuzzy rule-base and defuzzification. The fuzzy system has two outputs: a voted value as the main fusion output, and n "faultiness measures" (instead of status bits) for the sensors. Each faultiness measure is a quantitative evaluation of voter's belief in the faultiness of a sensor in $[0, 1]$, with a value of 1 for total belief.

A hard voter outputs a fused value and n status bits, showing the occurrence of faults in the sensors. More precisely, the hard voter does not need a fuzzy rule-base. Instead, its outputs are determined based on the results of comparing $\alpha_{i,j}$ values with an agreement threshold. For instance, in the case of $n = 3$ if $\alpha_{1,2}$ and $\alpha_{1,3}$ are higher than the threshold (i.e., S_1 and S_2 do not agree with each other; so do the pair of S_1 and S_3) and $\alpha_{2,3}$ is lower than the threshold (i.e., S_2 and S_3 agree with each other), then the hard voter will deduce that S_1 is faulty. In this case, the fused output will be the average of S_2 and S_3 and the faultiness status bits will be 100 for S_1, S_2 and S_3, respectively.

The agreement threshold is important in the voting process. It is tuned based on the $\alpha_{i,j}$ values in a normal working condition, when no sensor is faulty. They should be greater than the maximum $\alpha_{i,j}$ values in normal conditions, in such a way that conversion errors don't cause the voter to incorrectly assume that two sensors disagree. However, if a sensor gradually deviates from its true values because of sensor drifts or noise or short circuits, then the large thresholds cause a long delay in detection of the fault by a majority voter.

Our proposed soft voting method is mainly intended to solve the problem of late fault detection, and to prevent large discontinuities in the fusion output. Like any

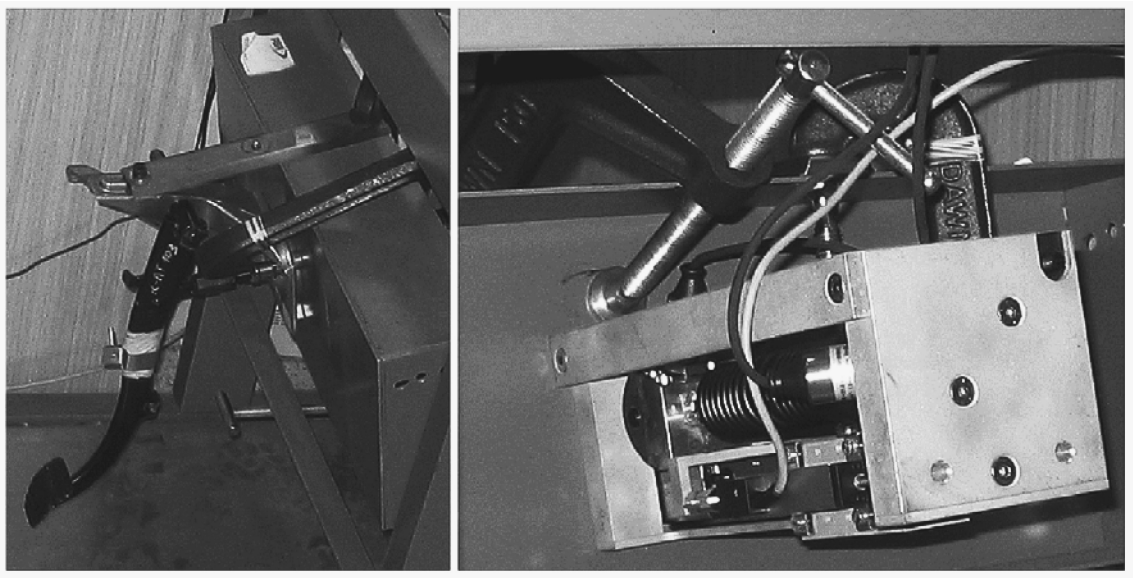


Fig. 3. Brake pedal and its sensors in our case study.

fuzzy system, $\alpha_{i,j}$ inputs are fuzzified first. We define three fuzzy sets of Large, Medium and Small agreements by their membership functions. These definitions are based on empirical maximum values of $\alpha_{i,j}$, derived from measurements and conversions. In practice, we collect some measurements from fine sensors and calculate the $\alpha_{i,j}$ values for each multi-sensory measurement. In case of triangular membership functions, if the maximum of $\alpha_{i,j}$ values is $\alpha_{\max}$, then breaking points of the Small fuzzy set are 0–$1.7\alpha_{\max}$, the breaking points of the Medium fuzzy set are $\alpha_{\max}$–$1.7\alpha_{\max}$–$2.3\alpha_{\max}$, and the breaking points of the Large fuzzy set are $1.7\alpha_{\max}$–$2.3\alpha_{\max}$. Generally, the application experts can determine the proper levels of $\alpha_{i,j}$ set as breaking points for Small, Medium and Large fuzzy sets. Based on the logic of majority voting, each fuzzy rule in the rule-base determines a voted output and n faultiness measures. For example to vote three sensors, a typical fuzzy rule is expressed as follows:

IF
S_1 and S_2 agreement is Small
AND S_2 and S_3 agreement is Large
AND S_3 and S_1 agreement is Small
THEN
The fused output is the average of S_2 and S_3
AND S_1 faultiness is Large
AND S_2 faultiness is Small
AND S_3 faultiness is Small.

This rule explains what is logically expected as a voting result if S_1 does not agree with the other two sensors. The final defuzzified fusion output is calculated as a weighted average of all possible expected outputs by the following equation:

$$\text{Fused Output} = \sum_{i=1}^{M}(w_i O_i) \Big/ \sum_{i=1}^{M} w_i \qquad (2)$$

where M is the number of rules in the rule-base, O_i is the fused output as it appears in the consequence of the ith fuzzy rule and the weight w_i is the product of membership values of the conjoined parts of the antecedent of the rule. If the exemplar rule given above is the kth fuzzy rule in the rule-base, then $O_k = (x_2 + x_3)/2$ where x_2 and x_3 are the filtered sensory signals of S_2 and S_3 after conversion to the internal representation, as shown in Fig. 2. These weights smoothly change from 0 to 1 or reverse, and the fused output is smoothly switched from one vote to the other, hence the name soft voter. Sensor faultiness measures are defuzzified into crisp outputs by a fuzzy centroid method. In this method, a fuzzy number is transformed to crisp by taking the centre of gravity of its membership function. More precisely, if Y is a fuzzy number with its membership functions determined as $\mu_Y(y)$, then the centroid crisp of Y is given as below:

$$y = \int_{-\infty}^{+\infty} \alpha \mu_Y(\alpha) d\alpha.$$

3. EXPERIMENTAL RESULTS

We implemented our fuzzy voter to fuse the redundant information provided by three sensors mounted on a brake-by-wire pedal. Two sensors measure the force and the third sensor measures the pedal displacement. Although the sensors are different, they are redundant sources of information in the sense that they provide measurements for the same quantity: driver's brake demand. A photograph of the brake pedal and its sensors are displayed in Fig. 3.

As we have shown in the brake-by-wire diagram in Fig. 1, the displacement and force signals are preprocessed (low-pass filtering and missing data handling) by fault tolerant processors in the pedal interface unit

and then transferred to four wheels via a fault tolerant communication bus (e.g. a LIN-bus). The processed sensory data are also sent to an electronic control unit (ECU) that includes a number of redundant processors generating the high level braking commands, such as anti-skid braking system (ABS), vehicle stability control (VSC) or traction control (TC).

In order to provide a reliable estimate for the driver's brake demand, pedal sensor data are voted in the ECU, where the resulting brake demand is then fused with the other vehicle sensor data (e.g. wheel speed or INS—Inertial Navigation System—sensors like accelerometers and gyros) to generate four final brake commands. To activate the brake actuators, these commands are sent to the local controllers in the four brake callipers via a fault tolerant time-triggered communication network. If for any reason the ECU is faulty then pedal sensory data will be voted in the local controller of each wheel unit, leading to generation of a brake response on each wheel. The main purpose of voting is to detect sensor faults (such as excessive noise, short circuits or sensor drifts) and to remove the effects of faulty measurements from the brake demand. In the presence of a fault or a substantial level of noise in sensor signals, they will not agree with each other. A voter should detect these disagreements and use them to identify faulty sensors. A hard voter simply discards faulty sensor data and outputs the average of agreeing sensors.

Fig. 4 shows a block diagram of the pedal sensor fusion scheme which is the revised version of the diagram shown in Fig. 2, for our experiments. S_1 and S_2 are the two force sensors giving f_1 and f_2, and S_3 is the displacement sensor with its signal denoted by x. Force is the quantity selected as the internal representation for fusion of the three sensors. In other words, the pedal displacement signal is converted to equivalent force signals $\hat{f}_1$ and $\hat{f}_2$ to be compared with the signals provided by the other two sensors. In order to perform this conversion, a model is required to mathematically relate the three signals x, f_1 and f_2. The passive push-return mechanism of the pedal can be modelled with an ideal spring in parallel with a damper, as shown in Fig. 5. The two force sensors are located at the two ends of the paralleled spring and damper model. Since the acceleration of pedal movements is too small to be considered in the model, the effect of the pedal mass is neglected. Thus, the two force sensor measurements are very close and have been simply labelled with f in Fig. 5 and the following equations. Based on the simplified damper-spring model, the following equation expresses the measured force signals in terms of the measured displacement signal:

$$f = kx + b\dot{x} \tag{3}$$

where k and b are the spring and damping factors, respectively.

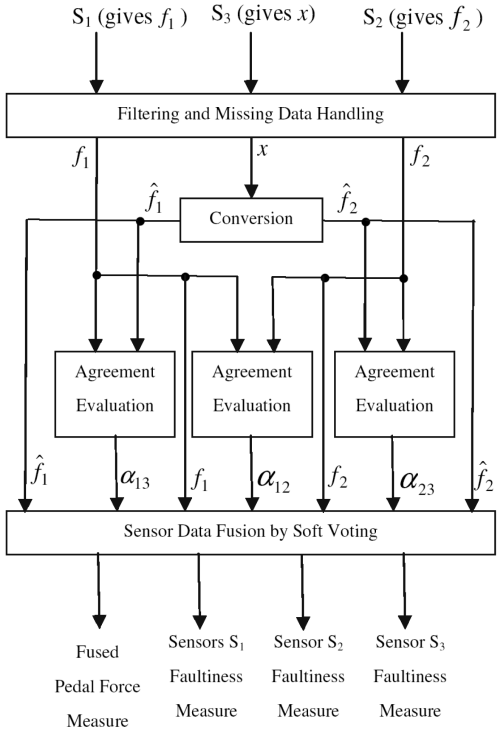


Fig. 4. Block diagram of pedal sensor fusion.

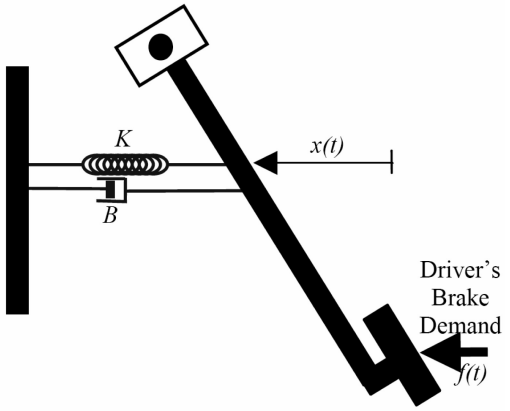


Fig. 5. A simplified model of the pedal and its sensors.

In order to validate the model and estimate its parameters, we ran a number of experiments and collected the three sensors measurements. In these experiments, the pedal set was installed in a car and a driver used it for different braking scenarios such as continuous soft brakes, frequent push-release and panic brakes. Using the collected sensory data, we examined the linearity between force, displacement and velocity using a least squares (LS) technique. More precisely, we utilised the recorded signals f, x and dx/dt and obtained a LS estimate of the parameters k and b in (3). This resulted in a low correlation coefficient and large difference between the measured forces f and the force values $\hat{f} = kx + b\dot{x}$. These results showed a poor linear relationship between those quantities and a single linear model that would describe the repeated experiments could not be found.

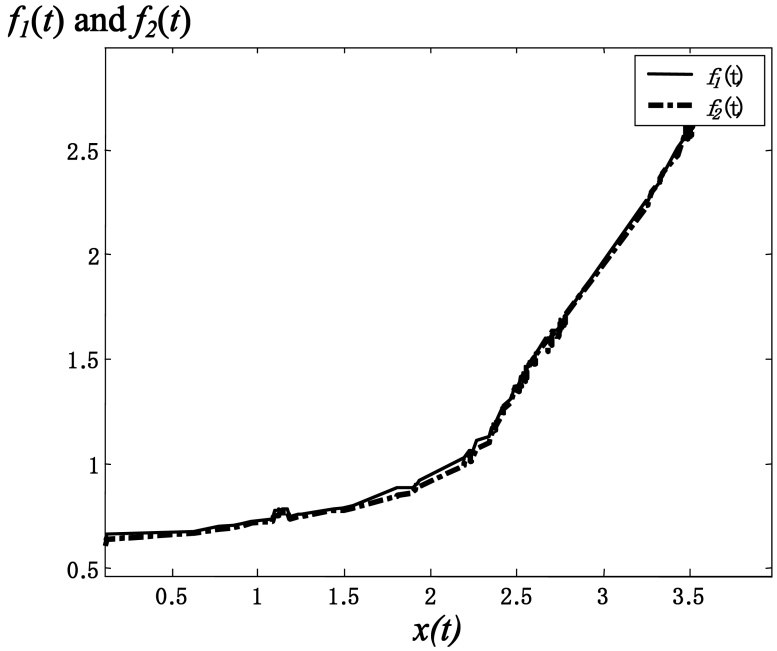


Fig. 6. Force signals versus displacement sensors at the time instants when the pedal is stationary.

Thus, a linear model for our spring and damper is not sufficient and their nonlinearity should also be taken into account. We examined a generalised version of the above linear model (3):

$$f = g_1(x) + g_2(\dot{x}). \tag{4}$$

In order to find the proper mathematical form of g_1, we examined the recorded force and displacement data for the stationary pedal, i.e., the data samples with almost zero velocity. Fig. 6 shows the force versus displacement plotted at the time instants when the pedal is stationary. The very close distance between the two static force signals confirms our assumption on negligibility of the spring and damper masses. Fig. 6 also shows that $g_1(x)$ can be properly modelled by a quadratic polynomial:

$$\hat{f}|_{dx/dt=0} = Ax^2 + Bx + C. \tag{5}$$

This model complies with the fact that the spring force substantially increases when it is compressed beyond a linear region. Using the recorded static data, we achieved a LS estimate for the parameters A, B and C in (5).

For the function g_2 in (4), another quadratic model was chosen and its parameters were also estimated by the LS technique. The viscous friction substantially increases when the pedal speed rises beyond the linear damper model, and this phenomenon is actually realised by the quadratic model for g_2. The models used for conversion of displacement measurements to equivalent force values are presented as follows:

$$\hat{f}_1 = A_1x^2 + B_1x + C_1 + D_1\dot{x}^2 + E_1\dot{x} \tag{6}$$

$$\hat{f}_2 = A_2x^2 + B_2x + C_2 + D_2\dot{x}^2 + E_2\dot{x}. \tag{7}$$

The LS estimates of A_1 and A_2 are very close to each other, and so are B_1 and B_2, C_1 and C_2, D_1 and D_2, and E_1 and E_2. This validates our assumption on negligibility of the effect of pedal mass and the sufficiency of a first order dynamic model. As shown in Fig. 4, after using the quadratic models, shown in (6)–(7), with their estimated parameters to convert the displacement sensor output to their equivalent force signals, the four signals f_1, f_2, $\hat{f}_1$ and $\hat{f}_2$ can now be utilised to evaluate the sensors agreement by calculating $\alpha_{1,2}$, $\alpha_{1,3}$ and $\alpha_{2,3}$ values. More precisely, the internal representation of signals in Fig. 4 is the "force" quantity and f_1 and f_2 are same as x_1 and x_2 in Fig. 2. Since the displacement measurement x is converted to two estimates $\hat{f}_1$ and $\hat{f}_2$ (to be compared with f_1 and f_2), x_3 in Fig. 2 has two corresponding signals in Fig. 4: $\hat{f}_1$ and $\hat{f}_2$.

These values along with the forces and converted signals are then given to a fuzzy system where the agreement values are fuzzified. Fig. 7 shows the definitions of the fuzzy sets for fuzzification of agreement evaluations. Because of the conversion errors, α-coordinates of the break-points of the piece-wise linear membership functions for $\alpha_{2,3}$ and $\alpha_{1,3}$ are higher than the α-coordinates of the break-points for $\alpha_{1,2}$. Since a lower $\alpha_{i,j}$ value means stronger agreement between S_i and S_j, the Large and Small fuzzy sets are associated with lower and higher $\alpha_{i,j}$ values, respectively. The resulting membership values are then used by a fuzzy rule-base for fuzzy inference. In our case study, the rule-base contains seven fuzzy rules as shown in Table I. The third fuzzy rule is the same rule stated before in Section 2. Based on the details given in Table I, the final fused value for the driver's brake demand is computed by (2)

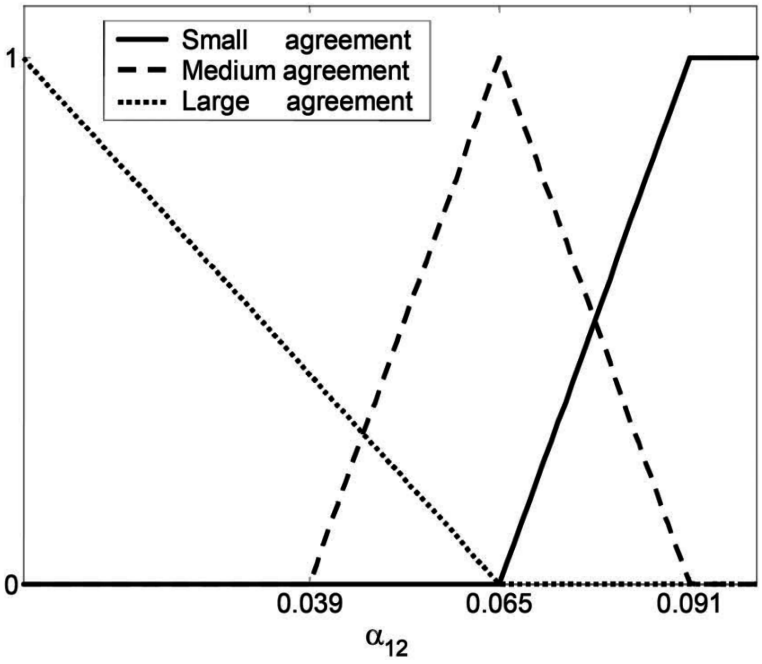


Fig. 7. Definition of three fuzzy sets for fuzzification of $S_1 - S_2$ agreement evaluation: Similar definitions apply to fuzzification of agreement evaluations of $S_1 - S_3$ and $S_2 - S_3$, however due to conversion errors the α-coordinates of the break-points $\{0.039, 0.065, 0.091\}$ change to higher values of $\{0.52, 0.78, 1.04\}$.

TABLE I
The Fuzzy Rule-Base Utilised in for Sensor Fusion in our Experiments with the Brake-by-Wire Pedal (L = Large, M = Medium, S = Small)

i	α_{12}	α_{23}	α_{13}	O_i	S_1 Faultiness	S_2 Faultiness	S_3 Faultiness
1	L	L	L	$\frac{f_1 + \hat{f}_1 + f_2 + \hat{f}_2}{4}$	Small	Small	Small
2	L	S	S	$\frac{f_1 + f_2}{2}$	Small	Small	Large
3	S	L	S	$\frac{f_2 + \hat{f}_2}{2}$	Large	Small	Small
4	S	S	L	$\frac{f_1 + \hat{f}_1}{2}$	Small	Large	Small
5	L	M	M	$\frac{f_1 + f_2}{2}$	Small	Small	Medium
6	M	L	M	$\frac{f_2 + \hat{f}_2}{2}$	Medium	Small	Small
7	M	M	L	$\frac{f_1 + \hat{f}_1}{2}$	Small	Medium	Small

with O_i and w_i given as below:

$$
\begin{aligned}
w_1 &= \mu_L(\alpha_{12})\mu_L(\alpha_{23})\mu_L(\alpha_{13}), & O_1 &= (f_1 + \hat{f}_1 + f_2 + \hat{f}_2)/4 \\
w_2 &= \mu_L(\alpha_{12})\mu_S(\alpha_{23})\mu_S(\alpha_{13}), & O_2 &= (f_1 + f_2)/2 \\
w_3 &= \mu_S(\alpha_{12})\mu_L(\alpha_{23})\mu_S(\alpha_{13}), & O_3 &= (f_2 + \hat{f}_2)/2 \\
w_4 &= \mu_S(\alpha_{12})\mu_S(\alpha_{23})\mu_L(\alpha_{13}), & O_4 &= (f_1 + \hat{f}_1)/2 \\
w_5 &= \mu_L(\alpha_{12})\mu_M(\alpha_{23})\mu_M(\alpha_{13}), & O_5 &= (f_1 + f_2)/2 \\
w_6 &= \mu_M(\alpha_{12})\mu_L(\alpha_{23})\mu_M(\alpha_{13}), & O_6 &= (f_2 + \hat{f}_2)/2 \\
w_7 &= \mu_M(\alpha_{12})\mu_M(\alpha_{23})\mu_L(\alpha_{13}), & O_7 &= (f_1 + \hat{f}_1)/2.
\end{aligned}
\tag{8}
$$

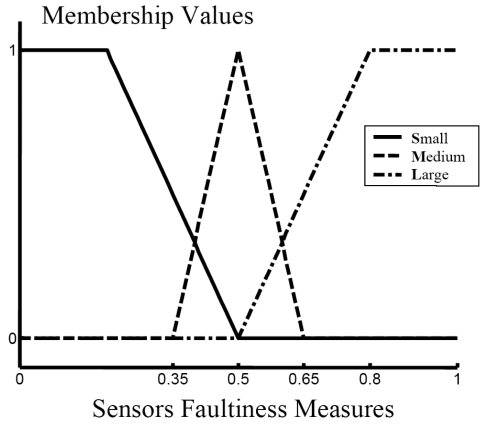


Fig. 8. Fuzzy sets definition for defuzzification of sensor faultiness measures.

In the consequences of the rules, the faultiness measures belong to one of the Small, Medium or Large fuzzy sets with piece-wise linear membership functions as shown in Fig. 8. The resulting faultiness measures are defuzzified by the fuzzy centroid method.

In our validation experiments, we applied different types of brake commands in various conditions such as a continuous panic brake, short-time panic brakes, short-time soft brakes, a continuous soft brake and so on. Total length of each experiment was 110 s. Fig. 9 shows the signals of the three sensors recorded during the validation experiments. S_1 and S_2 signals (pedal force measurements) are very close to each other and one of them is shown in Fig. 9. In this figure and the next signal plots, the vertical coordinate units are "volt," as the filtered "electrical" measurement signals and their fused measures have been plotted and all

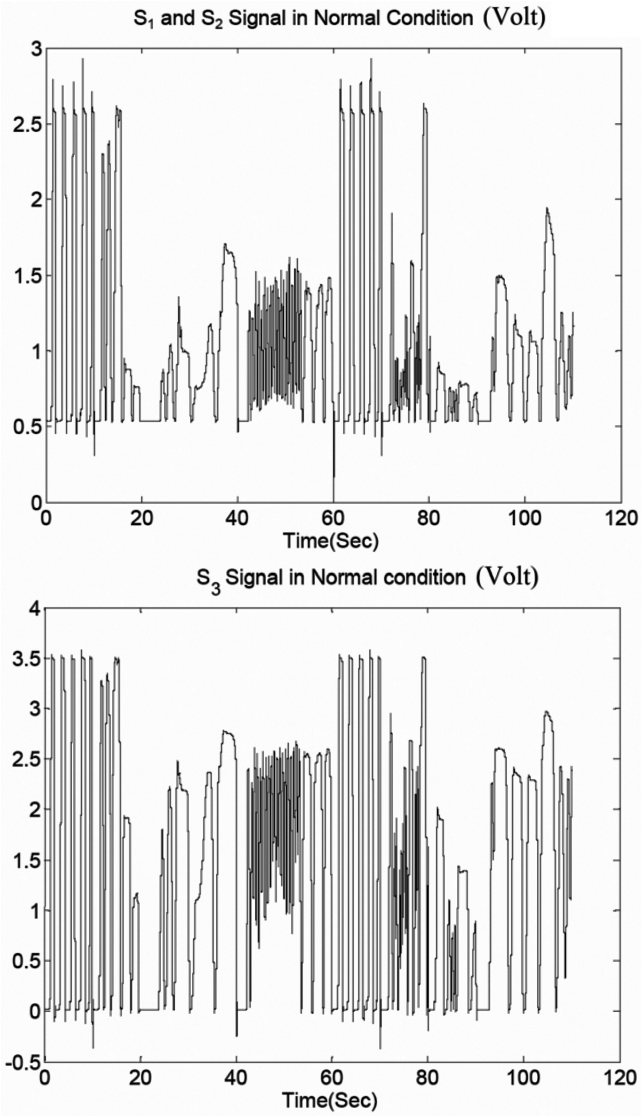


Fig. 9. Recorded sensory signals in normal (no fault) condition.

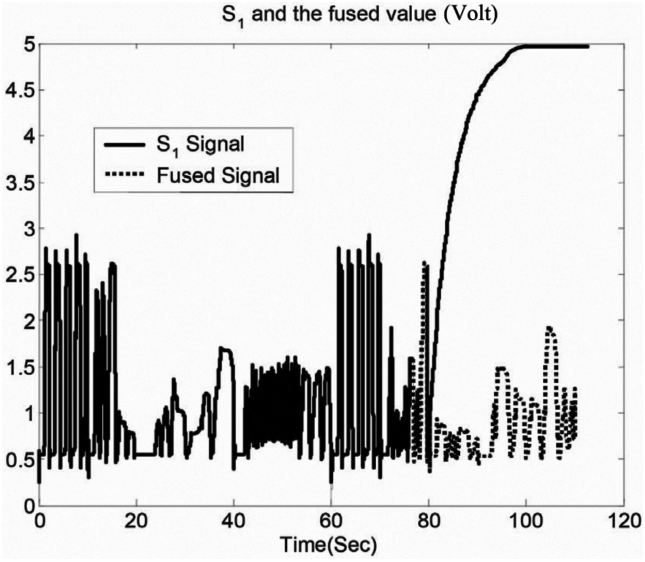


Fig. 10. Soft voting result when S_1 is short circuit and gradually rises toward supply voltage.

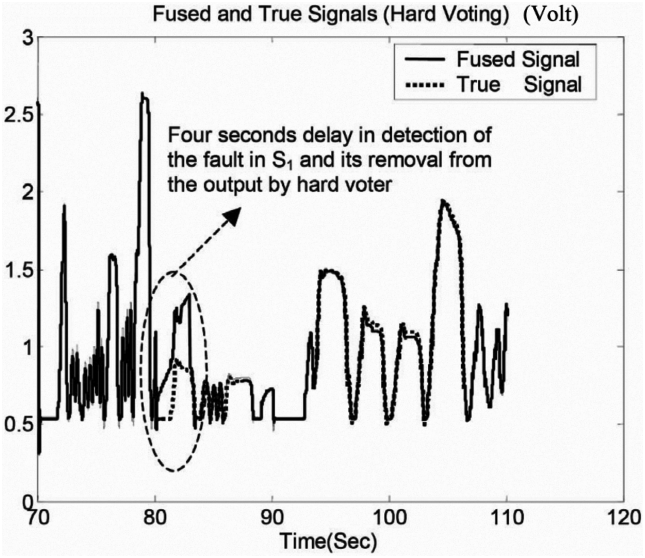


Fig. 11. Hard voting result when S_1 is short circuit and gradually moves toward supply voltage.

of them are proportional to the internal representation quantity (force) with a constant factor. We then injected several types of synthetic faults into S_1 during the time interval [80,110] and used both the hard and the soft (fuzzy) voting methods to fuse the sensor data. Fig. 10 shows the results when the S_1 signal is short-circuited to supply. Because of the RC circuitry connected to the input of analogue to digital converters (ADCs) the S_1 signal does not suddenly jump to the supply voltage, but rises gradually. Soft voting detects the fault and removes the S_1 signal from voting process in a timely manner. We also applied hard voting to detect the same fault. Fig. 11 shows the fused signal and its expected true values in the time interval, starting 10 s before the short circuit event. It is observed that the short circuit is detected by hard voting after four seconds as the short circuit starts at $t = 80$ s but the deviation of the fused signal from the true signal returns to almost zero at $t = 84$ s. During these four seconds the hard voter provides a wrong fused measurement. This is fairly dangerous and unacceptable in a brake-by-wire application.

Pedal sensors data may also drift due to temperature variations during motor warm-up or cool-down periods. Fig. 12 shows a linear drift of 1000 mV injected into S_1 and the result of soft voting by which the drift is detected and removed. On the other hand, the hard voting method does not detect the drift, because the threshold of agreement evaluation is larger than the 1000 mV drift. Hard voting result is presented in Fig. 13. Faultiness measures resulted from soft voting in the presence of the linear drift in S_1 are also shown in Fig. 14. It is observed that faultiness for S_1 is always large and faultiness for S_2 and S_3 are initially large but decrease while the drift in S_1 grows. To examine the performance of the proposed technique for a noisy signal, excessive noise

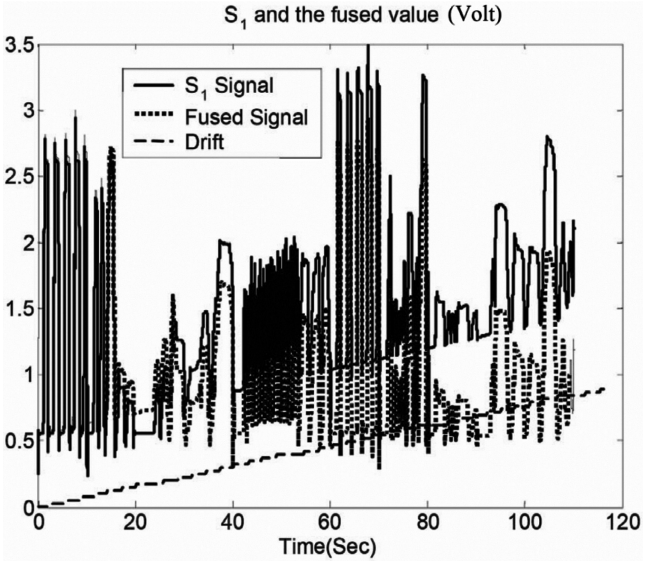


Fig. 12. Soft voting result when there is a linear drift in S_1.

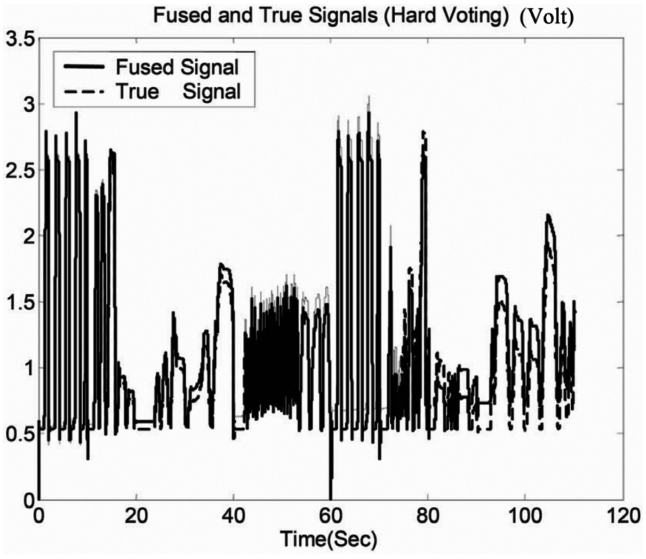


Fig. 13. Hard voting result when there is a linear drift in S_1.

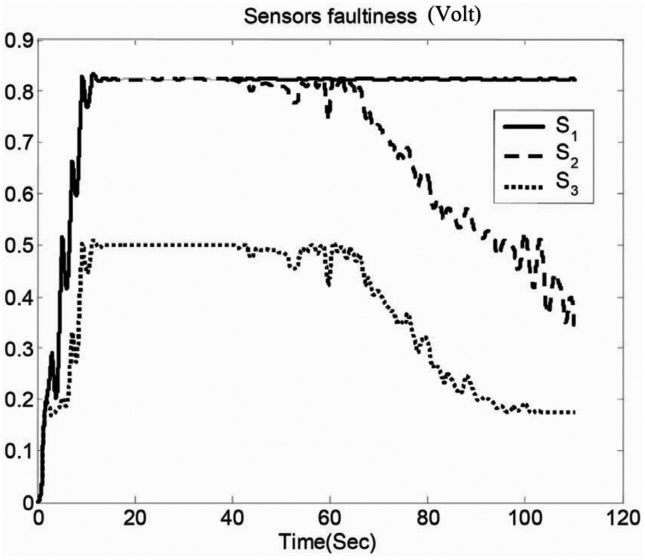


Fig. 14. Faultiness measures resulted by soft voting result in presence of a linear drift in S_1.

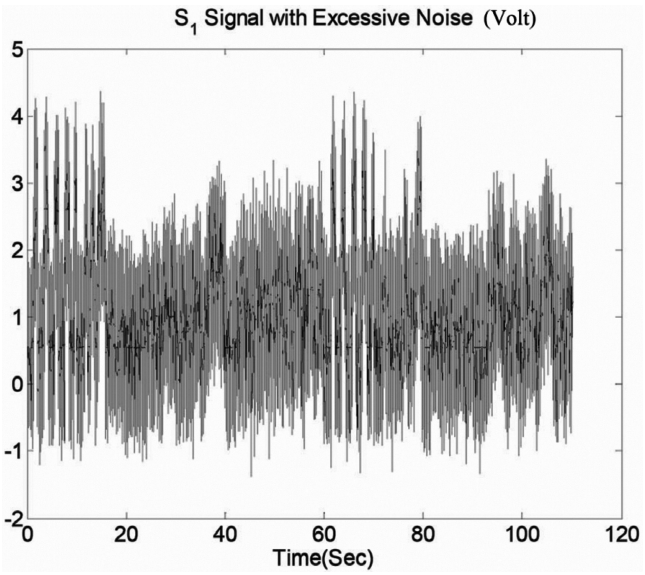


Fig. 15. S_1 signal in presence of excessive noise.

was injected into the S_1 signal as depicted in Fig. 15. As shown in Fig. 16, soft voting has been able to effectively detect and remove the noise from sensor fusion output, and Fig. 17 shows that hard voting can not substantially reduce the noise.

In order to compare the performance of the majority (hard) voting method with our proposed soft voting method quantitatively, we computed the mean square error (MSE) for soft and hard voting methods in the presence of various faults. Table II shows the result of our error computation. Overall, the MSE was reduced by 82% in soft voting compared to hard voting. That is because of the early fault detection and removal capability of the soft voter. Finally, it should be noted that our proposed method is a voting method, i.e., we do not expect it to detect a fault if it exists in the majority of sensors (two or more sensors in our case study). For example if a short circuit happens for both S_1 and S_2, then both the hard and the soft voter will incorrectly deduce that S_3 is faulty because it does not agree with the other two sensors.

4. CONCLUSIONS

In this paper, we introduced a new method for fusion of redundant sensory information in fault tolerant systems with focus on a by-wire braking system. We applied our method to fuse the redundant data provided by two force sensors and one displacement sensor in a by-wire brake pedal. Because of the sensor conversion errors, sensor agreement thresholds in a majority voter are so large that an unacceptable delay in fault detection occurs. Our proposed soft voting method applies a fuzzy

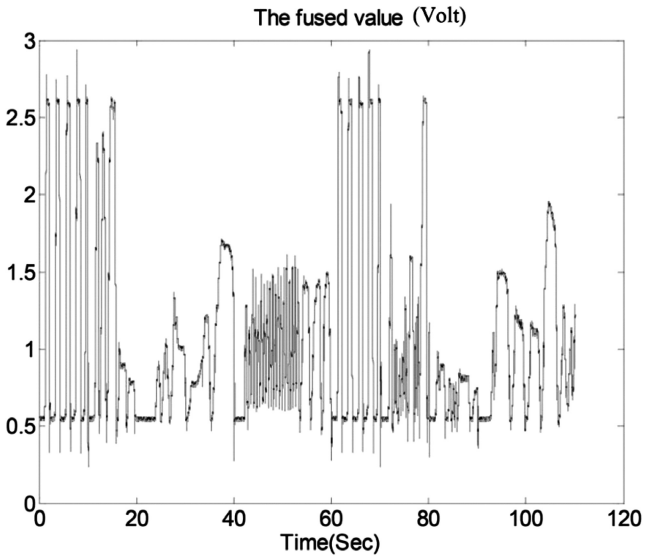


Fig. 16. Soft voting result when in presence of excessive noise in S_1.

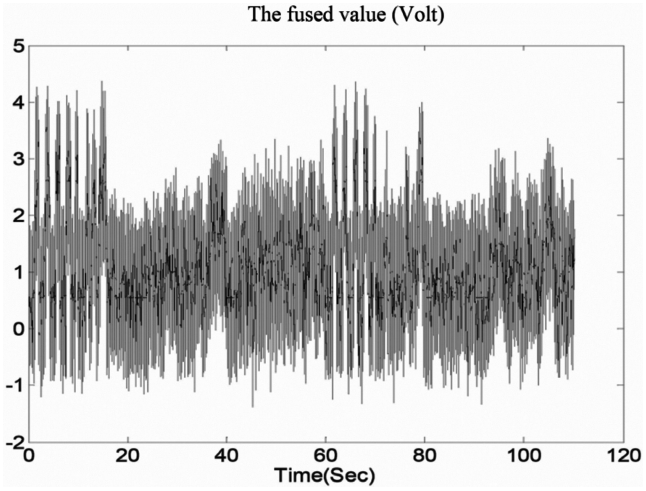


Fig. 17. Hard voting result when in presence of excessive noise in S_1.

rule-base to perform voting. The fuzzy rules here are designed in such a way that the voter output is smoothly switched from one majority voted value to another in case of a sensor fault. The proposed soft voter also gives faultiness measures for all sensors.

The novel idea in our approach is that we calculate the averaging weights as a normalised sum of products of membership values. The implementation of the proposed technique is straightforward and its execution is time efficient. As such, it is an appropriate solution for real-time and safety critical applications such as brake-by-wire, where computational load and memory requirements as well as convergence and stability are important issues. Experimental results show that our proposed method is successful in fault detection for cases where a majority voting approach either results in late detection or fails completely. Experiments also show that the soft voting total error (in terms of MSE) is reduced by around 82% compared to a hard voting technique.

TABLE II
MSE Error for Pedal Sensor Fusion by Soft and Hard Voting in Presence of Various Faults

Injected Fault	Hard Voter	Soft Voter
Gradually Short to Ground	0.1932	0.0367
Gradually Short to Supply	0.1033	0.0272
Suddenly Short to Ground	0.2123	0.0298
Suddenly Short to Supply	0.2099	0.0245
Noise (Substantial SNR)	0.1277	0.0434
Drift	0.2108	0.0269
Total MSE	1.0572	0.1885

REFERENCES

[1] A. Avizienis
The N-version approach to fault-tolerant software.
IEEE Transactions on Software Engineering, **1** (1985), 1491–1501.

[2] J. M. Bass, G. R. Latif-Shabgahi and S. Bennett
Experimental comparison of voting algorithms in cases of disagreement.
In *Proceedings of 23rd Euromicro Conference*, Budapest, Hungary, 1997, 516–523.

[3] D. Baum, C. D. Hamann and E. Schubert
High performance ACC system based on sensor fusion with distance sensor, image processing unit, and navigation system.
Vehicle System Dynamics, **28**, 6 (1997), 327–338.

[4] N. E. Belabbes, A. J. Guterman, Y. Savaria and M. Dagenais
Ratioed voter circuit for testing and fault-tolerance in VLSI processing arrays.
IEEE Transactions on Circuits and Systems I—Fundamental Theory and Applications, **43**, 2 (1996), 143–152.

[5] A. Bogliolo, M. Favalli and M. Damiani
Enabling testability of fault-tolerant circuits by means of I-DDQ-checkable voters.
IEEE Transactions on VLSI Systems, **8**, 4 (2000), 415–419.

[6] S. Dajani-Brown, D. Cofer and A. Bouali
Formal verification of an avionics sensor voter using SCADE.
In *Proceedings of Lecture Notes in Computer Science*, 3253 (2004), 5–20.

[7] S. Dajani-Brown, D. Cofer, G. Hartmann and S. Pratt
Formal modeling and analysis of an avionics triplex sensor voter.
In *Proceedings of Lecture Notes in Computer Science*, 2648 (2003), 34–48.

[8] M. Favalli and C. Metra
TMR voting in the presence of crosstalk faults at the voter inputs.
IEEE Transactions on Reliability, **53**, 3 (2004), 342–348.

[9] T. A. Johansen, I. Petersen, J. Kalkkuhl and J. Ludemann
Gain-scheduled wheel slip control in automotive brake systems.
IEEE Transactions on Control System Technology, **11**, 6 (Nov. 2003), 799–811.

[10] R. Hoseinnezhad and A. Bab-Hadiashar
Missing data compensation for safety-critical components in a drive-by-wire system.
IEEE Transactions on Vehicular Technology, **54**, 4 (July 2005), 1304–1311.

[11] R. Hoseinnezhad, A. Bab-Hadiashar and P. Harding
Missing data handling by a multi-step ahead predictive filter.
In *Proceedings of International Conference on Computational Intelligence for Modelling Control and Automation* CIMCA'2004, Gold Coast, Australia, 2004, 991–999.

[12] R. Hoseinnezhad
Position sensing in brake-by-wire callipers using resolvers.
IEEE Transactions on Vehicular Technology, **55**, 3 (May 2006), 924–932.

[13] H. Kim, H. J. Jeon, H. Lee and H. Lee
The design and evaluation of all voting triple modular redundancy system.
In *Proceedings of Annual Reliability and Maintainability Symposium*, Seattle, WA, 2002, 439–444.

[14] G. R. Latif-Shabgahi
A novel algorithm for weighted average voting used in fault tolerant computing systems.
Microprocessors and Microsystems, **28**, 7 (2004), 357–361.

[15] G. R. Latif-Shabgahi, S. Bennett and J. M. Bass
Smoothing voter: A novel voting algorithm for handling multiple errors in fault-tolerant control systems.
Microprocessors and Microsystems, 27 (2003), 303–313.

[16] B. Parhami
Voting algorithms.
IEEE Transaction on Reliability, **43**, 4 (1994), 617–629.

[17] K. Park and S. J. Heo
A study on the brake-by-wire system using hardware-in-the-loop simulation.
International Journal of Vehicle Design, **36**, 1 (2004), 38–49.

[18] J. M. Quintana, M. J. Avedillo and J. L. Huertas
Efficient realization of a threshold voter for self-purging redundancy.
Journal of Electronic Testing—Theory and Applications, **17**, 1 (2001), 69–73.

[19] F. Rothlauf
Population sizing for the redundant trivial voting mapping.
In *Proceedings of Lecture Notes in Computer Science*, 2724 (2003), 1307–1319.

[20] F. Tahami, R. Kazemi and S. Farhangi
A novel driver assist stability system for all-wheel-drive electric vehicles.
IEEE Transactions on Vehicular Technology, **52**, 3 (May 2003), 683–692.

[21] Z. Tong and R. Kain
Vote assignments in weighted voting mechanisms.
In *Proceedings of the Seventh Symposium on Reliable Distributed Systems*, Columbus, OH, 1988, 138–143.

Reza Hoseinnezhad was born in Tehran, Iran, in 1973. He received his B.E., M.E., and Ph.D. degrees from the University of Tehran, Iran, in 1994, 1996, and 2002, all in electrical engineering. From 2002 to 2003, he was an assistant professor at the University of Tehran. From July 2003 to October 2005, he was a postdoctoral research fellow, and since October 2005 he has been a senior research fellow, at Swinburne University of Technology, Victoria, Australia.

Dr. Hoseinnezhad's research is focused on new methodologies for signal processing and sensor data fusion techniques applied in drive-by-wire systems, and on the analysis of robust estimation techniques in computer vision. He has two international patents on brake-by-wire systems.

Alireza Bab-Hadiashar was born in Iran, in 1964. He received his B.E., M.E., and Ph.D. degrees in 1988 from the University of Tehran, Iran, in 1994 from the University of Sydney, Australia, and in 1997 from the Monash University, Australia, respectively. Since 1997, he has held various academic positions at both Monash University and the Swinburne University of Technology (Australia) where he is currently an associate professor and program manager of robotics and mechatronics. His research interest is the development of robust data analysis techniques for engineering applications in general and computer vision in particular.

Augmented State Integrated Probabilistic Data Association Smoothing for Automatic Track Initiation in Clutter

RAJIB CHAKRAVORTY

SUBHASH CHALLA

Networked Sensor Technology Laboratory

We introduce a fixed lag smoother algorithm based on the integrated probabilistic data association (IPDA) algorithm. IPDA jointly estimates both the target state and its existence. In this paper the joint density of target state and existence is extended for fixed lag smoothing. The proposed smoothing algorithm is also tested against various multiple target tracking parameters like state RMS estimation, number of true target detected, number of false target confirmed and target termination time and simulation results are also presented in the paper.

Manuscript received November 7, 2004; revised July 27, 2005 and January 30, 2006.

Refereeing of this contribution was handled by Dr. Jean Dezert.

The authors would also like to acknowledge the contribution of Center for Autonomous Systems (CAS), UTS for its contribution towards the reasearch work.

Authors' address: Networked Sensor Technology (NeST) Laboratory, Faculty of Engineering, University of Technology, Sydney, Broadway-2007, Sydney, NSW, E-mail: (rchakra@eng.uts.edu.au, schalla@eng.uts.edu.au).

1. INTRODUCTION

Smoothing within the state estimation context is technically defined as a process where the current measurements are used to improve the estimates of the past states of the object of interest. In the target tracking problem, this corresponds to estimating the past target states and associated tracker performance parameters using current measurements. Formally, one can define the track estimation problem as follows. Let $x(t_k)$ denote the target state at time k and $y^k = \{y(1), y(2), \ldots, y(k)\}$ denote the measurement sequence up to time k where $y(i)$ denotes the measurement at time i. The target estimation problem can then be defined as the problem of computing the conditional mean estimate of the target state

$$\hat{x}(t_{k-L|k}) = E[x(t_{k-L}) \mid y^k] \tag{1}$$

and its associated error covariance

$$P_{k-L|k} = E[(x(t_{k-L}) - \hat{x}(t_{k-L|k}))(x(t_k) - \hat{x}(t_{k-L|k}))^T \mid y^k] \tag{2}$$

where $L = 0$, $L < 0$ and $L > 0$ are for three types of estimation namely filtering, prediction and smoothing respectively.

Smoothing algorithms were shown to provide significant performance improvements in terms of RMS errors in several important tracking problems like maneuvering target tracking using IMM smoothers by Helmick et al. [13]–[14], tracking in clutter using PDA smoothers by Mahalanabis et el. [17] and using IMMPDA smoothing for maneuvering target tracking in clutter [12]. More recently, the augmented state smoothing framework was used for dealing with out of sequence measurements by Challa et al. [11], [25].

One of the very important assumptions made in all these efforts is the fact that the target exists. However, in reality, target existence must first be established before using one of the above methods. Several techniques to achieve this are available in literature, like the heuristic M out of N detections method [6] and the Bayesian approaches like IMMPDA and IPDA.

IMMPDA, when used in the context of automatic track formation [2], uses two models—one that assumes that the target is "observable" by the sensor with a detection probability of P_D, $0 < P_D \leq 1$, and the other assumes that the target is not "observable" with probability of detection $P_D = 0$. The algorithm uses a probability measure (the model probability) for each of the models and estimates the "true target probability." If that probability crosses a suitable threshold, a decision on target's presence (existence) is taken.

Another effective algorithm to solve automatic track formation in clutter, referred to as Integrated Probabilistic Data Association (IPDA), is proposed by Musicki and Evans [22]–[23]. Many of its variants for use in difficult environments are proposed in [18], [21]. IPDA

models target existence as a random event satisfying Markovian properties between existence and nonexistence states and provides a mechanism to associate a probability measure (the target existence probability) to it. Similarly to the method of IMMPDA, the target existence probability is estimated along with the target states and if the existence probability crosses a threshold, a decision on target's presence (existence) is taken.

IMMPDA and IPDA are Bayesian approaches and are amenable for treatment within the smoothing framework. In this paper, we focus on the IPDA based approach for target existence and develop a new algorithm for automatic track initiation in clutter within the augmented state smoothing framework. We investigate the effect of smoothed track existence probability on tracker performance measures, e.g. true/false track discrimination, by comparing its performance with the standard IPDA algorithm. The flow chart of the algorithm is also presented in this paper. Simulation results are also provided, where the improvements in true/false track statistics are found to be significant with a potential to improve all higher layer functions of tracking systems like situation and threat assessment.

The paper is organized as follows. Following the introduction, Section 2 formulates the automatic track formulation problem as conceptualized by IPDA. The theory of the augmented state IPDA smoother is described in Section 3. The complete mathematical derivation of the algorithm is carried out in Section 4. The flow chart of the algorithm is presented in Section 5. The simulation scenario and results are presented in Section 6. Conclusions are drawn in Section 7.

2. PROBLEM FORMULATION

The target tracking algorithm starts with a priori knowledge of the target dynamic model. Each target within the surveillance region is assumed to follow the dynamic equation

$$x_k = Fx_{k-1} + Gw_k \tag{3}$$

where

- the target state x_k consists of kinematic states e.g. position, velocity etc.,
- F is the state transition matrix,
- w_k is the noise (called "process noise"). w_k is assumed normally distributed with mean zero and variance Q. It is also assumed that $E\{w_k w_j\} = 0$ if $k \neq j$.

A measurement model is defined as

$$y_k = Hx_k + v_k \tag{4}$$

where

- H is the state to measurement transition matrix,
- v_k is noise (called "measurement noise"). v_k is assumed to be normally distributed with mean zero and variance R. It is also assumed that $E\{v_k v_j\} = 0$ if $k \neq j$. Moreover $E\{w_i v_j\} = 0$ for any i, j.

IPDA takes "track existence" as a random event and finds the probability of the event to solve the problem of "automatic track maintenance." IPDA models the existence of a track as a two state random variable, E_k, where

- $E_k = 1$ refers to the event that the track exists at time $t = k$,
- $E_k = 0$ refers to the event that the track does not exist at time $t = k$.

A target or track can also switch between these two states according to a predefined switching probability matrix which is

$$\Gamma = \begin{bmatrix} \Gamma_{11} & \Gamma_{10} \\ \Gamma_{01} & \Gamma_{00} \end{bmatrix} \tag{5}$$

where

$$\Gamma_{ij} = p(E_k = j \mid E_{k-1} = i), \qquad i, j \in 0, 1. \tag{6}$$

In the rest of the text, $E_k = 1$ will be denoted as E_k and $E_k = 0$ will be denoted as $\bar{E}_k$ with the definition of Γ_{ij} considered to be understood as defined by (6).

IPDA solves the uncertainty in "target existence" automatically by estimating

$$p(E_{k-L} \mid y^k) \tag{7}$$

where again $L = 0$, $L < 0$ and $L > 0$ are for filtering, prediction and smoothing types of estimation respectively. The state is estimated with the condition that the target exists and thus the state estimation is redefined by the introduction of a conditional parameter as

$$p(x_{k-L} \mid E_{k-L}, y^k). \tag{8}$$

3. FIXED LAG AUGMENTED STATE IPDA (AS-IPDA) SMOOTHING

In an augmented approach for a lag of N, the target dynamic model and measurement equation of (3) and (4) respectively will be replaced by the augmented vectors

$$\mathcal{X}_k = \mathcal{F}\mathcal{X}_{k-1} + \mathcal{G}w_k \tag{9}$$

$$\mathcal{Y}_k = \mathcal{H}\mathcal{X}_k + v_k \tag{10}$$

where

$$\mathcal{X}_k = [x_k \;\; x_{k-1} \cdots x_{k-N}]^T \tag{11}$$

$$\mathcal{F} = \begin{bmatrix} F & 0 & 0 & \cdots & 0 \\ I & 0 & 0 & \cdots & 0 \\ 0 & I & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \cdots & \vdots \\ 0 & 0 & \cdots & I & 0 \end{bmatrix} \tag{12}$$

$$\mathcal{Y}_k = [y_k] \tag{13}$$

$$\mathcal{H} = [H \;\; 0 \cdots 0]. \tag{14}$$

The noise variance matrix Q_k will also be adjusted to

$$\mathcal{Q} = \begin{bmatrix} Q & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}. \tag{15}$$

But the IPDA concept suggests that along with the target's state, its existence event also needs to be augmented. Based on that conceptual framework, two possible combinations of target state and its existence are possible at any given instant of time. These are:

- $C_k^1 = \{x_k, E_k\}$, the target exists and so does its state,
- $C_k^2 = \{\phi, \bar{E}_k\}$, the target does not exist and so neither does its state.

Thus for an entire fixed lag of N, the augmentation can be carried out in the following manner

$$[C_k^a \;\; C_{k-1}^b \cdots C_{k-N}^d]^T \tag{16}$$

where $a, b, \ldots, d \in 1, 2$. This suggests that there can be more than one augmented hypothesis possible.

From the implementation point of view of IPDA, when a target goes out of existence, it remains that way for all future time. In that context, the transition matrix of (5) can be made more specific

$$\Gamma = \begin{bmatrix} \Gamma_{11} & \Gamma_{10} \\ 0 & 1 \end{bmatrix}. \tag{17}$$

All published results of IPDA follow this transition matrix.

Therefore not all the combinations of C_ks are valid in (16). Thus at any time, eliminating the impossible hypotheses, there remain $N + 2$ permissible augmented hypotheses. These are

- Hypothesis 1:

$$\mathbf{H}_k = [\mathcal{X}_k, \mathcal{E}_k] = \begin{bmatrix} x_k, E_k \\ x_{k-1}, E_{k-1} \\ \vdots \\ x_{k-N}, E_{k-N} \end{bmatrix} \tag{18}$$

- Hypothesis m:

$$\mathbf{H}_k^m = [\mathcal{X}_k^m, \mathcal{E}_k^m] = \begin{bmatrix} \phi, \bar{E}_k \\ \vdots \\ \phi, \bar{E}_{k-m} \\ x_{k-m-1}, E_{k-m-1} \\ \vdots \\ x_{k-N}, E_{k-N} \end{bmatrix} \tag{19}$$

where $m = 0, 1, 2, \ldots, N-1$.

- Hypothesis n:

$$\mathbf{H}_k^n = [\mathcal{X}_k^n, \mathcal{E}_k^n] = \begin{bmatrix} \phi, \bar{E}_k \\ \vdots \\ \phi, \bar{E}_{k-N} \end{bmatrix}. \tag{20}$$

Except for hypothesis one, the other hypotheses assume that the target does not exist at the current time. It is also shown in the Appendix that hypotheses $\mathbf{H}_k^m$ and $\mathbf{H}_k^n$ also do not contribute in the state update. This is also supported by the fact that as these hypotheses refer to the scenario that at the current time the target does not exist, measurements at the current time will therefore contain no information about the target. Thus the smoothing of a track is concerned only with the first hypothesis.

Thus the underlying Bayesian approach for developing an IPDA smoothing algorithm reduces to the calculation of the probability density,

$$p(\mathcal{X}_k, \mathcal{E}_k \mid y^k) = p(\mathcal{X}_k \mid \mathcal{E}_k, y^k) p(\mathcal{E}_k \mid y^k). \tag{21}$$

Furthermore the existence probabilities at each time instant are readily given by,

$$p(E_k) = p(\mathcal{E}_k \mid y^k) \tag{22}$$

$$p(E_{k-d}) = p(\mathcal{E}_k \mid y^k) + \sum_{j=0}^{d-1} p(\mathcal{E}_k^j \mid y^k) \tag{23}$$

where $d = 1, 2, \ldots, N$.

The conditional state estimate of (21) and the existence probabilities of (22) and (23) together solve the IPDA smoothing problem. In the next section, the calculation of these estimates will be carried out.

4. DERIVATION OF AS-IPDA SMOOTHING

In this section, one iteration of the state estimate and existence probability estimates are derived separately for clarity.

4.1 Conditional State Estimate

The conditional state estimate $p(\mathcal{X}_k \mid \mathcal{E}_k, y^k)$ can be expanded through Bayes' Theorem:

$$\begin{aligned} p(\mathcal{X}_k \mid \mathcal{E}_k, y^k) &= p(\mathcal{X}_k \mid \mathcal{E}_k, y_k, y^{k-1}) \\ &= \frac{p(y_k \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}).p(\mathcal{X}_k \mid \mathcal{E}_k, y^{k-1})}{p(y_k \mid \mathcal{E}_k, y^{k-1})} \\ &= \frac{\text{Likelihood} \;\times\; \text{Prediction}}{\text{Normalization}}. \end{aligned} \tag{24}$$

The three terms likelihood, prediction and normalization will be derived step by step. The a priori target state is assumed known (either from a previous iteration or from initialization of the tracks) with a Gaussian distribution having mean $\hat{\mathcal{X}}_{k-1|k-1}$ and covariance $\mathcal{P}_{k-1|k-1}$.

Likelihood

As there is uncertainty involved about the origin of measurements, the measurements that fall within a validating gate are used for the track state update. The volume of the elliptical validation region is V_k.

To calculate the likelihood, the assumptions made in the literature are

- the probability mass function of the number of false measurements conditioned on the past measurement history at time k is a Poisson distribution,

$$P(m_k \mid y^{k-1}) \equiv P_0(m_k) = \frac{\lambda^{m_k} e^{-\lambda}}{m_k!} \tag{25}$$

where λ is the expected number of validated measurements and is given by

$$\lambda = \begin{cases} 0 & \text{if} \quad m_k = 0 \\ m_k - P_D P_G p(E_k \mid y^{k-1}) & \text{if} \quad m_k > 0 \end{cases} \tag{26}$$

where m_k is the number of validated measurements at the time $t = k$. P_D and P_G denote the detection probability and gate probability respectively while $p(E_k \mid y^{k-1})$ can be obtained from $p(E_{k-1} \mid y^{k-1})$ by using the Markov Transition Probability as

$$p(E_k \mid y^{k-1}) = \Gamma_{11} p(E_{k-1} \mid y^{k-1}) \tag{27}$$

- the hypotheses that
  —α_0: all validated measurements are false measurement or clutter
  —α_i: the ith validated measurement is target originated and all others are false measurements
  These are complementary sets and hence the following conditional probabilities can be defined
  1) No validated measurement is target originated given the target exists

$$P(\alpha_0 \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) = 1 - P_D P_G. \tag{28}$$

  2) The ith validated measurement is target originated given the target exists

$$P(\alpha_i \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) = \frac{P_D P_G}{m_k}. \tag{29}$$

Based on the above defined parameters, the likelihood in (24) is calculated as

$$\begin{aligned} & p(y_k \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) \\ &= \sum_{i=0}^{m_k} p(y_k \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}, y^{k-1}, \alpha_i, m_k) \\ &\quad .P(m_k \mid y^{k-1}).P(\alpha_i \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) \\ &= \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) P(\alpha_0 \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) \\ &\quad + \sum_{i=1}^{m_k} \left(\frac{1}{V_k}\right)^{m_k - 1} P_0(m_k - 1) \times p(y_k(i) \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}, \alpha_i) \\ &\quad \times P(\alpha_i \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) \\ &= \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) P(\alpha_0 \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) \\ &\quad + \left(\frac{1}{V_k}\right)^{m_k - 1} \frac{m_k}{\lambda} P_0(m_k) \sum_{i=1}^{m_k} p(y_k(i) \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}, \alpha_i) \\ &\quad .P(\alpha_i \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) \\ &= \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) \\ &\quad \times \left\{ 1 - P_D P_G + P_D P_G \frac{V_k}{\lambda} \sum_{i=1}^{m_k} p(y_k(i) \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}, \alpha_i) \right\} \end{aligned} \tag{30}$$

where V_k is the volume of the measurement validation gate at time $t = k$.

Under the assumption of Gaussian measurement noise, the likelihood of the ith validated measurement is also Gaussian and hence the likelihood term within the summation sign is a Gaussian PDF,

$$p(y_k(i) \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}, \alpha_i) \approx \mathcal{N}(y_k(i); \mathcal{H}\mathcal{X}_k, \mathcal{R}). \tag{31}$$

Therefore the expression for likelihood from (30) is

$$\begin{aligned} & p(y_k \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) \\ &= \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) \left\{ 1 - P_D P_G + P_D P_G \frac{V_k}{\lambda} \sum_{i=1}^{m_k} \mathcal{N}(y_k(i); \mathcal{H}\mathcal{X}_k, \mathcal{R}) \right\}. \end{aligned} \tag{32}$$

Prediction

Given the linear process and measurement equations of (25) and (26), the prediction can be directly derived from Kalman filter theory and is given as

$$p(\mathcal{X}_k \mid \mathcal{E}_k, y^{k-1}) = \mathcal{N}(\mathcal{X}_k; \hat{\mathcal{X}}_{k|k-1}, \mathcal{P}_{k|k-1}) \tag{33}$$

where

$$\hat{\mathcal{X}}_{k|k-1} = \mathcal{F}\hat{\mathcal{X}}_{k-1|k-1} \tag{34}$$

$$\mathcal{P}_{k|k-1} = \mathcal{F}\mathcal{P}_{k-1|k-1}\mathcal{F}^T + \mathcal{Q}. \tag{35}$$

Normalization

From (24), the normalization is

$$\begin{aligned} & p(y_k \mid \mathcal{E}_k, y^{k-1}) = \delta \\ &= \int_{\mathcal{X}_k} p(y_k \mid \mathcal{X}_k, \mathcal{E}_k, y^{k-1}) \times p(\mathcal{X}_k \mid \mathcal{E}_k, y^{k-1}) d\mathcal{X}_k \\ &= \int_{\mathcal{X}_k} \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) \\ &\quad \times \left\{ 1 - P_D P_G + P_D P_G \frac{V_k}{\lambda} \sum_{i=1}^{m_k} \mathcal{N}(y_k(i); \mathcal{H}\mathcal{X}_k, \mathcal{R}) \right\} \\ &\quad \times \mathcal{N}(\mathcal{X}_k; \hat{\mathcal{X}}_{k|k-1}, \mathcal{P}_{k|k-1}) d\mathcal{X}_k. \end{aligned} \tag{36}$$

In (36)

$$\begin{aligned}
&\mathcal{N}(y_k(i);\mathcal{H}\mathcal{X}_k,\mathcal{R})\mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k-1},\mathcal{P}_{k|k-1}) \\
&\quad = \frac{\mathcal{N}(y_k(i);\mathcal{H}\mathcal{X}_k,\mathcal{R})\mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k-1},\mathcal{P}_{k|k-1})}{\mathcal{N}(y_k(i);H\hat{\mathcal{X}}_{k|k-1},\mathcal{S})} \\
&\qquad \times \mathcal{N}(y_k(i);H\hat{\mathcal{X}}_{k|k-1},\mathcal{S}) \\
&\quad = \mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k}(i),\mathcal{P}_{k|k}(i)) \times \mathcal{N}(y_k(i);H\hat{\mathcal{X}}_{k|k-1},\mathcal{S})
\end{aligned} \tag{37}$$

where

$$\begin{aligned}
\mathcal{S} &= \mathcal{H}\mathcal{P}_{k|k-1}\mathcal{H}^T + \mathcal{R} \\
\mathcal{K} &= \mathcal{P}_{k|k-1}\mathcal{H}^T(\mathcal{S})^{-1} \\
\hat{\mathcal{X}}_{k|k}(i) &= \hat{\mathcal{X}}_{k|k-1} + \mathcal{K}(y_k(i) - \mathcal{H}\hat{\mathcal{X}}_{k|k-1}) \\
\mathcal{P}_{k|k}(i) &= (\mathcal{I} - \mathcal{K}\mathcal{H})\mathcal{P}_{k|k-1}.
\end{aligned}$$

Therefore (36) becomes

$$\begin{aligned}
&p(y_k \mid \mathcal{E}_k, y^{k-1}) = \delta \\
&\quad = \int_{\mathcal{X}_k} \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) \\
&\qquad \times \left\{1 - P_D P_G + P_D P_G \frac{V_k}{\lambda} \sum_{i=1}^{m_k} \mathcal{N}(y_k(i);H\hat{\mathcal{X}}_{k|k-1},\mathcal{S})\right\} \\
&\qquad \times \mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k}(i),\mathcal{P}_{k|k}(i))d\mathcal{X}_k \\
&\quad = \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) \\
&\qquad \times \left\{1 - P_D P_G + P_D P_G \frac{V_k}{\lambda} \sum_{i=1}^{m_k} \mathcal{N}(y_k(i);H\hat{\mathcal{X}}_{k|k-1},\mathcal{S})\right\}.
\end{aligned} \tag{38}$$

Now putting all the respective expressions in (24), the conditional state estimate becomes

$$\begin{aligned}
&p(\mathcal{X}_k \mid \mathcal{E}_k, y_k) \\
&\quad = \frac{1}{\delta}\mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k-1},\mathcal{P}_{k|k-1}) \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) \\
&\qquad \times \left\{1 - P_D P_G + P_D P_G \frac{V_k}{\lambda} \sum_{i=1}^{m_k} \mathcal{N}(y_k(i);\mathcal{H}\mathcal{X}_k,\mathcal{R})\right\} \\
&\quad = \frac{1}{\delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k)(1 - P_D P_G)\mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k-1},\mathcal{P}_{k|k-1}) \\
&\qquad + \sum_{i=1}^{m_k} \frac{1}{\delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) P_D P_G \frac{V_k}{\lambda} \\
&\qquad \times \mathcal{N}(y_k(i);\mathcal{H}\mathcal{X}_k,\mathcal{R})\mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k-1},\mathcal{P}_{k|k-1}).
\end{aligned} \tag{39}$$

Using (37), (39) can be reduced to

$$\begin{aligned}
&p(\mathcal{X}_k \mid \mathcal{E}_k, y_k) \\
&\quad = \frac{1}{\delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k)(1 - P_D P_G)\mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k-1},\mathcal{P}_{k|k-1}) \\
&\qquad + \sum_{i=1}^{m_k} \frac{1}{\delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) P_D P_G \frac{V_k}{\lambda}\mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k}(i),\mathcal{P}_{k|k}(i)) \\
&\qquad \times \mathcal{N}(y_k(i);\mathcal{H}\hat{\mathcal{X}}_{k|k-1},\mathcal{S}) \\
&\quad = \beta_k(0)\mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k-1},\mathcal{P}_{k|k-1}) \\
&\qquad + \sum_{i=1}^{m_k} \beta_k(i)\mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k}(i),\mathcal{P}_{k|k}(i)) \\
&\quad = \sum_{i=0}^{m_k} \beta_k(i) \times \mathcal{N}(\mathcal{X}_k;\hat{\mathcal{X}}_{k|k}(i),\mathcal{P}_{k|k}(i))
\end{aligned} \tag{40}$$

where

$$\beta_k(0) = \frac{1}{\delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k)(1 - P_D P_G) \tag{41}$$

$$\beta_k(i) = \frac{1}{\delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) P_D P_G \frac{V_k}{\lambda}\mathcal{N}(y_k(i);H\hat{\mathcal{X}}_{k|k-1},\mathcal{S}) \tag{42}$$

and taking

$$\hat{\mathcal{X}}_{k|k}(0) = \hat{\mathcal{X}}_{k|k-1} \tag{43}$$

$$\mathcal{P}_{k|k}(0) = \mathcal{P}_{k|k-1}. \tag{44}$$

From (40), the estimates of state and covariance are derived as

$$\hat{\mathcal{X}}_{k|k} = \sum_{i=0}^{m_k} \beta_k(i)\hat{\mathcal{X}}_{k|k}(i) \tag{45}$$

$$\mathcal{P}_{k|k} = \sum_{i=0}^{m_k} \beta_k(i)\mathcal{P}_{k|k}(i) + \sum_{i=0}^{m_k} \beta_k(i)\hat{\mathcal{X}}_{k|k}(i)\hat{\mathcal{X}}_{k|k}(i)^T - \hat{\mathcal{X}}_{k|k}\hat{\mathcal{X}}_{k|k}^T. \tag{46}$$

The expressions in (45) and (46) give the state estimate and its covariance matrix conditioned on the target existence.

4.2 Existence Probability Estimate

Smoothing of the existence probability requires two steps,

- first, calculation of the probabilities of the augmented existence hypotheses,
- second, from there calculation of existence probabilities at each time instant using (22) and (23).

Here the derivation of the probabilities is shown in details.

Hypothesis 1:

$$
\begin{aligned}
p(\mathcal{E}_k \mid y^k) &= p(\mathcal{E}_k \mid y_k, y^{k-1}) \\
&= \frac{p(y_k \mid \mathcal{E}_k, y^{k-1}).p(\mathcal{E}_k \mid y^{k-1})}{\Delta} \\
&= \frac{1}{\Delta}\delta \times [p(E_k,\ldots,E_{k-N},E_{K-N-1} \mid y^{k-1}) \\
&\quad + p(E_k,\ldots,E_{k-N},\bar{E}_{K-N-1} \mid y^{k-1})] \\
&= \frac{1}{\Delta}\delta[p(E_k \mid E_{k-1}).\ldots.p(E_{k-N} \mid E_{k-N-1}).p(E_{k-N-1} \mid y_{k-1})] \\
&\quad + p(E_k \mid E_{k-1}).\ldots.p(E_{k-N} \mid \bar{E}_{k-N-1}).p(\bar{E}_{k-N-1} \mid y_{k-1})] \\
&= \frac{1}{\Delta}\delta \times (\Gamma_{11})^{N+1} \times p(E_{k-N-1} \mid y_{k-1}) \qquad (47)
\end{aligned}
$$

where

$$
\begin{aligned}
\Delta &= p(y_k \mid \mathcal{E}_k, y^{k-1}).p(\mathcal{E}_k \mid y^{k-1}) \\
&\quad + \sum_{m=0}^{N-1} p(y_k \mid \mathcal{E}_k^m, y^{k-1}).p(\mathcal{E}_k^m \mid y^{k-1}) \\
&\quad + p(y_k \mid \mathcal{E}_k^n, y^{k-1}).p(\mathcal{E}_k^n \mid y^{k-1}). \qquad (48)
\end{aligned}
$$

Hypothesis m:

$$
\begin{aligned}
&p(\mathcal{E}_k^m \mid y^k) \\
&\quad = p(\mathcal{E}_k^m \mid y_k, y^{k-1}) \\
&\quad = \frac{p(y_k \mid \mathcal{E}_k^m, y^{k-1}).p(\mathcal{E}_k^m \mid y^{k-1})}{\Delta} \\
&\quad = \frac{1}{\Delta} p(y_k \mid \bar{E}_k, y^{k-1}) \\
&\quad .p(\bar{E}_k,\ldots,\bar{E}_{k-m},E_{k-m-1},\ldots,E_{k-N} \mid y^{k-1}) \\
&\quad = \frac{1}{\Delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k) \\
&\quad\quad [p(\bar{E}_k,\ldots,\bar{E}_{k-m},E_{k-m-1},\ldots,E_{k-N},E_{k-N-1} \mid y^{k-1}) \\
&\quad\quad + p(\bar{E}_k,\ldots,\bar{E}_{k-m},E_{k-m-1},\ldots,E_{k-N},\bar{E}_{k-N-1} \mid y^{k-1})] \\
&\quad = \frac{1}{\Delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k)(\Gamma_{00})^m.\Gamma_{10}.(\Gamma_{11})^{N-m} p(E_{k-N-1} \mid y^{k-1}). \\
&\qquad\qquad (49)
\end{aligned}
$$

Hypothesis n:

$$
\begin{aligned}
p(\mathcal{E}_k^n \mid y^k) &= p(\mathcal{E}_k^n \mid y_k, y^{k-1}) \\
&= \frac{p(y_k \mid \mathcal{E}_k^n, y^{k-1}).p(\mathcal{E}_k^n \mid y^{k-1})}{\Delta} \\
&= \frac{1}{\Delta} p(y_k \mid \bar{E}_k, y^{k-1}).p(\bar{E}_k,\ldots,\bar{E}_{k-N} \mid y^{k-1}) \\
&= \frac{1}{\Delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k)[p(\bar{E}_k,\ldots,\bar{E}_{k-N},E_{k-N-1} \mid y^{k-1}) \\
&\quad + p(\bar{E}_k,\ldots,\bar{E}_{k-N},\bar{E}_{k-N-1} \mid y^{k-1})] \\
&= \frac{1}{\Delta}\left(\frac{1}{V_k}\right)^{m_k} P_0(m_k)[\Gamma_{00}^N.\Gamma_{10} p(E_{k-N-1} \mid y^{k-1}) \\
&\quad + \Gamma_{00}^{N+1}(1 - p(E_{k-N-1} \mid y^{k-1}))]. \qquad (50)
\end{aligned}
$$

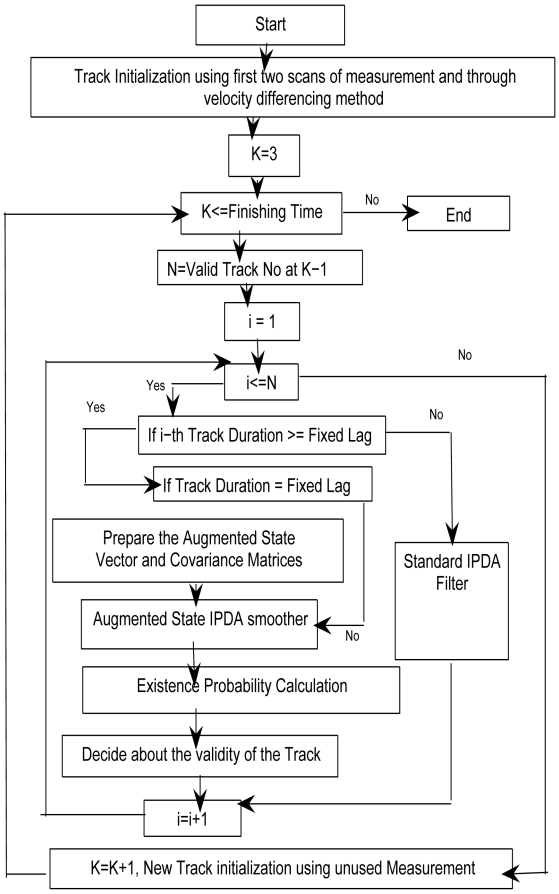


Fig. 1. Flow chart of AS-IPDA smoothing.

Both in Hypothesis m and Hypothesis n, the target does not exist at the current time $t = k$ and so by definition

$$
\begin{aligned}
P(\alpha_0 \mid \bar{E}_k, y^{k-1}) &= 1 \\
P(\alpha_i \mid \bar{E}_k, y^{k-1}) &= 0 \qquad \text{for} \quad i = 1,2,\ldots,m_k.
\end{aligned}
$$

Therefore the likelihood,

$$
p(y_k \mid \bar{E}_k, y^{k-1}) = \left(\frac{1}{V_k}\right)^{m_k} \qquad (51)
$$

is used in the derivation of (49) and (50).

Thus (47)–(50) give the probabilities of the augmented existence hypotheses. From these expressions, the track existence probability at each time step (of the entire lag of N) can be obtained by using (22) and (23).

5. ALGORITHM FLOW CHART

In this section the proposed smoothing algorithm is converted into a flow chart for direct implementation. The flow chart is given in Fig. 1.

6. SIMULATION RESULT

Simulations were carried out to investigate the performance of the proposed AS-IPDA smoother with standard IPDA filter. The simulation scenario consists of nonmaneuvering targets moving in a 500 m long and

200 m wide two-dimensional surveillance region. The target dynamic state is assumed to consist of position and velocity in each axis. The state transition matrix is defined as

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where is the sampling period $T = 1$ s.

The sensor receives the position of each target. Hence the state to measurement conversion matrix is defined as

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (52)$$

The process noise is zero mean with covariance $E[w(k)w(j)'] = Q$ where

$$Q = q \begin{bmatrix} T^4/4 & T^3/2 & 0 & 0 \\ T^3/2 & T^2 & 0 & 0 \\ 0 & 0 & T^4/4 & T^3/2 \\ 0 & 0 & T^3/2 & T^2 \end{bmatrix}$$

where $q = 0.25$. The sensor introduces an error with variance 3 m in either coordinate axis.

The number of clutter points is generated according to a Poisson distribution with a density of 1.0×10^{-4}/scan/m^2. The clutter points are also uniformly distributed within the whole surveillance region. The tracks are initiated by two-point differencing assuming a maximum velocity of 50 m/s. The detection probability is 0.90.

The target existence transition matrix used is

$$\Gamma = \begin{bmatrix} 0.98 & 0.02 \\ 0 & 1 \end{bmatrix}$$

The measurement validation gate threshold is 9, which ensures a gating probability of $P_G = 0.99$. If the existence probability of a track equals or goes above 0.9, i.e

$$p(E_k \mid y^k) \geqslant 0.9$$

the track is moved from tentative to confirmed while if

$$p(E_k \mid y^k) < 0.05$$

the track is terminated.

The track-to-track association threshold is taken to be 0.005. The parameters are tested using 1000 Monte-Carlo run (for true/false track discrimination 400 runs were used) where each time the target reappears with state $[100\text{ m}\ \ 25\text{ m/s}\ \ 100\text{ m}\ \ 5\text{ m/s}]^T$. Also a default fix lag of three was used for the smoother.

6.1 Termination Time Detection

In this simulation scenario the simulation is carried out for 40 scans while the single target is dropped at the 30th scan. The remaining 10 scans consist of only clutter. Two different switching matrices are used for comparison purposes—one with $\Gamma_{11} = 0.98$ and the other with $\Gamma_{11} = 0.90$ where, by definition, $\Gamma_{10} = 1 - \Gamma_{11}$. The results for these two cases are shown in Table I and Table II.

TABLE I
$\Gamma_{11} = 0.98$, Actual Termination Time = 30

Filter Detection	Smoother Detection			
	Lag 1	Lag 2	Lag 3	Lag 4
34	33	32	31	30

TABLE II
$\Gamma_{11} = 0.9$, Actual Termination Time = 30

Filter Detection	Smoother Detection			
	Lag 1	Lag 2	Lag 3	Lag 4
33	32	32	32	31

6.2 Target State Estimation

The filter and smoother are compared in terms of RMS error in position and velocity for both axes. A detection of probability of 0.90 was used. The results are shown in Figs. 2–5.

6.3 True/False Discrimination

For the simulation of the smoothing performance in terms of true/false track detection, the scenario is set as a 400 run of the simulation. A single target reappears at the beginning of each run. The smoother uses a fixed lag of three. At each time instant, the tracks that are updated with true target originated measurements are considered as true tracks. The detected number of confirmed true tracks against time is shown in the Fig. 6. Also the number of false tracks that are confirmed by both algorithms are shown in Fig. 7.

7. CONCLUSION

In this paper, a fixed lag smoother is derived to solve the target existence uncertainty problem in clutter. Besides providing better state estimation, the smoother performs better in distinguishing between true and false tracks as well as determining true targets' termination time compared to a standard filter algorithm. The proposed algorithm extends automatic track initiation into the smoothing type of estimation and establishes the immediate gain in estimation. Applications that take higher level decisions or make strategies shall obtain a great advantage due to such an improvement by using smoothers such as the proposed one.

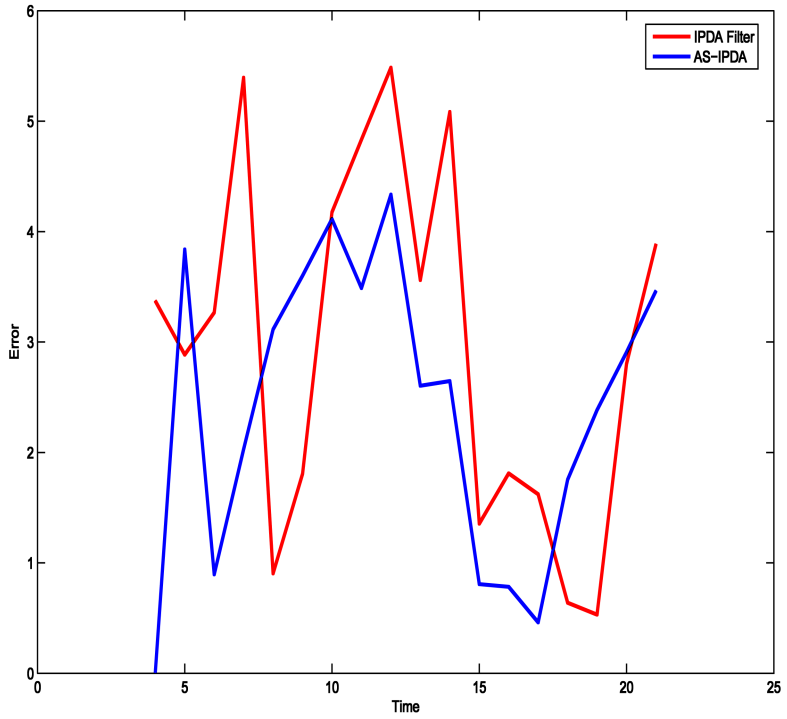


Fig. 2. RMS error comparison of x-direction position using 1000 Monte Carlo runs with detection probability 0.90.

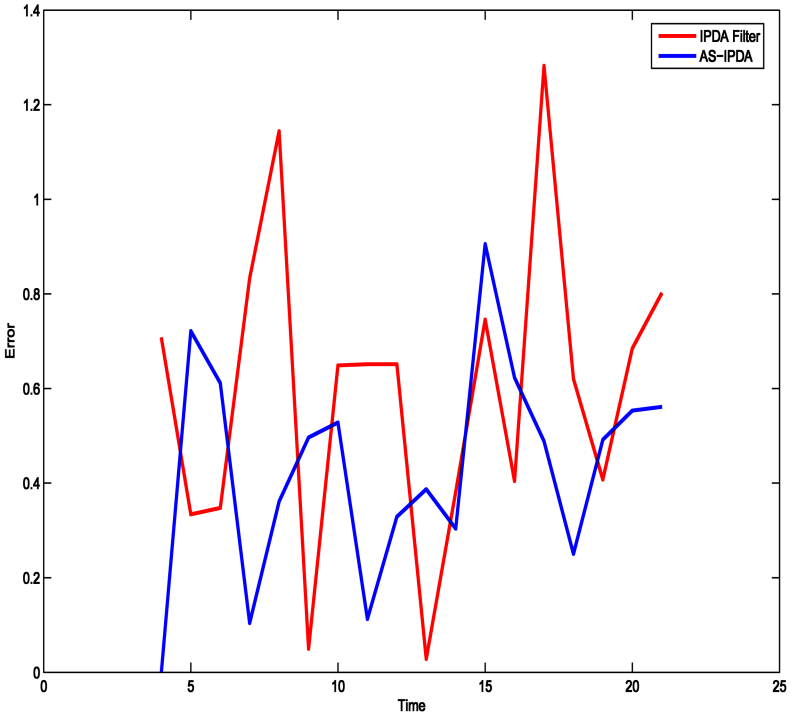


Fig. 3. RMS error comparison of x-direction velocity using 1000 Monte Carlo runs with detection probability 0.90.

APPENDIX. THE HYPOTHESES $\mathbf{H}_k^m$ and $\mathbf{H}_k^n$ DO NOT CONTRIBUTE TO THE STATE UPDATE

$$\begin{aligned} p(\mathcal{X}_k^m \mid \mathcal{E}_k^m, y^k) &= p(\mathcal{X}_k^m \mid \mathcal{E}_k^m, y_k, y^{k-1}) \\ &= \frac{p(y_k \mid \mathcal{X}_k^m, \mathcal{E}_k^m, y^{k-1}).p(\mathcal{X}_k^m \mid \mathcal{E}_k^m, y^{k-1})}{p(y_k \mid \mathcal{E}_k^m, \mathcal{Y}^{k-1})} \\ &= \frac{\text{Likelihood} \times \text{Prediction}}{\text{Normalization}}. \end{aligned} \tag{53}$$

Under the assumption that the target does not exist at time $t = k$, the likelihood term in (53) reduces to,

$$p(y_k \mid \mathcal{X}_k^m, \mathcal{E}_k^m, y^{k-1}) = p(y_k \mid \bar{E}_k, y^{k-1}) = \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k). \tag{54}$$

The prediction term is

$$p(\mathcal{X}_k^m \mid \mathcal{E}_k^m, y^{k-1}) = p(x_{k-m-1}, \ldots, x_{k-N} \mid E_{k-m-1}, \ldots, E_{k-N}, y^{k-1}). \tag{55}$$

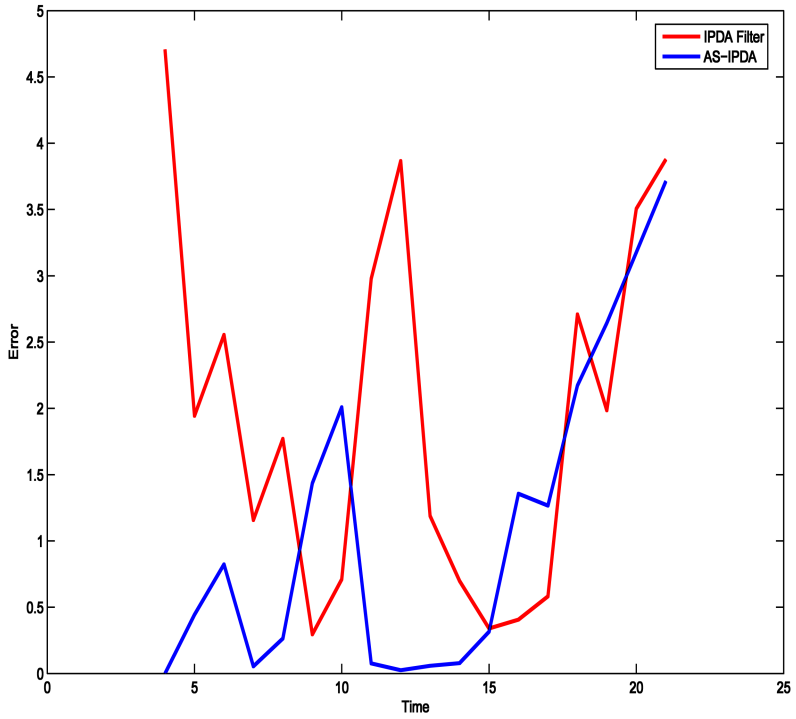


Fig. 4. RMS error comparison of y-direction position using 1000 Monte Carlo runs with detection probability 0.90.

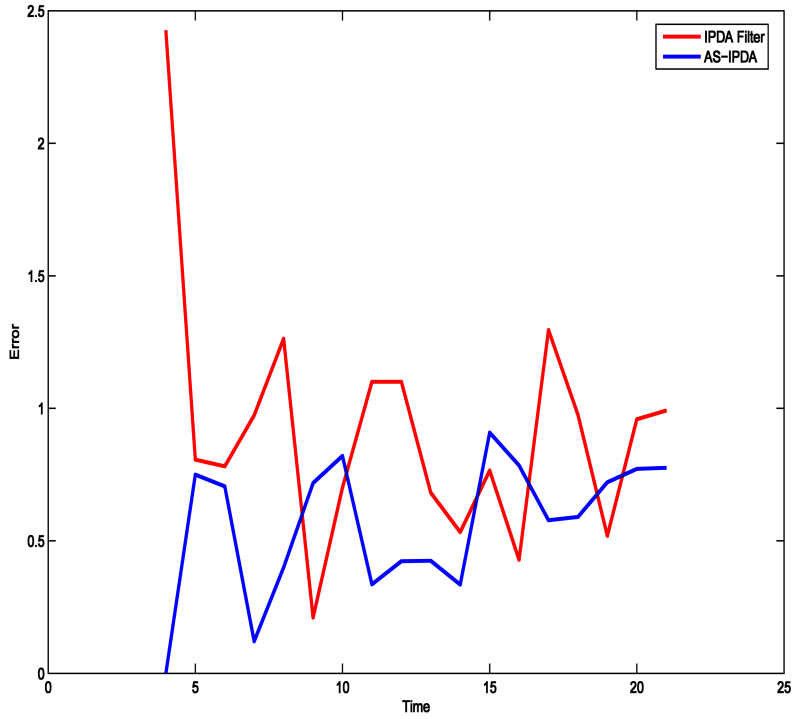


Fig. 5. RMS error comparison of y-direction velocity using 1000 Monte Carlo runs with detection probability 0.90.

Lastly the normalization is

$$\begin{aligned} & p(y_k \mid \mathcal{E}_k^m, y^{k-1}) \\ & \quad = \int_{\mathcal{X}_k^m} p(y_k \mid \mathcal{X}_k^m, \mathcal{E}_k^m, y^{k-1}) \times p(\mathcal{X}_k^m, \mathcal{E}_k^m, y^{k-1})) d\mathcal{X}_k^m \\ & \quad = \left(\frac{1}{V_k}\right)^{m_k} P_0(m_k). \end{aligned} \tag{56}$$

As the likelihood and normalization terms are the same, (53) can be simplified as

$$\begin{aligned} & p(\mathcal{X}_k^m \mid \mathcal{E}_k^m, y^k) \\ & \quad = p(x_{k-m-1}, \ldots, x_{k-N} \mid E_{k-m-1}, \ldots, E_{k-N}, y^{k-1}) \\ & \qquad \times p(\mathcal{X}_k^m \mid \mathcal{E}_k^m, y^{k-1}). \end{aligned} \tag{57}$$

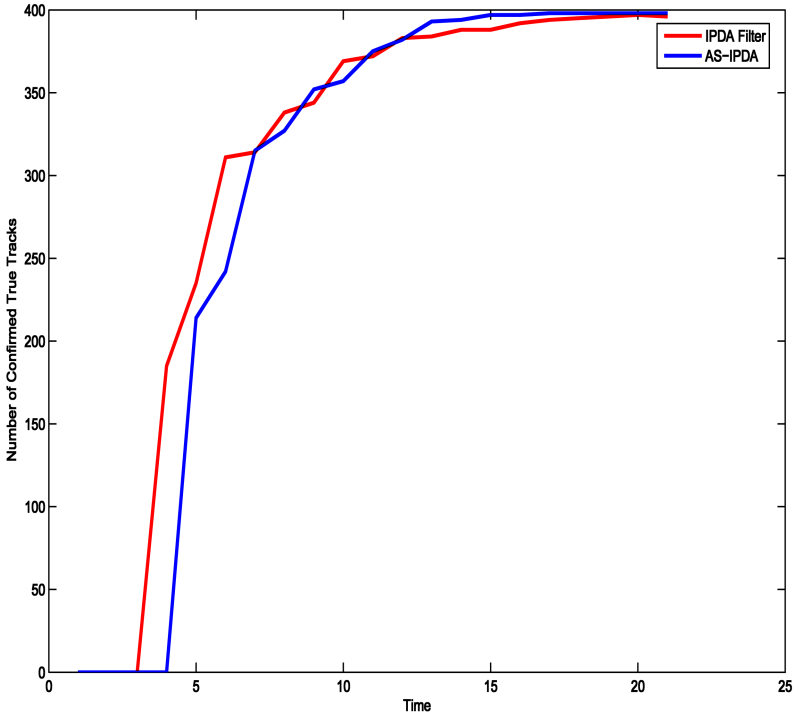


Fig. 6. Number of confirmed true tracks.

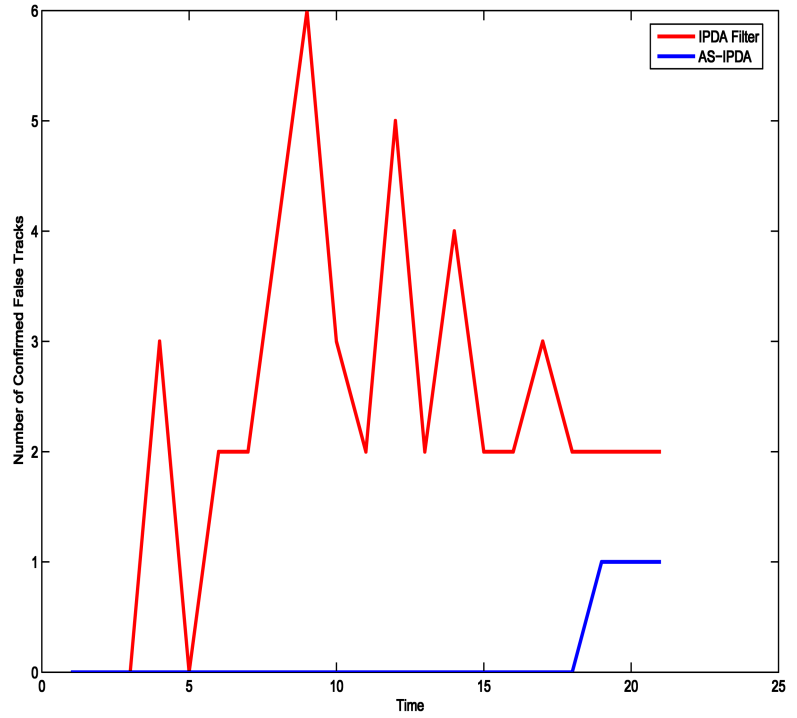


Fig. 7. Number of confirmed false tracks.

Thus (57) shows that the hypotheses $\mathbf{H}_k^m$ and $\mathbf{H}_k^n$, where the target does not exist at current time $t = k$, do not contribute to the update of the state and covariance. So if the target does not exist at current time, the previous smoothed or filtered values are retained as is.

REFERENCES

[1] Y. Bar-Shalom
Tracking methods in a multitarget environment.
IEEE Transactions on Automatic Control, 4 (1978), 618–626.

[2] Y. Bar-Shalom, K. C. Chang and H. A. P. Blom
Automatic track formation in clutter with a recursive algorithm.
In *Proceedings of the 28th IEEE Conference on Decision and Control*, vol. 2, 1989, 1402–1408.

[3] Y. Bar-Shalom and X. R. Li
Estimation and Tracking: Principle and Software.
Boston: Artech House, 1993.

[4] Y. Bar-Shalom and X. R. Li
Multitarget-Multisensor Tracking: Principles and Techniques.
Storrs, CT: 1995.

[5] Y. Bar-Shalom and E. Tse
Tracking in a cluttered environment with probabilistic data association.
Automatica, **11** (1975), 451–460.

[6] F. R. Castella
Sliding window detection probabilities.
IEEE Transactions of Aerospace and Electronic Systems, **AES-12** (1976), 815–819.

[7] R. Chakravorty and S. Challa
A single lag smoothing technique for track maintenance in clutter.
In *Proceedings of IEEE conference of Cybernetics and Intelligent Systems*, vol. 2, Dec. 1–3, 2004, 1265–1269.

[8] R. Chakravorty and S. Challa
Fixed lag smoothing technique for track maintenance in clutter.
In *Proceedings of International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, Dec. 14–17, 2004, 119–124.

[9] R. Chakravorty and S. Challa
Smoothing framework for automatic track initiation in clutter.
In *Proceedings of 7th International Conference on Information Fusion*, vol. 1, July 25–28, 2005, 54–61.

[10] S. Challa, R. Evans and D. Musicki
Target tracking—A Bayesian perspective.
In *Fourteenth International Conference on Digital Signal Processing*, vol. 1, 2002, 437–440.

[11] S. Challa, R. Evans and X. Wang
A Bayesian solution and its approximations to out-of-sequence measurement problems.
Information Fusion, vol. 4, 2003, 185–199.

[12] B. Chen and J. K. Tuganit
Multisensor tracking of a maneuvering target in clutter using IMMPDA fixed-lag smoothing.
IEEE Transactions on Aerospace and Electronic Systems, **36**, 3 (2000), 983–991.

[13] R. E. Helmick, W. D. Blair and S. A. Hoffman
Interacting multiple-model approach to fixed interval smoothing.
In *Proceedings of the 32nd IEEE Conference on Decision and Control*, vol. 4, Dec. 15–17, 1993, 3052–3057.

[14] R. E. Helmick, W. D. Blair and S. A. Hoffman
One step fixed-lag smoothers for Markovian switching systems.
IEEE Transactions on Automatic Control, **41**, 7 (July 1996), 1051–1056.

[15] R. E. Kalman
A new approach to linear filtering and prediction problems.
Transactions on ASME, Journal of Engineering, **82** (Mar. 1960), 34–45.

[16] R. E. Kalman and R. S. Bucy
New results in linear filtering and prediction theory.
Transactions on ASME, Journal of Engineering, **83** (Mar. 1961), 95–108.

[17] A. K. Mahalanabis, B. Zhou and N. K. Bose
Improved multi-target tracking in clutter by PDA smoothing.
IEEE Transactions on Aerospace and Electronic Systems, **26**, 1 (1990), 113–121.

[18] D. Musicki and R. Evans
Joint integrated probabilistic data association: JIPDA.
In *Proceedings of the Fifth International Conference on Information Fusion*, vol. 2, July 8–11, 2002, 1120–1125.

[19] D. Musicki and R. Evans
Tracking in clutter using probabilistic data association.
In *Proceedings of International Conference on Radar*, Oct. 12–13, 1992, 82–85.

[20] D. Musicki and R. Evans
Linear joint integrated probabilistic data association: JIPDA.
In *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 3, Dec. 10–13, 2002, 2415–2420.

[21] D. Musicki and R. Evans
Joint integrated probabilistic data association: JIPDA.
IEEE Transactions on Aerospace and Electronic Systems, **40**, 3 (July 2004), 1093–1099.

[22] D. Musicki, R. Evans and S. Stankovic
Integrated probabilistic data association (IPDA).
In *Proceedings of the 31st IEEE Conference on Decision and Control*, vol. 4, Dec. 16–18, 1992, 3796–3798.

[23] D. Musicki, R. Evans and S. Stankovic
Integrated probabilistic data association.
IEEE Transactions on Automatic Control, **39**, 6 (June 1994), 1237–1241.

[24] P. J. Vesselin, X. R. Li and L. Lu
Performance enhancement of the IMM estimation by smoothing.
In *Proceedings of 5th International conference on Information Fusion*, Annapolis, MD, 713–720.

[25] X. Wang and S. Challa
Augmented state IMM-PDA for OOSM solution to maneuvering target tracking in clutter.
In *Proceedings of International Radar Conference*, 2003, 479–485.

Rajib Chakravorty received the B.Sc. degree in Electrical and Electronics Engineering from Bangladesh University of Engineering and Technology, Dhaka, Bangladesh in 2003. Currently he is completing his Ph.D. degree from the University of Technology, Sydney, NSW, Australia.

His research interests include stochastic estimation and target tracking with particular emphasis on smoothing applied to problems involving target existence uncertainty. He is also engaged in practical implementation of data fusion using vision sensors.

Subhash Challa is a Professor of Computer Systems and the director of the Networked Sensor Technologies Lab at the University of Technology Sydney. He received his B. Tech. degree from Jawaharlal Nehru Technological University, Hyderabad, India, in 1994 and a Ph.D. degree from Queensland University of Technology, Brisbane, Australia 1999.

He was a visiting fellow at Harvard University, in 1997, before joining the Department of Electrical Engineering, at the University of Melbourne, Melbourne, Australia in 1998, where he headed the Tracking and Data Fusion Laboratory.

His current collaborative research involves collaborations with RTA (Remote Bridge Monitoring, New South Whales Sydney), Cradle Technologies (Multi DSP/RISC processor company, USA), Intersystems (Embedded Data Bases, Sydney), Rolachem (Water Quality Monitoring), ADI (Design of Wireless Sensor Networks, Sydney), Iomniscient (Multi-Camera Video Surveillance, Sydney), Tenix Defense Systems (Multi Sensor, Multi-Target Tracking) and Citect (SCADA Software Company). Prior to this, he has managed a number of defence contracts from the Defence Science Technology Organisation (DSTO), Australia, Defence Advanced Research Program (DARPA), through Veridian (General Dynamics), and Scientific Systems, Boston. He has cofounded Sensen Networks, a sensor networks systems company, in 2005 in Australia and Reline Technologies in 2003 in India. He has published over 70 papers in various international journals and conferences in the areas of information fusion and tracking.

He has been an active participant in ISIF (International Society for Information Fusion) activities and has served ISIF in various capacities including as a session chair for joint target tracking and recognition in FUSION2000, and as a publicity chair for Fusion 2002 held in Washington DC. His most notable contribution to ISIF was the organization of the FUSION2003 conference that was held in Cairns, Australia in July 2003. In addition, he has been the program chair for the International Decision and Control Conference (IDC2002), Adelaide, Australia. He was an invited speaker at the International Radar Symposium India in 2001 and 2003. He has given over 10 Industry courses in the areas of tracking and data fusion. He was a Tan-Chun-Tau Fellow at Nanyang Technological University, Singapore from October 2002 to February 2003.

Robust Bayesianism: Relation to Evidence Theory

STEFAN ARNBORG
Kungliga Tekniska Högskolan

We are interested in understanding the relationship between Bayesian inference and evidence theory. The concept of a set of probability distributions is central both in robust Bayesian analysis and in some versions of Dempster-Shafer's evidence theory. We interpret imprecise probabilities as imprecise posteriors obtainable from imprecise likelihoods and priors, both of which are convex sets that can be considered as evidence and represented with, e.g., DS-structures. Likelihoods and prior are in Bayesian analysis combined with Laplace's parallel composition. The natural and simple robust combination operator makes all pairwise combinations of elements from the two sets representing prior and likelihood. Our proposed combination operator is unique, and it has interesting normative and factual properties. We compare its behavior with other proposed fusion rules, and earlier efforts to reconcile Bayesian analysis and evidence theory. The behavior of the robust rule is consistent with the behavior of Fixsen/Mahler's modified Dempster's (MDS) rule, but not with Dempster's rule. The Bayesian framework is liberal in allowing all significant uncertainty concepts to be modeled and taken care of and is therefore a viable, but probably not the only, unifying structure that can be economically taught and in which alternative solutions can be modeled, compared and explained.

Manuscript received August 29, 2005; revised January 30, 2006 and April 20, 2006.

Refereeing of this contribution was handled by Pierre Valin.

Author's address: Kungliga Tekniska Högskolan, Stockholm, SE-100 44, Sweden, E-mail: (stefan@nada.kth.se).

1. INTRODUCTION

Several, apparently incomparable, approaches exist for uncertainty management. Uncertainty management is a broad area applied in many different fields, where information about some underlying, not directly observable, truth—the state of the world—is sought from a set of observations that are more or less reliable. These observations can be, for example, measurements with random and/or systematic errors, sensor readings, or reports submitted by observers. In order that conclusions about the conditions of interest be possible, there must be some assumptions made on how the observations relate to the underlying state about which information is sought. Most such assumptions are numerical in nature, giving a measure that indicates how plausible different underlying states are. Such measures can usually be normalized so that the end result looks very much like a probability distribution over the possible states of the world, or over sets of possible world states. However, uncertainty management and information fusion is often concerned with complex technical, social or biological systems that are incompletely understood, and it would be naive to think that the relationship between observation and state can be completely captured. At the same time, such systems must have at least some approximate ways to relate observation with state in order to make uncertainty management at all possible.

It has been a goal in research to encompass all aspects of uncertainty management in a single framework. Attaining this goal should make the topic teachable in undergraduate and graduate engineering curricula and facilitate engineering applications development. We propose here that robust Bayesian analysis is such a framework. The Dempster-Shafer or evidence theory originated within Bayesian statistical analysis [19], but when developed by Shafer [51] took the concept of belief assignment rather than probability distribution as primitive. The assumption being that bodies of evidence—beliefs about the possible worlds of interest—can be taken as primitives rather than sampling functions and priors. Although this idea has had considerable popularity, it is inherently dangerous since it seems to move application away from foundational justification. When the connection to Bayes' method and Dempster's application model is broken, it is no longer necessary to use the Dempster combination rule, and evidence theory abounds with proposals on how bodies of evidence should be interpreted and combined, as a rule with convincing but disparate argumentation. But there seems not to exist other bases for obtaining bodies of evidence than likelihoods and priors, and therefore an analysis of a hypothetical Bayesian obtainment of bodies of evidence can bring light to problems in evidence theory. Particularly, a body of evidence represented by a DS-structure has an interpretation as a set of possible probability distributions, and combining or aggregating two such structures can be done in robust

Bayesian analysis. The resulting combination operator is trivial, but compared to other similar operators it has interesting, even surprising, behavior and normative advantages. Some concrete progress in working with convex sets of probability vectors has been described in [41, 57, 29]. It appears that the robust combination operator we discuss has not been analyzed in detail and compared to its alternatives, and is missing in recent overviews of evidence and imprecise probability theory. Our ideas are closely related to problems discussed in [32] and in the recent and voluminous report [21], which also contains a quite comprehensive bibliography. The Workshop hosted by the SANDIA lab has resulted in an overview of current probabilistic uncertainty management methods [34]. A current overview of alternative fusion and estimation operators for tracking and classification is given in [45].

The main objective of this paper is to propose that precise and robust Bayesian analysis are unifying, simple and viable methods for information fusion, and that the large number of methods possible can and should be evaluated by taking into account the appropriateness of statistical models chosen in the particular application where it is used. We are aware, however, that the construction of Bayesian analysis as a unifying concept has no objective truth. It is meant as a post-modernistic project facilitating teaching and returning artistic freedom to objective science. The Bayesian method is so liberal that it almost never provides unique exact solutions to inference and fusion problems, but is completely dependent on insightful modeling. The main obstacle to achieving acceptance of the main objective seems to be the somewhat antagonistic relationship between the different schools where sometimes sweeping arguments have been made that seem rather unfair whoever launched them, typical examples being [42, 51] and the discussions following them.

Another objective is to investigate the appropriateness of particular fusion and estimation operations, and their relationships to the robust as well as the precise Bayesian concept. Specifically, we show that the choice between different fusion and estimation operations can be guided by a Bayesian investigation of the application.

We also want to connect the analysis to practical concerns in information fusion and keep the mathematical/theoretical level of the presentation as simple as possible, while also examining the problem to its full depth. A quite related paper promoting similar ideas is Mahler [43], which however is terser and uses somewhat heavier mathematical machinery.

Quite many comparisons have been made of Bayesian and evidential reasoning with the objective of guiding practice, among others [47, 10, 11, 50]. It is generally found that the methods are different and therefore one should choose a method that matches the application in terms of quantities available (evidence or likelihoods and priors), or the prevailing culture and construction of the application. Although the easiest way forward, this advice seems somewhat short-sighted given the quite large lifespan of typical advanced applications and the significant changes in understanding and availability of all kinds of data during this life-span.

In Section 2 we review Bayesian analysis and in Section 3 dynamic Bayesian (Chapman Kolmogorov/Kalman) analysis. In Section 4 we describe robust Bayesian analysis analysis and some of its relations to DS theory; in Section 5 we discuss decisions under uncertainty and imprecision and in Section 6 Zadeh's well-known example. In Section 7 we derive some evidence fusion operations and the robust combination operator. We illustrate their performance on a paradoxical example related to Zadeh's in Section 8, and wrap up with conclusions in Section 9.

2. BAYESIAN ANALYSIS

Bayesian analysis is usually explained [7, 38, 52, 24] using the formula

$$f(\lambda \mid x) \propto f(x \mid \lambda) f(\lambda) \tag{1}$$

where $\lambda \in \Lambda$ is the world of interest among $n = |\Lambda|$ possible worlds (sometimes called parameter space), and $x \in X$ is an observation among possible observations. The distinction between observation and world space is not necessary but is convenient—it indicates what our inputs are (observations) and what our outputs are (belief about possible worlds). The functions in the formula are probability distributions, discrete or continuous. We use a generic function notation common in statistics, so the different occurrences of f denote different functions suggested by their arguments. The sign $\propto$ indicates that the left side is proportional to the right side (as a function of λ), with the normalization constant left out. In (1), $f(x \mid \lambda)$ is a sampling distribution, or likelihood when regarded as a function of λ for a given x, which connects observation space and possible world space by giving a probability distribution of observed value for each possible world, and $f(\lambda)$ is a prior describing our expectation on what the world might be. The rule (1) gives the posterior distribution $f(\lambda \mid x)$ over possible worlds λ conditional on observations x. A paradox arises if the supports of $f(\lambda)$ and $f(x \mid \lambda)$ are disjoint (since each possible world is ruled out either by the prior or by the likelihood), a possibility we will ignore throughout this paper. Equation (1) is free of technical complication and easily explainable. It generalizes however to surprisingly complex settings, as required of any device helpful in design of complex technical systems. In such systems, it is possible that x represents a quantity which is not immediately observable, but instead our information about x is given by a probability distribution $f(x)$, typically obtained as a posterior from (1). Such observations are sometimes called fuzzy observations. In this case, instead of using (1) we apply:

$$f(\lambda \mid f(x)) \propto \int f(x \mid \lambda) f(x) f(\lambda) dx. \tag{2}$$

Ed Jaynes made (1) the basis for teaching science and interpretation of measurements [38]. In general, for infinite (compact metric) observation spaces or possible world sets, some measure-theoretic caution is called for, but it is also possible to base the analysis on well-behaved limit processes in each case as pointed out by, among others, Jaynes [38]. We will here follow Jaynes' approach and thus discuss only the finite case. That generalization to infinite and/or complexly structured unions of spaces of different dimensions and quotiented over symmetry relations is possible is known although maybe not obvious. Mahler claims that such applications are not Bayesian in [43], but they can apparently be described by (1) and similar problems are investigated within the Bayesian framework, for example by Green [26]. Needless to say, since the observation and world spaces can be high-dimensional and the prior and likelihood can be arbitrarily complex, practical work with (1) is full of pitfalls and one often encounters what looks like counterintuitive behaviors. On closer investigation, such problems can lead to finding a modeling error, but more often it shows that (1) is indeed better than one's first intuitive attitude.

It has been an important philosophical question to characterize the scope of applicability of (1), which lead to the distinction between objective and subjective probability, among other things. Several books and papers, among others [17, 49, 42, 15], claim that, under reasonable assumptions, (1) is the only consistent basis for uncertainty management. However, the minimal assumptions truly required to obtain this result turn out on closer inspection to be rather complex, as discussed in [7, 64, 33, 31, 46, 35, 2]. One simple assumption usually made in those studies that conclude in favor of (1) is that uncertainty is measured by a real number or on an ordered scale. Many established uncertainty management methods however measure uncertainty on a partially ordered scale and do apparently not use (1) and the accompanying philosophy. Among probability based alternatives to Bayesian analysis with partially ordered uncertainty concepts are imprecise probabilities or lower/upper prevision theory [62], the Dempster-Shafer (DS) [51], the Fixsen/Mahler (MDS) [22] and Dezert-Smarandache (DSmT) [53] theories. In these schools, it is considered important to develop the theory without reference to classical Bayesian thinking. In particular, the assumption of precise prior and sampling distributions is considered indefensible. Those assumptions are referred to as the dogma of precision in Bayesian analysis [63].

Indeed, when the inference process is widened from an individual to a social or multi-agent context, there must be ways to accommodate different assessments of priors and likelihoods. Thus, there is a possibility that two experts make the same inference using different likelihoods and priors. If expert 1 obtained observation set $X_1 \subseteq X$ and expert 2 obtained observation set $X_2 \subseteq X$, they would obtain a posterior belief of, e.g., a patient's condition expressible as $f_i(\lambda_i \mid X_i) \propto f_i(X_i \mid \lambda_i) f_i(\lambda_i)$, for $i = 1,2$. Here we have not assumed that the two experts used the same sampling and prior distributions. Even if training aims at giving the two experts the same "knowledge" in the form of sampling function and prior, this ideal cannot be achieved completely in practice. The Bayesian method prescribes that expert i states the probability distribution $f_i(\lambda_i \mid X_i)$ as his belief about the patient. If they use the same sampling function and prior, the Bayesian method also allows them to combine their findings to obtain:

$$\begin{aligned} f(\lambda \mid \{X_1, X_2\}) &\propto f(\{X_1, X_2\} \mid \lambda) f(\lambda) \\ &= f(X_1 \mid \lambda) f(X_2 \mid \lambda) f(\lambda) \end{aligned} \tag{3}$$

under the assumption:

$$f(\{X_1, X_2\} \mid \lambda) = f(X_1 \mid \lambda) f(X_2 \mid \lambda).$$

The assumption appears reasonable in many cases. In cases where it is not, the discrepancy should be entered in the statistical model. This is particularly important in information fusion for those cases where the first set of observations was used to define the second investigation, as in sensor management. This is an instance of selection bias. Ways of handling data selection biases are discussed thoroughly in [24]. Data selection bias is naturally and closely related to the missing data problem that has profound importance in statistics [48] and has also been examined in depth in the context of imprecise probability fusion [16].

It is important to observe that it is the two experts likelihood functions, not their posterior beliefs, that can be combined, otherwise we would replace the prior by its normalized square and the real uncertainty would be underestimated. This is at least the case if the experts obtained their training from a common body of medical experience coded in textbooks. If the posterior is reported and we happen to know the prior, the likelihood can be obtained by $f(X \mid \lambda) \propto f(\lambda \mid X)/f(\lambda)$ and the fusion rule becomes

$$f(\lambda \mid X_1, X_2) \propto f(\lambda \mid X_1) f(\lambda \mid X_2)/f(\lambda). \tag{4}$$

The existence of different agents with different priors and likelihoods is maybe the most compelling argument to open the possibility for robust Bayesian analysis, where the likelihood and prior sets would in the first approximation be the convex closure of the likelihoods and prior of different experts.

3. WHAT IS REQUIRED FOR SUCCESSFUL APPLICATION OF BAYES METHOD?

The formula (1) is deceptively simple, and hides the complexity of a real world application where many engineering compromises are inevitable. Nevertheless, any method claimed to be Bayesian must relate to (1) and include all substantive application knowledge in the parameter and observation spaces, the likelihood and the prior. It is in general quite easy to show the Bayesian

method to be better or worse than an alternative by not including relevant and necessary application knowledge in (1) or in the alternative method. Let us illustrate this by an analysis of the comparison made in [56]. The problem is to track and classify a single target. The tracking problem is solved with a dynamic version of Bayes method, known as the Bayesian Chapman-Kolmogorov relationship:

$$f(\lambda_t \mid D_t) \propto f(d_t \mid \lambda_t) \int f(\lambda_t \mid \lambda_{t-1}) f(\lambda_{t-1} \mid D_{t-1}) d\lambda_{t-1}$$

$$f(\lambda_0 \mid D_0) = f(\lambda_0). \tag{5}$$

Here $D_t = (d_1, \ldots, d_t)$ is the sequence of observations obtained at different times, and $f(\lambda_t \mid \lambda_{t-1})$ is the maneuvering (process innovation) noise assumed. The latter is a probability distribution function (pdf) over state λ_t dependent on the state at the previous time-step, λ_{t-1}. When tracking targets that display different levels of maneuvering like transportation, attack and dog-fight for a fighter airplane, it has been found appropriate to apply (5) with different filters with levels of innovation noise corresponding to the maneuvering states, and to declare the maneuvering state that corresponds to the best matching filter. In the paper [56] the same method is proposed for a different purpose, namely the classification of aircraft (civilian, bomber, fighter) based on their acceleration capabilities. This is done by ad hoc modifications of (5) that do not seem to reflect substantive application knowledge, namely that the true target class is unlikely to change, and hence does not work well. The Bayesian solution to this problem would involve looking at (5) with a critical mind. Since we want to jointly track and classify, the state space should be, e.g., $P \times V \times C$, where P and V are position and velocity spaces and C is the class set, $\{c, b, f\}$. The innovation process should take account of the facts that the target class in this case does not change, and that the civilian and bomber aircraft have bounded acceleration capacities. This translates to two requirements on the process innovation component $f(\lambda_t \mid \lambda_{t-1})$ that (assuming unit time sampling):

$$f((p_t, v_t, c_t) \mid (p_{t-1}, v_{t-1}, c_{t-1})) = 0 \quad \text{if} \quad c_t \neq c_{t-1}$$

$$f((p_t, v_t, k) \mid (p_{t-1}, v_{t-1}, k)) = 0 \quad \text{if} \quad |v_t - v_{t-1}| > a_k$$

where a_k is the highest possible acceleration of target class k. Such an innovation term can be (and often is) described by a Gaussian with variance tuned to a_k, or by a bank of Gaussians. With this innovation term, the observation of a high acceleration dampens permanently the marginal probability of having a target class incapable of such acceleration. This is the natural Bayesian approach to the joint tracking and classification problems. Similar effects can be obtained in the robust Bayes and TBM [56] frameworks. As a contrast, the experiments reported by Oxenham et al. [44] use an appropriate innovation term and also give more reasonable results, both for the TBM and the Bayesian Chapman Kolmogorov approaches. The above is not meant as an argument that one of the two approaches compared in [56] is the preferred one. Our intention is rather to suggest that appropriate modeling may be beneficial for both approaches.

The range of applications where an uncertainty management problem is approached using (1) or (5) is extremely broad. In the above example, the parameter λ consists of one state vector (position and velocity vectors of a target) and its target label, thus the parameter space is (for 3D tracking) $R^6 \times C$ where C is a finite set of targets labels. In our main example, λ is just an indicator with three possible values. In many image processing applications, the parameter λ is the scene to be reconstructed from the data x, which is commonly called the film even if it is nowadays not registered on photographic film and is not even necessarily represented as a 2D image. This approach has been found excellent both for ordinary camera reconstruction problems and for special types of cameras as exemplified by Positron Emission Tomography and functional Magnetic Resonance Imaging, the type of camera and reconstruction objective having a profound influence on the choice of likelihood and priors, see [3, 27]. In genetic investigations, complex Bayesian models are also used a lot, and here the parameter λ could be a description of how reproduction in a set of individuals in a family has been produced by selection of chromosomes from parents, the positions of crossovers and the position of one or more hypothesized disease-causing gene(s), whereas the data are the genotypes and disease status of individuals, plus individual covariates that may environmentally influence development of disease. For a unified treatment of this problem family, see [14]. Another fascinating example is Bayesian identification of state space dynamics in time series, where the parameter is the time series of invisible underlying states, a signaling distribution (output distribution as a function of latent state) and the state change probability distributions [59].

Characteristic of cases where (1) and (5) are not as easily accepted is the presence of two different kinds of uncertainty, often called aleatory and epistemic uncertainty, where the former can be called "pure randomness" as one perceives dice (Latin: alea) throwing, while the latter is caused by "lack of knowledge" (from the Greek word for knowledge, episteme). Although one can argue about the relevance of this distinction, application owners have typically a strong sense of the distinction, particularly in risk assessment. The consequence is that the concepts of well-defined priors and likelihoods can be, and have been, questioned. The Bayesian answer to this critique is robust Bayesian analysis.

4. ROBUST BAYES AND EVIDENCE THEORY

In (global) robust Bayesian analysis [5, 36], one acknowledges that there can be ambiguity about the prior

and sampling distributions, and it is accepted that a convex set of such distributions is used in inference. The idea of robust Bayesian analysis goes back to the pioneers of Bayesian analysis [17, 39], but the computational and conceptual complexities involved meant that it could not be fully developed in those days. Instead, a lot of effort went into the idea of finding a canonical and unique prior, an idea that seems to have failed except for finite problems with some kind of symmetry, where a natural generalization of Bernoulli's indifference principle has become accepted. The problem is that no proposed priors are invariant under arbitrary rescaling of numerical quantities or non-uniform coarsening or refinement of the current frame of discernment. The difficulty of finding precise and unique priors has been taken as an argument to use some other methods, like evidence theory. However, as we shall see, this is an illusion, and avoiding use of an explicit prior usually means implicit reliance on Bernoulli's principle of indifference anyway. Likewise, should there be an acceptable prior, it can and should be used both in evidence theory and in Bayesian theory. This was pointed out, e.g., in [6, ch. 3.4].

Convex sets of probability distributions can be arbitrarily complex. Such a set can be generated by mixing of a set of "corners" (called simplices in linear programming theory) and the set of corners can be arbitrarily large already for sets of probability distributions over three elements.

In evidence theory, the DS-structure is a representation of a belief over a frame of discernment (set of possible worlds) Λ (commonly called the frame of discernment Θ in evidence theory) by a probability distribution m over its power-set (excluding the empty set), a basic probability assignment bpa, basic belief assignment bba, bma, or DS-structure (terminology is not stable, we will use DS-structure). The sets assigned non-zero probability in a DS-structure are called its focal elements, and those that are singletons are called atoms. A DS-structure with no mass assigned to non-atoms is a precise (sometimes called Bayesian) DS-structure. Even if it is considered important in many versions of DS theory not to equate a DS-structure with a set of possible distributions, such a perspective is prevalent in tutorials (e.g., [30, ch. 7] and [8, ch. 8]), explicit in Dempster's work [18], and almost unavoidable in a teaching situation. It is also compellingly suggested by the common phrase that the belief assigned to a non-singleton can flow freely to its singleton members, and the equivalence between a DS-structure with no mass assigned to non-singletons and the corresponding probability distribution [55]. Among publications elaborating on the possible difference between probability and other numerical uncertainty measures are [32, 55, 20].

A DS-structure seen as a set of distributions is a type of Choquet capacity, and these capacities form a particularly concise and flexible family of sets of distributions (the full theory of Choquet capacities is rich and of no immediate importance for us—we use the term capacity interpretation only to indicate a set of distributions obtained from a DS-structure in a way we will define precisely). Interpreting DS-structures as sets of probability distributions entails saying that the probability of a union of outcomes $e \subset \Lambda$ lies between the belief of e ($\sum_{w \subset e} m(w)$) and the plausibility of e ($\sum_{w \cap e \neq \emptyset} m(w)$). The parametric representation of the family of distributions it can represent, with parameters α_{ew}, $e \in 2^\Lambda$, $w \in \Lambda$, is $P(w) = \sum_e \alpha_{ew} m(e)$, all $w \in \Lambda$, where $\alpha_{ew} = 0$ if $w \notin e$, $\sum_{w \in e} \alpha_{ew} = 1$, and all α_{ew} are non-negative. This representation is used in Blackman and Popoli [8, ch. 8.5.3]. The pignistic transformation used in evidence theory to estimate a precise probability distribution from a DS-structure is obtained by making the α_{ew} equal for each e, $\alpha_{ew} = 1/|e|$ if $w \in e$. The relative plausibility transformation proposed by, among others, Voorbraak [60], Cobb and Shenoy [12, 13], on the other hand, is the result of normalizing the plausibilities of the atoms in Λ. It is also possible to translate a pdf over Λ to a DS-structure. Indeed, a pdf is already a (precise) DS-structure, but Sudano [58] studied inverse pignistic transformations that result in non-precise DS-structures by coarsening. They have considerable appeal but are not in the main line of argumentation in this paper [58].

It is illuminating to see how the pignistic and relative plausibility transformations emerge from a precise Bayesian inference: The observation space can in this case be considered to be 2^Λ, since this represents the only distinction among observation sets surviving from the likelihoods. The likelihood will be a function $l : 2^\Lambda \times \Lambda \to R$, the probability of seeing evidence e given world state λ. Given a precise $e \in 2^\Lambda$ as observation and a uniform prior, the inference over Λ would be $f(\lambda \mid e) \propto l(e, \lambda)$, but since we in this case have a probability distribution over the observation space, we should use (2), weighting the likelihoods by the masses of the DS-structures. Applying the indifference principle, $l(e, \lambda)$ should be constant for λ varying over the members of e, for each e. The other likelihood values ($\lambda \notin e$) will be zero. Two natural choices of likelihood are $l_1(e, \lambda) \propto 1$ and $l_2(e, \lambda) \propto 1/|e|$, for $\lambda \in e$. Amazingly, these two choices lead to the relative plausibility transformation and to the pignistic transformation, respectively:

$$f_i(\lambda \mid m) \propto \sum_{\{e:\lambda \in e\}} m(e) l_i(e, \lambda) = \begin{cases} \sum_{\{e:\lambda \in e\}} m(e) \Big/ \sum_e |e| m(e), & i = 1 \\ \sum_{\{e:\lambda \in e\}} m(e)/|e|, & i = 2. \end{cases} \tag{6}$$

Despite a lot of discussion, there seems thus to exist no fundamental reason to prefer one to the other, since

they result from two different and completely plausible statistical models and a common application of an indifference principle. The choice between the models (i.e., the two proposed likelihoods) can in principle be determined by (statistical) testing on the application's historic data.

The capacity corresponding to a DS-structure can be represented by $2^n - 2$ real numbers—the corresponding DS-structure is a normalized distribution over $2^n - 1$ elements (whereas an arbitrary convex set can need any number of distributions to span it and needs an arbitrary number of reals to represent it—thus capacities form a proper and really small subset of all convex sets of distributions).

It is definitely possible—although we will not elaborate it here—to introduce more complex but still consistent uncertainty management by going beyond robust Bayesianism, grading the families of distributions and introducing rules on how the grade of combined distributions are obtained from the grades of their constituents. The grade would in some sense indicate how plausible a distribution in the set is. It seems however important to caution against unnecessarily diving into the more sophisticated robust and graded set approaches to Bayesian uncertainty management.

Finally, in multi-agent systems we must consider the possibility of a gaming component, where an agent must be aware of the possible reasoning processes of other agents, and use information about their actions and goals to decide its own actions. In this case there appears to be no simple way to separate—as there is in a single agent setting—the uncertainty domain (what is happening?) from the decision domain (what shall I do?) because these get entangled by the uncertainties of what other agents will believe, desire and do. This problem is not addressed here, but can be approached by game-theoretic analyses, see, e.g., [9].

A Bayesian data fusion system or subsystem can thus use any level in a ladder with increasing complexity:

- Logic—no quantified uncertainty
- Precise Bayesian fusion
- Robust Bayesianism with DS-structures interpreted as capacities
- General robust Bayesianism (or lower/upper previsions)
- Robust Bayesianism with graded sets of distributions

Whether or not this simplistic view (ladder of Bayesianisms) on uncertainty management is tenable in the long run in an educational or philosophical sense is currently not settled. We will not further consider the first and the last rungs of the ladder.

4.1. Rounding

A set of distributions which is not a capacity can be approximated by rounding it to a minimal capacity that contains it (see Fig. 1), and this rounded set can be represented by a DS-structure. This rounding "upwards" is accomplished by means of lower probabilities (beliefs) of subsets of Λ. Specifically, in this example we list the minimum probabilities of all subsets of $\Lambda = \{A,B,C\}$ over the four corners of the polytope, to get lower bounds for the beliefs. These can be converted to masses using the Möbius inversion, or, in this simple example, manually from small to large events. For example, $m(A) = \text{bel}(A)$, $m(\{A,B\}) = \text{bel}(\{A,B\}) - m(A) - m(B)$, and $m(\{A,B,C\}) = \text{bel}(\{A,B,C\}) - m(\{A,B\}) - m(\{A,C\}) - m(\{B,C\}) - m(A) - m(B) - m(C)$. Since we have not necessarily started with a capacity, this may give negative masses to some elements. In that case, some mass must be moved up in the lattice to make all masses non-negative, and this can in the general case be done in several ways, but each way gives a minimal enclosing polytope. In the example, we have four corners, and the computation is shown in Table I. In this example we immediately obtain non-negative masses, and the rounded polytope is thus unique.

In the resulting up-rounded bba, when transforming it to a capacity, we must consider $2 * 2 * 3 = 12$ possible corner points. However, only five of these are actually corners of the convex hull in this case, and those are the corners visible in the enclosing capacity of Fig. 1. The other possible corner points turn out to lie inside, or inside the facets of, the convex hull. As an example, consider the lowest horizontal blue-dashed line; this is a facet of the polytope characterized by no mass flowing to B from the focal elements $\{A,C\}$, $\{B,C\}$ and $\{A,B,C\}$. The masses of $\{A,C\}$ and $\{A,B,C\}$ can thus be assigned either to A or to C. Assigning both to C gives the left end-point of the facet, both to A gives the right end-point, and assigning one to A and the other to C gives two interior points on the line.

It is also possible, using linear programming, to round downwards to a maximal capacity contained in a set. Neither type of rounding is unique, i.e., in general there may be several incomparable (by set inclusion) up- or down-rounded capacities for a set of distributions.

5. DECISIONS UNDER UNCERTAINTY AND IMPRECISION

The ultimate use of data fusion is usually decision making. Precise Bayesianism results in quantities—probabilities of possible worlds—that can be used immediately for expected utility decision making [49, 4]. Suppose the profit in choosing a from a set $\mathcal{A}$ of possible actions when the world state is λ is given by the utility function $u(a,\lambda)$ mapping action a and world state λ to a real valued utility (e.g., dollars). Then the action maximizing expected profit is $\arg\max_a \int u(a,\lambda) f(\lambda \mid x) d\lambda$. In robust Bayesian analysis one uses either minimax criteria or estimates a precise probability distribution to decide from. Examples of the latter are the pignistic and relative plausibility transformations. An example of

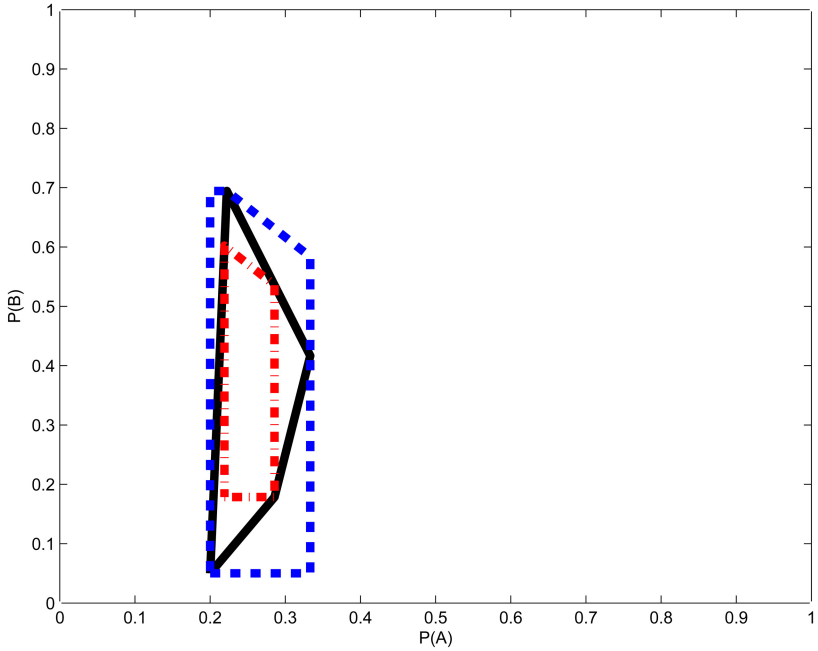


Fig. 1. Rounding a set of distributions over $\{A,B,C\}$. The coordinates are the probabilities of A and B. A set spanned by four corner distributions (black solid), its minimal enclosing (blue dashed), and one of its maximal enclosed (red dash-dotted), capacities.

TABLE I
Rounding a Convex Set of Distributions Given by its Corners*

Focal	Corners				min	m
A	0.200	0.222	0.333	0.286	0.200	0.200
B	0.050	0.694	0.417	0.179	0.050	0.050
C	0.750	0.083	0.250	0.536	0.083	0.083
$\{A,B\}$	0.250	0.916	0.750	0.465	0.250	0
$\{A,C\}$	0.950	0.305	0.583	0.822	0.305	0.022
$\{B,C\}$	0.800	0.777	0.667	0.715	0.667	0.534
$\{A,B,C\}$	1.000	1.000	1.000	1.000	1.000	0.111

*Corners of the black polygon of Fig. 1 are listed clockwise, starting at bottom left.

a decision-theoretically motivated estimate is the maximum entropy estimate, often used in robust probability applications [38]. This choice can be given a decision-theoretic motivation since it minimizes a game-theoretic loss function, and can also be generalized to a range of loss functions [28]. Specifically, a Decision maker must select a distribution q while Nature selects a distribution p from a convex set Γ. Nature selects an outcome x according to its chosen distribution p, and the decision makers loss is $-\log q(x)$. This makes the Decision maker's expected loss equal to $E_p\{-\log q(X)\}$. The minimum (over q) of the maximum (over p) expected loss is then obtained when q is chosen to be the maximum entropy distribution in Γ. Thus, if this loss function is accepted, it is optimal to use the maximum entropy transformation for decision making.

The maximum entropy principle differs significantly from the relative plausibility and pignistic transformations, since it tends to select a point on the boundary of a set of distributions (if the set does not contain the uniform distribution), whereas the pignistic transformation selects an interior point.

The pignistic and relative plausibility transformations are linear estimators, by which we mean that they are obtained by normalization of a linear function of the masses in the DS-structure. If we buy the concept of a DS-structure as a set of possible probability distributions, it would be natural to require that as estimate we choose a possible distribution, and then the pignistic transformation of Smets gets the edge—it is not difficult to prove the following:

PROPOSITION 1 *The pignistic transformation is the only linear estimator of a probability distribution from a DS-structure that is symmetric over Λ and always returns a distribution in the capacity represented by the DS-structure.*

Although we have no theorem to this effect, it seems as if the pignistic transformation is also a reasonable decision-oriented estimator approximately minimizing the maximum Euclidean norm of difference between the chosen distribution and the possible distributions, and better than the relative plausibility transformation as well as the maximum entropy estimate for this objective function. The estimator minimizing this maximum norm is the center of the smallest enclosing sphere. It will not be linear in m, but can be computed with some effort using methods presented, e.g., in [23]. The centroid is sometimes proposed as an estimator, but it does not correspond exactly to any known robust loss function—rather it is based on the assumption that the probability vector is uniformly distributed over the imprecision polytope.

The standard expected utility decision rule in precise probability translates in imprecise probability to producing an expected utility interval for each decision alternative, the utility of an action a being given by the interval $I_a = \bigcup_{f \in F} \int u(a,\lambda) f(\lambda \mid x) d\lambda$. In a refinement proposed by Voorbraak [61], decision alternatives are compared for each pdf in the set of possible pdfs: $I_{af} = \int u(a,\lambda) f(\lambda \mid x) d\lambda$, for $f \in F$. Decision a is now better than decision b if $I_{af} > I_{bf}$ for all $f \in F$.

Some decision alternatives will fall out because they are dominated in utility by others, but in general several possible decisions with overlapping utility intervals will remain. In principle, if no more information exists, any of these decisions can be considered right. But they are characterized by larger or smaller risk and opportunity.

6. ZADEH'S EXAMPLE

We will now discuss our problem in the context of Zadeh's example of two physicians who investigated a patient independently—a case prototypical, e.g., for the important fusion for target classification problem. The two physicians agree that the problem (the diagnosis of the patient) is within the set $\{M, C, T\}$, where M is Meningitis, C is Concussion and T is brain Tumor. However, they express their beliefs differently, as a probability distribution which is $(0.99, 0, 0.01)$ for the first physician and $(0, 0.99, 0.01)$ for the second. The question is what a third party can say about the patients condition with no more information than that given. If the two expert opinions are taken as likelihoods, or as posteriors with a common uniform prior, this problem is solved by taking Laplace's parallel composition (1) of the two probability vectors, giving the result $(0, 0, 1)$, i.e., the case T is certain. This example has been discussed a lot in the literature, see e.g. [53]. It is a classical example on how two independent sets of observations can together eliminate cases to end up with a case not really indicated by any of the two sets in separation. Several such examples have been brought up as good and prototypical in the Bayesian literature, e.g., in [38]. However, in the evidence theory literature the Bayesian solution (which is also obtained from using Dempster's and the Modified Dempster's rule) has been considered inadequate and this particular example has been the starting point for several proposals of alternative fusion rules.

The following are reactions I have met from professionals—physicians, psychiatrists, teachers and military commanders—confronted with similar problems. They are also prototypical for current discussions on evidence theory.

- One of the experts probably made a serious mistake.
- These experts seem not to know what probability zero means, and should be sent back to school.
- It is completely plausible that one eliminated M and the other C in a sound way. So T is the main alternative, or rather T or something else, since there are most likely more possibilities left.
- It seems as if estimates are combined at a too coarse level: it is in this case necessary to distinguish in Λ between different cases of the three conditions that are most likely to effect the likelihoods from observations: type, size and position of tumor, bacterial, viral or purely inflammatory meningitis, position of concussion. The frame of discernment should thus not be determined solely from the frame of interest, but also on what one could call homogeneity of likelihoods or evidence.
- The assessments for T are probably based mostly on prior information (rareness) or invisibility in a standard MR scan, so the combined judgment should not make T less likely, rather the opposite.
- An investigation is always guided by the patient's subjective beliefs, and an investigation affects those beliefs. So it is implausible that the two investigations of the same patient are "really" independent. This is a possible explanation for the Ulysses syndrome, where persons are seen to embark on endless journeys through the health care system. This view would call for a game-theoretic approach (with parameters difficult to assess).

What the example reactions teach us is that subjects confronted with paradoxical information typically start building their own mental models about the case and insist on bringing in more information, in the form of information about the problem area, the observation protocols underlying the assessments, a new investigation, or pure speculation. The professionals handling of the information problem is usually rational enough, but very different conclusions arise from small differences in mental models. This is a possible interpretation of the prospect theory of Kahneman and Tversky [40].

To sum things up, if we are sure that the experts are reliable and have the same definitions of the three neurological conditions, the result given by Bayes' and Dempster's rules are appropriate. If not, the assumptions and hence the statistical model must be modified. It seems obvious that the decision makers belief in the experts reliability must be explicitly elicited in similar situations.

7. FUSION IN EVIDENCE AND ROBUST BAYESIAN THEORY

The Dempster-Shafer combination rule [51] is a straightforward generalization of Laplace's parallel composition rule. By this statement we do not claim that this is the way DS theory is usually motivated. But the model in which Dempster's rule is motivated [18] is different from ours: there it is assumed that each source has its own possible world set, but precise beliefs about it. The impreciseness results only from a multi-valued mapping, ambiguity in how the information of

the sources should be translated to a common frame of discernment. It is fairly plausible that the information given by the source is well representable as a DS structure interpreted as a capacity. What is much less plausible is that the information combined from several sources is well captured by Dempster's rule rather than by the Fixsen/Mahler combination rule or the robust combination rule to be described shortly. The precise assumptions behind Dempster's rule are seldom explained in tutorials and seem not well known, so we recapitulate them tersely: It is assumed that evidence comes from a set of sources, where source i has obtained a precise probability estimate p_i over its private frame X_i. This information is to be translated into a common frame Λ, but only a multi-valued mapping Γ_i is available, mapping elements of X_i to subsets of Λ. For the tuple of elements $x_1,\ldots,x_n$, their joint probability could be guessed to be $p_1(x_1)\cdots p_n(x_n)$, but we have made assumptions such that we know that this tuple is only possible if $\Gamma_1(x_1)\cap\cdots\cap\Gamma_n(x_n)$ is non-empty. So the probabilities of tuples should be added to the corresponding subset of Λ probabilities, and then conditioning on non-emptiness should be performed and the remaining subset probabilities normalized, a simple application of (1). From these assumptions Dempster's rule follows.

This is postulated by Dempster as the model required. One can note that it is not based on inference, but derived from an explicit and exact probability model. It was claimed incoherent (i.e. violating the consistent betting paradigm) by Lindley [42], but Goodman, Nguyen and Rogers showed that it is not incoherent [25]. Indeed, the assumption of multi-valued mappings seems completely innocent, if somewhat arbitrary, and it would be unlikely to lead to inconsistencies. The recently introduced Fixsen/Mahler MDS combination rule [22] involves a re-weighting of the terms involved in the set intersection operation: whereas Dempster's combination rule can be expressed as

$$m_{\mathrm{DS}}(e)\propto\sum_{e=e_1\cap e_2} m_1(e_1)m_2(e_2),\qquad e\neq\emptyset \tag{7}$$

the MDS rule is

$$m_{\mathrm{MDS}}(e)\propto\sum_{e=e_1\cap e_2} m_1(e_1)m_2(e_2)\frac{|e|}{|e_1||e_2|},\qquad e\neq\emptyset. \tag{8}$$

The MDS rule was introduced to account for non-uniform prior information about the world and evidence that contains prior information common to all sources. In this case $|e|$, etc, in the formula are replaced by the prior probabilities of the respective sets. The rule (8) is completely analogous to (4): the denominator of the correction term takes the priors out of the posteriors of both operands, and the numerator $|e|$ reinserts it once in the result. But as we now will see, the MDS rule can also be considered a natural result of fusing likelihood describing information with a different likelihood function.

It is possible to analyze the source fusion problem in a (precise) Bayesian setting. If we model the situation with the likelihoods on $2^\Lambda\times\Lambda$ of (6), Section 4, we find the task of combining the two likelihoods $\sum_e m_1(e)l(e,\lambda)$ and $\sum_e m_2(e)l(e,\lambda)$ using Laplace's parallel composition as in (2) over Λ, giving

$$f(\lambda)\propto\sum_{e_1,e_2} m_1(e_1)m_2(e_2)l_i(e_1,\lambda)l_i(e_2,\lambda).$$

For the choice $i=1$, this gives the relative plausibility of the result of fusing the evidences with Dempster's rule; for the likelihood l_2 associated with the pignistic transformation, we get $\sum_{e_1,e_2} m_1(e_1)m_2(e_2)l(e_1,\lambda)l(e_2,\lambda)/(|e_1||e_2|)$. This is the pignistic transformation of the result of combining m_1 and m_2 using the MDS rule. In the discussions for and against different combination and estimation operators, it has sometimes been claimed that the estimation operator should propagate through the combination operator. This claim is only valid if the above indicated precise Bayesian approach is bought, which would render DS-structures and convex sets of distributions unnecessary. In the robust Bayesian framework, the maximum entropy estimate is completely kosher, but it does not propagate through any well known combination operation. The combination of Dempster's rule and the pignistic transformation cannot easily be defended in a precise Bayesian framework, but Dempster's rule can be defended under the assumption of multi-valued mappings and reliable sources, whereas the pignistic transformation can be defended in three ways: (1) It can be seen as "natural" since it results, e.g., from an indifference principle applied to the parametric representation of Blackman and Popoli; (2) Smets argument [54] is that the estimation operator (e.g., the pignistic transformation) should propagate, not through the combination operator, but through linear mixing; (3) An even more convincing argument would relate to decisions made, e.g., it seems as if the pignistic transformation is, not exactly but approximately, minimizing the norm of the maximum (over Nature's choice) error made measured as the Euclidean norm of the difference between the selected distribution and Nature's choice.

7.1 The Robust Combination Rule

The combination of evidence—likelihood functions normalized so they can be seen as probability distributions—and a prior over a finite space is thus done simply by component-wise multiplication followed by normalization [41, 57]. The resulting combination operation agrees with the DS and the MDS rules for precise beliefs. The robust Bayesian version of this would replace the probability distributions by sets of probability distributions, for example represented as DS-structures. The

most obvious combination rule would yield the set of probability functions that can be obtained by taking one member from each set and combining them. Intuitively, membership means that the distribution can possibly be right, and we would get the final result, a set of distributions that can be obtained by combining a number of distributions each of which could possibly be right. The combination rule (3) would thus take the form (where F denotes convex families of functions):

$$F(\lambda \mid \{X_1, X_2\}) \propto F(\{X_1, X_2\} \mid \lambda) \times F(\lambda) = F(X_1 \mid \lambda) \times F(X_2 \mid \lambda) \times F(\lambda). \quad (9)$$

DEFINITION 1 The robust Bayesian combination operator $\times$ combines two sets of probability distributions over a common space Λ. The value of $F_1 \times F_2$ is $\{cf_1f_2 : f_1 \in F_1, f_2 \in F_2, c = 1/\sum_{\lambda\in\Lambda} f_1(\lambda)f_2(\lambda)\}$.

The operator can easily be applied to give too much impreciseness, for reasons similar to the corresponding problem in interval arithmetic: the impreciseness of likelihood functions has typically a number of sources, and the proposed technique can give too large uncertainties when these sources do not have their full range of variation within the evidences that will be combined. A most extreme example is the sequence of plots returned by a sensor: variability can have its source in the target, in the sensor itself, and in the environment. But when a particular sensor follows a particular target, the variability of these sources are not fully materialized. The variability has its source only in the state (distance, inclination, etc) of the target, so it would seem wasteful to assume that each new plot comes from an arbitrarily selected sensor and target. This, and similar problems, are inherent in system design, and can be addressed by detailed analyses of sources of variation, if such are feasible.

We must now explain how to compute the operator of Definition 1. The definition given of the robust Bayesian combination operator involves infinite sets in general and is not computable directly. For singleton sets it is easily computed, though, with Laplace's parallel composition rule. It is also the case that every corner in the resulting set can be generated by combining two corners, one from each of the operands. This observation gives the method for implementation of the robust operator. After the potential corners of the result have been obtained, a convex hull computation as found, e.g., in MATLAB and OCTAVE, is used to tessellate the boundary and remove those points falling in the interior of the polytope. The figures of this paper were produced by a Matlab implementation of robust combination, Dempster's and the MDS rule, maximum entropy estimation, and rounding. The state of the art in computational geometry software thus allows easy and efficient solutions, but of course as the state space and/or the number of facets of the imprecision polytopes become very large, some tailored approximation methods will be called for. The DS and MDS rules have exponential complexity in the worst case. The robust rule will have a complexity quadratic in the number of corners of the operands, and will thus depend on rounding for feasibility. For very high-dimensional problems additional pruning of the corner set will be necessary (as is also the case with the DS and MDS operators).

We can now make a few statements, most of which are implicitly present in [19, Discussion by Aitchison] and [32], about fusion in the robust Bayesian framework:

- The combination operator is associative and commutative, since it inherits these properties from the multiplication operator it uses.
- Precise beliefs combined gives the same result as Dempster's rule and yield new precise beliefs.
- A precise belief combined with an imprecise belief will yield an imprecise belief in general—thus Dempster's rule underestimates imprecision compared to the robust operator.
- Ignorance is represented by a uniform precise belief, not by the vacuous assignment of DS-theory.
- The vacuous belief in the robust framework is a belief that represents total skepticism, and will when combined with anything yield a new vacuous belief (it is thus an absorbing element). This belief has limited use in the robust Bayesian context.
- Total skepticism cannot be expressed with Dempster's rule, since it never introduces a focal element which is a superset of all focal elements in one operand.

DEFINITION 2 A rounded robust Bayesian combination operator combines two sets of probability distributions over a common space Λ. The robust operation is applied to the rounded operands, and the result is then rounded.

An important and distinguishing property of the robust rule is:

OBSERVATION 1 *The robust combination operator is, and the rounded robust operator can be made (note: it is not unique) monotone with respect to imprecision, i.e., if $F_i' \subseteq F_i$, then $F_1' \times F_2' \subseteq F_1 \times F_2$.*

PROPOSITION 2 *For any combination operator $\times'$ that is monotone* wrt *imprecision and is equal to the Bayesian (Dempster's) rule for precise arguments, $F_1 \times F_2 \subseteq F_1 \times' F_2$, where $\times$ is the robust rule.*

PROOF By contradiction; thus assume there is an $f \in F_1 \times F_2$ with $f \notin F_1 \times' F_2$. By the definition of $\times$, $f = \{f_1\} \times \{f_2\}$ for some $f_1 \in F_1$ and $f_2 \in F_2$. But then $f = \{f_1\} \times' \{f_2\}$, and since $\times'$ is monotone wrt imprecision, $f \in F_1 \times' F_2$, a contradiction.

We can also show that the MDS combination rule has the "nice" property of giving a result that always overlaps the robust rule result, under the capacity interpretation of DS-structures:

PROPOSITION 3 *Let m_1 and m_2 be two DS-structures and let F_1 and F_2 be the corresponding capacities. If F is the*

capacity representing $m = m_1 *_{\text{MDS}} m_2$ *and* F' *is* $F_1 \times F_2$*, then* F *and* F' *overlap.*

PROOF Since the pignistic transformation propagates through the MDS combination operator, and by Proposition 1 the pignistic transformation is a member of the capacity of the DS-structure, the parallel combination of the pignistic transformations of m_1 and m_2 is a member of F' and equal to the pignistic transformation of m, which for the same reason is a member of F. This concludes the proof.

The argument does not work for the original Dempster's rule, for reasons that will become apparent in the next section. It was proved by Jaffray [37] that Dempster's rule applied with one operand being precise gives a (precise) result inside the robust rule polytope. The same holds of course, by Proposition 3, for the MDS rule. We can also conjecture the following, based on extensive experimentation with our prototype implementation, but have failed in obtaining a short convincing proof:

CONJECTURE 1 *The MDS combination rule always gives a result which is, in the capacity interpretation, a subset of the robust rule result. The MDS combination rule is also a coarsest symmetric bilinear operator on DS-structures with this property.*

8. A PARADOXICAL EXAMPLE

In [1] we analyzed several versions of Zadeh's example with "discounted" evidences to illustrate the differences between robust fusion and the DS and MDS rules, as well as some different methods to summarize a convex set of pdfs as a precise pdf. Typically, the DS and MDS rules give much smaller imprecision in the result than the robust rule, which can be expected from their behavior with one precise and one imprecise operand. One would hope that the operators giving less imprecision would fall inside the robust rule result, in which case one would perhaps easily find some plausible motivation for giving less imprecision than indicated in the result. In practice this would mean that a system using robust fusion would sometimes find that there is not a unique best action while a system based on the DS or MDS rule would pick one of the remaining actions and claim it best, which is not obviously a bad thing. However, the DS, MDS and robust rules do not only give different imprecision in their results, they are also pairwise incompatible (sometimes having an empty intersection) except for the case mentioned in Conjecture 1. Here we will concentrate on a simple, somewhat paradoxical, case of combining two imprecise evidences and decide from the result.

Varying the parameters of discounting a little in Zadeh's example, it is not difficult to find cases where Dempster's rule gives a capacity disjoint (regarded as a geometric polytope) from the robust rule result. A simple Monte Carlo search indicates that disjointness does indeed happen in general, but infrequently. Typically, Dempster's rule gives an uncertainty polytope that is clearly narrower than that of the robust rule, and enclosed in it. In Fig. 2 we show an example where this is not the case. The two combined evidences are imprecise probabilities over three elements A, B and C, the first spanned by the probability distributions $(0.2, 0.2, 0.6)$ and $(0.2, 0.5, 0.3)$, the second by $(0.4, 0.1, 0.5)$ and $(0.4, 0.5, 0.1)$. These operands can be represented as DS structures, as shown in Table II, and they are shown as vertical green lines in Fig. 2. They can be combined with either the DS rule, the MDS rule, or the robust rule, as shown in Table III. The situation is illustrated in Fig. 2, where all sets of pdfs are depicted as lines or polygons projected on the first two probabilities. The figure shows that the robust rule claims the probability of the first event A (horizontal axis) to be between 0.2 and 0.33, whereas Dempster's rule would give it an exact probability around 0.157. The MDS rule gives a result that falls nicely inside the robust rule result, but it claims an exact value for the probability of A, namely 0.25. Asked to bet with odds six to one on the first event (by which we mean that the total gain is six on success and the loss is one on failure), the DS rule says decline, the robust and MDS rules say accept. For odds strictly between four and five to one, the robust rule would hesitate and MDS would still say yes. For odds strictly between three and four to one, DS and MDS would decline whereas the robust rule would not decide for or against. Including the refinement proposed by Voorbraak (see Section 5) would not alter this conclusion unless the imprecisions of the two operands were coupled, e.g., by common dependence on a third quantity.

In an effort to reconcile Bayesian and belief methods, Blackman and Popoli [8, ch. 7] propose that the result of fusion should be given the capacity interpretation as a convex set, whereas the likelihoods should not—an imprecise likelihood should instead be represented as the coarsest enclosing DS-structure having the same pignistic transformation as the original one. When combined with Dempster's rule, the result is again a prior for the next combination whose capacity interpretation shows its imprecision. The theorem proved—at some length—in [8, App. 8A] essentially says that this approach is compatible with our robust rule for precise likelihoods. In our example, if the second operand is coarsened to $\{m_2'(A) \mapsto 0.1, m_2'(\{A,B,C\}) \mapsto 0.9\}$, the fusion result will be a vertical line at 0.217, going from 0.2 to 0.49, just inside the robust rule result. However no mass will be assigned to a non-singleton set containing A, so the rule still gives a precise value to the probability of A. The philosophical justification of this approach appears weak.

The example shows that Dempster's rule is not compatible with the capacity interpretation, whereas the MDS rule is: there is no pair of possible pdfs for the operands that combine to any possible value in the

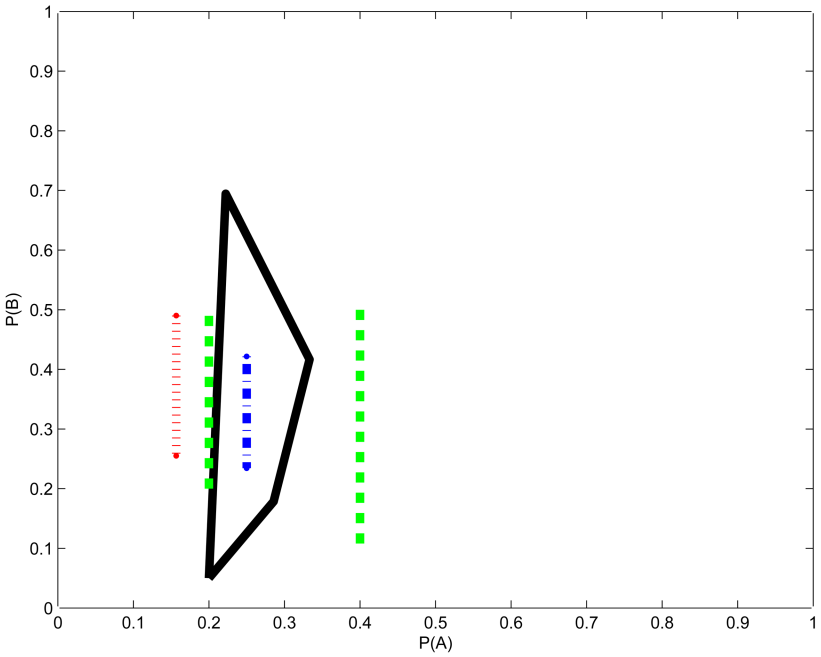


Fig. 2. A case where the robust rule and Dempster's rule give paradoxical results. The coordinates are the probabilities of A and B. The operands are shown in green dashed, the result of the robust combination rule is shown in black solid (same as in Fig. 1), Dempster's rule gives the result shown in red dotted, the Fixsen/Mahler MDS rule shown in blue dash-dotted lines.

Dempster's rule result, wheras every possible pdf in the MDS rule results from combining some pair of possible pdfs for the operands. If Conjecture 1 can be proved, the last is true for all pairs of operands, but there are also many particular examples where even Dempster's rule gives a compatible result. It has been noted by Walley that Dempster's rule is not the same as the robust combination rule [62], but I have not seen a demonstration that the two are incompatible in the above sense. There is, of course, a rational explanation of the apparent paradox, namely that the assumptions of private frames of discernment for sources and of a multi-valued mapping for each source is very different from the assumption of imprecise likelihoods, and this means that some informa-

TABLE II
Two Operands of the Paradoxical Example*

Focal	op_1			op_2		
	c_1	c_2	m	c_1	c_2	m
A	0.2	0.2	0.2	0.4	0.4	0.4
B	0.2	0.5	0.2	0.1	0.5	0.1
C	0.6	0.3	0.3	0.5	0.1	0.1
$\{B,C\}$			0.3			0.4

*Columns marked m denote DS-structures and those marked c_1, c_2 denote corners spanning the corresponding capacity. Values are exact.

TABLE III
Fusing the Operands of Table II with the DS, MDS and Robust Rules*

Focal	Fusion Result										
	DS			MDS			Robust				Uprounded
	c_1	c_2	m	c_1	c_2	m	c_{11}	c_{22}	c_{12}	c_{21}	m
A	0.157	0.157	0.157	0.250	0.250	0.250	0.200	0.222	0.333	0.286	0.200
B	0.255	0.490	0.255	0.422	0.234	0.234	0.050	0.694	0.417	0.179	0.050
C	0.588	0.353	0.353	0.328	0.516	0.328	0.750	0.083	0.250	0.536	0.083
$\{A,B\}$			0			0					0
$\{A,C\}$			0			0					0.022
$\{B,C\}$			0.235			0.188					0.534
$\{A,B,C\}$			0			0					0.111

*The result for DS and MDS shown as two corners (c_1 and c_2), and as an equivalent DS-structure (m). For the robust rule result, its four spanning corners are shown, where, e.g., c_{21} was obtained by combining the second corner c_2 of op_1 with c_1 of op_2, etc. These corners are the corners of the black polygon in Fig. 2. The robust rule result is also shown as a DS-structure for the up-rounded result (blue dashed line in Fig. 1). Values are rounded to three decimals.

tion in the private frames is still visible in the end result when Dempster's rule is used. Thus Dempster's rule effectively makes a combination in the frame 2^Λ instead of in Λ as done by the robust rule. It is perhaps more surprising that the paradoxical result is also obtainable in the frame Λ using precise Bayesian analysis and the likelihood $l_1(e, \lambda)$ (see Section 4). The main lesson here, as in other places, is that we should not use Dempster's rule unless we have reason to believe that imprecision is produced by the multi-valued mapping of Dempster's model rather than Fixsen/Mahler's model or incomplete knowledge of sampling functions and prior. If the MDS operator is used to combine likelihoods or a likelihood and a prior, then posteriors should be combined using the MDS rule (8), but with all set cardinalities squared.

Excluding Bayesian thinking from fusion may well lead to inferior designs.

9. CONCLUSIONS

Despite the normative claims of evidence theory and robust Bayesianism, the two have been considered different in their conclusions and general attitude towards uncertainty. The Bayesian framework can however describe most central features of evidence theory, and is thus a useful basis for teaching and comparison of different detailed approaches to information fusion. The teaching aspect is not limited to persuading engineers to think in certain ways. For higher level uncertainty management, dealing with quantities recognizable to users like medical researchers, military commanders, and their teachers in their roles as evaluators, the need for clarity and economy of concepts cannot be exaggerated. The arguments put forward above suggest that an approach based on the precise Bayesian and the robust Bayesian fusion operator is called for, and that choosing decision methods based on imprecise probabilities or DS structures should preferably be based on decision-theoretic arguments. Our example shows how dangerous it can be to apply evidence theory without investigating the validity in an application of its crucial assumption of reliable private frames for all sources of evidence and precise multi-valued mappings from this frame to the frame of interest. The robust rule seems to give a reasonable fit to most fusion rules based on different statistical models, with the notable exception of Dempster's rule. Thus, as long as the capacity interpretation is prevalent in evidence theory applications, there are good reasons to consider if the application would benefit from using the MDS rule (complemented with priors if available) also for combining information in the style of likelihoods. In this case, however, the combination of the MDS rule with pignistic transformation is interpretable as a precise Bayesian analysis. In most applications I expect that the precise Bayesian framework is adequate, and it is mainly in applications with the taste of risk analysis that the robust Bayesian framework will be appropriate.

ACKNOWLEDGMENTS

Discussions with members of the fusion group at the Swedish Defence Research Agency (FOI), students in the decision support group at KTH, and colleagues at Saab AB, Karolinska Institutet and the Swedish National Defense College (FHS) have been important for clarifying ideas presented above. The referees have further made clear the need to clarify the argumentation, and also by their comments made me strengthen my claims somewhat.

REFERENCES

[1] S. Arnborg
Robust Bayesianism: Imprecise and paradoxical reasoning. In P. Svensson and J. Schubert (Eds.), *Proceedings of the Seventh International Conference on Information Fusion*, Vol. I, Stockholm, Sweden, International Society of Information Fusion, June 2004, 407–414.

[2] S. Arnborg and G. Sjödin
Bayes rules in finite models.
In *Proceedings of European Conference on Artificial Intelligence*, Berlin, 2000, 571–575.

[3] R. G. Aykroyd and P. J. Green
Global and local priors, and the location of lesions using gamma camera imagery.
Philosophical Transactions of the Royal Society of London A, **337** (1991), 323–342.

[4] J. O. Berger
Statistical Decision Theory and Bayesian Analysis.
New York: Springer-Verlag, 1985.

[5] J. O. Berger
An overview of robust Bayesian analysis (with discussion).
Test, **3** (1994), 5–124.

[6] N. Bergman
Recursive Bayesian Estimation.
Ph.D. thesis, Linköping University, Linköping, 1999.

[7] J. M. Bernardo and A. F. Smith
Bayesian Theory.
New York: Wiley, 1994.

[8] S. Blackman and R. Popoli
Design and Analysis of Modern Tracking Systems.
Boston, London: Artech House, 1999.

[9] J. Brynielsson and S. Arnborg
Bayesian games for threat prediction and situation analysis.
In P. Svensson and J. Schubert (Eds.), *Proceedings of the Seventh International Conference on Information Fusion*, Vol. II, Stockholm, Sweden, International Society of Information Fusion, June 2004, 1125–1132.

[10] S. Challa and D. Koks
Bayesian and Dempster-Shafer fusion.
Sãdhãna, 2004, 145–174.

[11] B. Cobb and P. Shenoy
A comparison of Bayesian and belief function reasoning.
Technical Report, University of Kansas School of Business, 2003.

[12] B. Cobb and P. Shenoy
A comparison of methods for transforming belief function models to probability models.
In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, Vol. 2711, *LNCS*, Berlin: Springer, 2004, 255–266.

[13] B. Cobb and P. Shenoy
On the plausibility transformation method for translating belief function models to probability models.
International Journal of Approximate Reasoning, to appear.

[14] J. Corrander and M. Sillanpää
A unified approach to joint modeling of multiple quantitative and qualitative traits in gene mapping.
Journal of Theor. Biology, **218** (2002), 435–446.

[15] R. T. Cox
Probability, frequency, and reasonable expectation.
Am. Journal of Physics, **14** (1946), 1–13.

[16] G. de Cooman and M. Zaffalon
Updating beliefs with incomplete observations.
Artificial Intelligence, **159**, 1–2 (2004), 75–125.

[17] B. de Finetti
Theory of Probability.
London: Wiley, 1974.

[18] A. P. Dempster
Upper and lower probabilities induced by a multi-valued mapping.
Annals of Mathematical Statistics, **38** (1967), 325–339.

[19] A. P. Dempster
A generalization of Bayesian inference (with discussion).
Journal of the Royal statistical Society B, **30** (1968), 205–247.

[20] D. Dubois and H. Prade
Representing partial ignorance.
IEEE Transactions on Systems, Man, and Cybernetics, Pt. A, 1996, 361–377.

[21] S. Ferson, V. Kreinovich, L. Ginzburg, D. Myers, and K. Sentz
Constructing probability boxes and Dempster-Shafer structures.
Technical Report, Sandia National Laboratories, 2003.

[22] D. Fixsen and R. P. S. Mahler
The modified Dempster-Shafer approach to classification.
IEEE Transactions on SMC-A, **27**, 1 (Jan. 1997), 96–104.

[23] B. Gärtner
Fast and robust smallest enclosing balls.
In *ESA*, Vol. 1643, *LNCS*, Berlin: Springer, 1999, 325–338.

[24] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin
Bayesian Data Analysis (2nd ed.).
New York: Chapman & Hall, 2003.

[25] I. Goodman, H. T. Nguyen, and G. Rogers
On the scoring approach to admissibility of uncertainty measures in expert systems.
Journal of Math. Analysis Appl., **159** (1991), 550–594.

[26] P. J. Green
Reversible jump Markov chain Monte Carlo computation and Bayesian model determination.
Biometrika, **82** (1995), 711–732.

[27] P. J. Green
Markov Chain Monte Carlo in image analysis.
In W. R. Gilks, S. Richardson and D. J. Spiegelhalter (eds.), *Markov Chain Monte Carlo in Practice*, London: Chapman & Hall, 1996.

[28] P. D. Grünwald and A. P. Dawid
Game theory, maximum entropy, minimum discrepancy, and robust Bayesian decision theory.
Annals of Statistics, **32**, 4 (2004).

[29] V. Ha, A-H. Doan, V. Vu and P. Haddaway
Geometric foundations for interval-based probabilities.
Annals of Mathematics and Artificial Intelligence, 1998.

[30] D. L. Hall and J. Llinas
Handbook of Multisensor Data Fusion.
Boca Raton: CRC Press, May 2001.

[31] J. Halpern
A counterexample to theorems of Cox and Fine.
Journal of AI Research, **10** (1999), 67–85.

[32] J. Y. Halpern and R. Fagin
Two views of belief: Belief as generalized probability and belief as evidence.
Artificial Intelligence, **54** (1992), 275–318.

[33] D. Heath and W. Sudderth
On finitely additive priors, coherence, and extended admissibility.
Annals of Statistics, **6** (1978), 233–345.

[34] J. Helton and K. Weichselberger (Eds.)
Special issue on alternative representations of epistemic uncertainty.
Reliability Engineering and System Safety, 2004, 1–369.

[35] C. Howson and P. Urbach
Scientific Inference: The Bayesian Approach.
Chicago: Open Court Publishing Company, 1989.

[36] D. Rios Insua and F. Ruggeri (Eds.)
Robust Bayesian Analysis.
La Salle: Springer-Verlag, 2000.

[37] J. Jaffray
Bayesian updating and belief functions.
IEEE Transactions on Systems, Man, and Cybernetics, 1996, 1144–1152.

[38] E. T. Jaynes
Probability Theory: The Logic of Science.
Cambridge: Cambridge University Press, 2003.

[39] H. Jeffreys
Scientific Inference.
Cambridge: Cambridge University Press, 1931.

[40] D. Kahneman (Ed.)
Judgment Under Uncertainty: Heuristics and Biases.
Cambridge: Cambridge University Press, 1982.

[41] H. Kyburg and M. Pittarelli
Set-based Bayesianism.
IEEE Transactions on Systems, Man, and Cybernetics, 1996, 324–339.

[42] D. V. Lindley
Scoring rules and the inevitability of probability (with discussion).
Internat. Stat. Rev., **50** (1982), 1–26.

[43] R. Mahler
Can the Bayesian and Dempster-Shafer approaches be reconciled? Yes.
In *FUSION 2005*, International Society of Information Fusion, 2005, C7-3.

[44] M. G. Oxenham, S. Challa, and M. R. Morelande
Decentralised fusion of disparate identity estimates for shared situation awareness.
In P. Svensson and J. Schubert (Eds.), *Proceedings of the Seventh International Conference on Information Fusion*, vol. I, Stockholm, Sweden, International Society of Information Fusion, June 2004, 167–174.

[45] M. G. Oxenham, S. Challa, and M. R. Morelande
Fusion of disparate identity estimates for shared situation awareness in a network-centric environment.
Information Fusion, to appear.

[46] J. B. Paris
The Uncertain Reasoner's Companion.
Cambridge: Cambridge University Press, 1994.

[47] S. Parsons and A. Hunter
A review of uncertainty handling formalisms.
In *Applications of Uncertainty Formalisms*, Vol. 1455, *LNCS*, Berlin: Springer, 1998, 266–302.

[48] D. Rubin
Inference and missing data.
Biometrika, **63** (1976), 581–592.

[49] L. J. Savage
Foundations of Statistics.
New York: Wiley, 1954.

[50] J. Schubert and P. Svensson
Methodology for guaranteed and robust high level fusion performance: A literature study.
Technical Report FOI-D-0216-SE, Swedish Defence Research Agency, 2005.

[51] G. Shafer
A Mathematical Theory of Evidence.
Princeton: Princeton University Press, 1976.

[52] D. S. Sivia
Bayesian Data Analysis, A Bayesian Tutorial.
Oxford: Clarendon Press, 1996.

[53] F. Smarandache and J. Dezert (Eds.)
Advances and Applications of DSmT for Information Fusion.
Rehoboth: American Research Press, 2004.

[54] P. Smets
Decision making in the TBM: The necessity of the pignistic transformation.
International Journal of Approximate Reasoning, **38**, 2 (2005), 133–214.

[55] P. Smets and R. Kennes
The transferable belief model.
Artificial Intelligence, **66** (1994), 191–234.

[56] P. Smets and B. Ristic
Kalman filter and joint tracking and classification in the TBM framework.
In P. Svensson and J. Schubert (Eds.), *Proceedings of the Seventh International Conference on Information Fusion*, Vol. I, Stockholm, Sweden, International Society of Information Fusion, June 2004, 46–53.

[57] W. Stirling and A. Morelli
Convex Bayesianism decision theory.
IEEE Transactions on Systems, Man, and Cybernetics, (1991), 173–183.

[58] J. Sudano
Inverse pignistic probability transforms.
In *FUSION 2002*, International Society of Information Fusion, 2002, 763–768.

[59] H. Valpola and J. Karhunen
An unsupervised ensemble learning method for nonlinear dynamic state-space models.
Neural Computation, **14**, 11 (2002), 2647–2692.

[60] F. Voorbraak
A computationally efficient approximation of Dempster-Shafer theory.
International Journal of Man-Machine Studies, (1989), 525–536.

[61] F. Voorbraak
Partial probability: Theory and applications.
In G. de Cooman, F. G. Cozman, S. Moral, and P. Walley (Eds.), *Proceedings of the First International Symposium on Imprecise Probabilities and their Applications*, Gent University, 1999.

[62] P. Walley
Statistical Reasoning with Imprecise Probability.
London: Chapman and Hall, 1991.

[63] P. Walley
Measures of uncertainty in expert systems.
Artificial Intelligence, **83** (1996), 1–58.

[64] N. Wilson
Extended probability.
In *Proceedings of the 12th European Conference on Artificial Intelligence*, 1996, 667–671.

Stefan Arnborg took the civilingenjör (M.Sc.E.) exam in engineering physics at KTH in 1968, and the Ph.D. in information processing at KTH in 1972.

He worked as a supercomputer programmer at the Courant Institute (1966), with object-oriented programming systems at the Norwegian Computing Centre (1969–1970) and CAP Sweden (1971), as an OR analyst at the Swedish Defence Research Institute FOA (1972–1979), and as a distributed systems and computer guru at Philips Financial Terminal Systems (PEAB, 1979–1982). He spent sabbatical terms at the University of Oregon (1986, 1989, 1993) and was part-time advisor to the Swedish Institute of Computer Science SICS. He is the director of the theory group and of the M.Sc.E. program in computer engineering. He has been a professor of computer science at KTH since 1982.

His research interests are algorithms and complexity, verification, computer algebra, data engineering and uncertainty management, data mining, human brain informatics, command and control, and aesthetics in engineering education.

Information for Authors

The Journal of Advances in Information Fusion (JAIF) is a semi-annual, peer-reviewed archival journal that publishes papers concerned with the various aspects of information fusion. The boundaries of acceptable subject matter have been intentionally left flexible so that the journal can follow the research activities to better meet the needs of the research community and members of the International Society of Information Fusion (ISIF).

Restrictions on Publication: The International Society of Information Fusion (ISIF) publishes in JAIF only original material that has not been either published or submitted for publication in an archival journal. However, prior publication of an abbreviated and/or preliminary form of the material in conference proceedings or digests shall not preclude publication in JAIF when notice is made at the time of submission and the previous publications are properly referenced in the manuscript. Notice of prior publication of an abbreviated version of the manuscript must be given at the time of submission to the transactions. Submission of the manuscript elsewhere is precluded unless it is refused for publication or withdrawn by the author. Concurrent submission to other publications and this journal is viewed as a serious breach of ethics and, if detected, the manuscript under consideration will be rejection.

Sanctions for Plagiarism: If a manuscript is substantially similar to any previous journal submission by any of the authors, its history and some explanation must be provided. If a manuscript was previously reviewed and rejected by a different journal, the authors shall provide to the area editor copies of all correspondence involving the earlier submission. In addition, the authors must discuss the reasons for resubmission and be prepared to deliver further material if more is requested. Submission of a previously rejected manuscript is acceptable as long as the motive for submission to JAIF is explained and the authors have addressed the feedback of the earlier reviewers. The authors of a manuscript that is found to have been plagiarized from others or contain a crossover of more than 25% with another journal manuscript by any of the authors will incur sanctions by the ISIF. The following sanction will be applied for any of the above-noted infractions: (1) immediate rejection of the manuscript in question; (2) immediate withdrawal of all other submitted manuscripts by any of the authors to any of the ISIF publications (journals, conferences, workshops); (3) prohibition against all of the authors for any new submissions to ISIF publications, either individually, in combination with the authors of the plagiarizing manuscript as well as in combination with new coauthors. The prohibition shall continue for at least two years from notice of suspension.

Types of Contributions: Contributions may be in the form of regular papers or correspondence. The distinction between regular papers and correspondence is not one of quality, but of nature. Regular papers are to be a well-rounded treatment of a problem area, while a correspondence item makes one or two points concisely. In regular papers, the title, abstract, and introduction should be sufficiently informative to illuminate the essence of the manuscript to the broadest possible audience and to place the contributions in context with related work. The body of the manuscript should be understandable without undue effort by its intended audience. Correspondence should be less discursive but equally lucid. Authors should be aware that a well-written correspondence are usually published more rapidly than a regular paper.

Submission of Manuscript for Review: Manuscripts are submitted electronically via the internet at http://jaif.msubmit.net. For peer review, manuscripts should be formatted in a single column and double spaced.

General Information for JAIF Authors and Submission of Final Manuscript: General information for JAIF authors is available at http://www.isif.org/jaif.htm, and specific document preparation information for final manuscript for publication is given below and can be found at http://www.isif.org/JAIF Manuscript Preparation1.doc.

Copyright: It is the policy of the ISIF to own the copyright to the technical contributions it publishes; authors are required to sign an ISIF copyright transfer form before publication. This form appears in JAIF from time to time and is available on the web site listed above. If a copyright form is not submitted with the manuscript, it will be requested upon contribution acceptance. Publication will not take place without a completed copyright form.

Page Charges: Since ISIF has elected to support the publication of JAIF with the revenues of conferences and membership dues, page charges are not mandated for publication in JAIF. Page charges for journals of professional societies are traditional, and at some time in the future, page charges may be introduced to cover the expenses associated with the publication of JAIF. Payment of page charges will not be a prerequisite for publication.

Discloseability: The ISIF must, of necessity, assume that material submitted for publication is properly available for general dissemination to the audiences for which JAIF serves. It is the responsibility of the author, not ISIF, to determine whether disclosure of materials requires prior consent of other parties, and, if so, obtain it.

Preparation of Manuscript for Publication: After a manuscript is recommended for publication by the editorial staff, the corresponding author prepares the manuscript according to the guidelines below and submits the final version of the manuscript as a pdf file and the supporting manuscript files. Before prepar-

ing their manuscript for publication, authors should check for the more update version of the guidelines at http://jaif.msubmit.net.

All manuscripts must be submitted electronically in pdf format with the supporting files in one of the following formats: Plain TeX, AMS-TeX, LaTeX, or MS-Word with the MathType extension for equations and in-text mathematical material. The pdf file must be created with graphics resolution set to 300 dpi or greater. The author must approve the pdf file because it will be used as the basis for rectifying any inconsistencies in the supporting files. The supporting files must be prepared according to the following guidelines.

- Text files should include title, authors, affiliations, addresses, a footnote giving the dates for submission and revision and name of the editor that handed the review, and an optional footnote acknowledging a sponsor for the research. The information must be included in the text files in this order.
- Abstract must be included. The abstract must include the key words for the manuscript and include no more than 300 words for a regular paper and 150 words for a correspondence.
- Authors should number main section headings as 1, 2, 3, etc., and subsections as 1.1, 1.2, or 2.1. All headings should be typed with title format (i.e., first letter caps and lower case)—not in ALL CAPS. The typesetter will convert to all caps in final formatting as required.
- Authors should format references very carefully according to the examples given below. Style for authors' names are initials followed by last name and it must be followed precisely. References must be listed alphabetically by (i) last name of first author, (ii) initials of first author, (iii) last name of second author, (iv) initials of second author, etc. For manuscripts with common authors and order of authors, the date of publication should be used to select order with the earlier publications being list first. The names of publications must be spelled completely. In the references, the author should use only approved abbreviations that include: No., Vol., p., pp., AIAA, IEEE, IEE, and SPIE.
- Authors who use one of the TeX variants must provide a list of their macros and the files for the macros. Authors who use LaTeX must include the bbl and aux files.
- All figures must be submitted electronically as HIGH resolution (300 dpi or better) in Color graphics or GRAYSCALE (where shading is involved) or BLACK AND WHITE (if simple line art) graphics files (tif, eps, jpg, bmp). Each figure must be supplied as a separate graphics file. Graphics (or captions) should NOT be embedded in the text files. The figures must be included in the pdf file of the full article and the pdf file must be created with graphics resolution set to 300 dpi or greater.
- A separate file including all figure captions must be included.
- Each table must be submitted in a separate file. Tables (or captions) should NOT be included in text files and should be in form of DATA—rather than graphics—files.
- A separate file including all table captions must be included.
- A separate text of the biography of each author must be submitted. The text file should be less than 500 words.
- Separate graphics files of each author's photo should be provided as a grayscale graphics file or a color graphics file.

Examples of the references are alphabetized correctly and listed below.

BOOK:

[1] R. E. Blahut
Theory and Practice of Error Control Codes. Reading, MA: Addison-Wesley, 1983.

PROCEEDINGS ARTICLE:

[2] T. Fichna, M. Gartner, F. Gliem, and F. Rombeck
Fault-tolerance of spacebome semiconductor mass memories.
In Twenty-Eighth Annual International Symposium on Fault-Tolerant Computing, Digest of Papers, 1998, 408–413.

BOOK:

[3] P. K. Lala
Fault Tolerant and Fault Testable Hardware Design. Englewood Cliffs, NJ: Prentice-Hall, 1985.

WEB SITE:

[4] National Semiconductors Inc.
Homepage: http://www.national.com.

PROCEEDINGS ARTICLE:

[5] C. Paar and M. Rosner
Comparison of arithmetic architectures for reed-solomon decoders in reconfigurable hardware.
In *Proceedings of the Symposium on Field-Programmable Custom Computing Machines*, Apr. 1997, 219–225.

JOURNAL ARTICLE:

[6] N. R. Saxena and E. J. McCluskey
Parallel signature analysis design with bounds on aliasing.
IEEE Transactions on Computers, **46**, 4 (Apr. 1997), 425–438.

[7] C. I. Underwood and M. K. Oldfield
Observations on the reliability of cots-device-based solid state data recorders operating in low-Earth orbit.
IEEE Transactions on Nuclear Science, **47**, 4 (June 2000), 647–653.

WEB SITE:

[8] Xilinx Inc.
Homepage: http://www.xilinx.com.

INTERNATIONAL SOCIETY OF INFORMATION FUSION

ISIF Website: http://www.isif.org

BOARD OF DIRECTORS*

2004–2006	2005–2007	2006–2008
Erik Blasch	Per Svensson	David Hall
Jean Dezert	Darko Musicki	Ivan Kadar
Xiao-Rong Li	Eloi Bossé	Elisa Shahbazian

*Board of Directors are elected by the members of ISIF for a three year term.

PAST PRESIDENTS

Pierre Valin, 2006	Xiao-Rong Li, 2003	Yaakov Bar-Shalom, 2000
W. Dale Blair, 2005	Yaakov Bar-Shalom, 2002	Jim Llinas, 1999
Chee Chong, 2004	Parmod Varshney, 2001	Jim Llinas, 1998

SOCIETY VISION

The International Society of Information Fusion (ISIF) seeks to be the premier professional society and global information resource for multidisciplinary approaches for theoretical and applied information fusion technologies.

SOCIETY MISSION

Advocate

To advance the profession of fusion technologies, propose approaches for solving real-world problems, recognize emerging technologies, and foster the transfer of information.

Serve

To serve its members and engineering, business, and scientific communities by providing high-quality information, educational products, and services.

Communicate

To create international communication forums and hold international conferences in countries that provide for interaction of members of fusion communities with each other, with those in other disciplines, and with those in industry and academia.

Educate

To promote undergraduate and graduate education related to information fusion technologies at universities around the world. Sponsor educational courses and tutorials at conferences.

Integrate

Integrate ideas from various approaches for information fusion, and look for common threads and themes—look for overall principles, rather than a multitude of point solutions. Serve as the central focus for coordinating the activities of world-wide information fusion related societies or organizations. Serve as a professional liaison to industry, academia, and government.

Disseminate

To propagate the ideas for integrated approaches to information fusion so that others can build on them in both industry and academia.

Call for Papers

The Journal of Advances in Information Fusion (JAIF) seeks original contributions in the technical areas of research related to information fusion. Authors of papers in one of the technical areas listed on the inside cover of JAIF are encouraged to submit their papers for peer review at http://jaif.msubmit.net.

Call for Reviewers

The success of JAIF and its value to the research community is strongly dependant on the quality of its peer review process. Researchers in the technical areas related to information fusion are encouraged to register as a reviewer for JAIF at http://jaif.msubmit.net. Potential reviewers should notify via email the appropriate editors of their offer to serve as a reviewer.

Additional Copies

Additional copies of JAIF may be purchased through
FedEx Kinko's
440 Barrett Parkway, Suite 38
Kennesaw, Georgia 30144-4918
770-429-1389
usa1552@fedexkinkos.com

The pricing is as follows:

1 to 9 copies - $12.95 each plus tax and shipping
10 to 25 copies - $9.95 each plus tax and shipping
26 plus copies - $8.95 each plus tax and shipping

Journal of Advances in Information Fusion Volume 1 Number 1 July 2006 ISSN 1557-6418